BEng. Individual Project Report

Imperial College London

Department of Computing

Blockchain-mediated Layered Access to Data

Author:
Frederick Lindsey

Supervisor:
Dr. William Knottenbelt

February 18, 2017

Abstract

Over the last several hundred years, the way we access and manage the world's data has radically changed. Recounting medieval times when there was little to no public record of a person's assets or information, this presents a stark comparison to today's society, where our data and identities are traded on a global market, often without our knowledge. This paper, and it's accompanying proof of concept, seeks to describe a method of reinstating the ownership of data that was once commonplace in previous centuries, without compromising on the free flowing and global nature of communication today.

TODO: Add information from report findings and more focused text

Acknowledgements

I'd like to thank Dr. William Knottenbelt and Dr. Robert Learney for their support and guidance through this project. Without their support this project would not have been a success.

TODO: This is more of a placeholder at the minute whilst the project is in development

Contents

1 Introduction 2

2 Background 3

2.1 Social Issues . . . 3

2.1.1 Commoditisation of personal (and private) data . . . 3

2.2 Public De-centralised Ledgers . . . 3

2.2.1 Introduction to Public De-centralised Ledgers . . . 3

2.2.2 Relevance . . . 3

2.2.3 Blockchain.info . . . 3

2.2.4 Ethereum . . . 3

2.3 Proxy Re-Encryption . . . 4

2.3.1 Introduction to Proxy Re-Encryption . . . 4

2.3.2 Applying Proxy Re-Encryption . . . 4

2.3.3 ZeroDB . . . 5

3 Time-based Access For Encrypted Data 5

3.1 Layered Data Access . . . 5

4 Implementation Considerations 6

5 Project Plan 8

6 Evaluation Plan 9

6.1 Demonstrable core functionality . . . 9

6.1.1 Secure Distributed Storage . . . 9

6.1.2 Secure Layered Access . . . 9

6.1.3 Time-based Access . . . 9

6.1.4 Transparent and Public Logging . . . 9

6.1.5 Secure Access and Access Management . . . 10

6.2 Experimentation and Validation . . . 10

6.3 User testing and evaluation . . . 10

7 Bibliography 11

8 Appendix 12

1 Introduction

> "**When it comes to control over our own data, health data must be where we draw the line.**" (Wilbanks and Topol 2016)

Over the last 350 years, the general public has proceeded, often unwittingly, to give up their right to privacy by exchanging personal data for the convenience of the modern world. One might attribute the origin of this movement in the UK to the introduction of paper money by the Bank of England (The Bank Of England 2002) in 1694. This event heralded the idea of giving information about a person to an institution, be it a corporation or a government, in exchange for convenience. Before this time, one might have kept their savings 'under the mattress' and therefore there would be no sharing of one's wealth with another party. Whilst bank notes were introduced as a means to raise funds for a war, they also required the depositors (as a whole) to identify exactly how much money they had as a group. At this stage, this imposes no constraint on the depositor to give up any part of his unique identity, only form part of a wider, anonymous identity (the group).

2 Background

At present, not all sources have been cited. This will be done in future drafts.

2.1 Social Issues

At the core of the motivation for this project lay several issues corresponding to the way in which society has been manipulated over time. It is my belief that we find ourselves in the current position without any ownership of our data because we've been keen (even greedy) as a society to reap the benefits of our data without considering the longer term security effects. We have dismissed the need to care and be responsible for our data. Below, I have highlighted the key domains in which we lack control that we should have over our personal data. Whilst written as a piece of fiction, we should be aware and concerned that ignoring the social issues with data transfer allows a world to form much similar to that of George Orwell's 1984 (Orwell 1949) - we consider the likes of corporations synonymous with that of the 'Big Brother' character.

2.1.1 Commoditisation of personal (and private) data

There is no doubt that search tools such as those offered by Google and Microsoft, retail stores such as those offered by Amazon, and social networks such as Facebook and Twitter, dramatically enhance our lives and give us capabilities we would never have otherwise. Often as consumers we can be extremely eager to accept these benefits without considering the means with which they are offered to us.

2.2 Public De-centralised Ledgers

2.2.1 Introduction to Public De-centralised Ledgers

TODO

2.2.2 Relevance

TODO

2.2.3 Blockchain.info

TODO

2.2.4 Ethereum

TODO

2.3 Proxy Re-Encryption

2.3.1 Introduction to Proxy Re-Encryption

"In a proxy re-encryption scheme a semi-trusted proxy converts a ciphertext for Alice into a ciphertext for Bob without seeing the underlying plaintext" (Green and Atheniese 2006)

Fundamentally, proxy re-encryption is the process of taking a message M_a, encrypted by a party P_a, and re-encrypting it to be passed to party P_b. The message is then in the form of M_b, such that it is only readable by party P_b. Through the re-encryption process, the message is never actually decrypted, such that the data is never revealed to any non-trusted parties (including the proxy itself). This process relies on the functional relationship between the two ciphertexts, with the characteristics of the proxy re-encryption processed determined by the topology of this function.

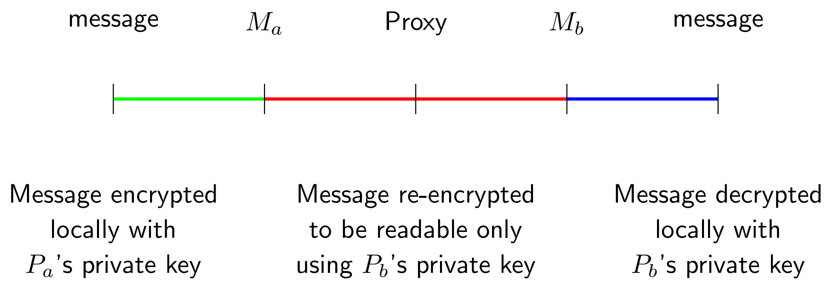
message
M_a
Proxy
M_b
message
Message encrypted locally with P_a's private key
Message re-encrypted to be readable only using P_b's private key
Message decrypted locally with P_b's private key

Figure 1: Example message journey through a proxy re-encryption scheme.

In figure 1, whether data is handled or manipulated by a fully trusted entity or not is indicated using green/blue and red lines respectively.

The encrypted message M_a is passed to the semi-trusted proxy along with the re-encryption key (expressed as $f(E_a, e_b)$[1])

2.3.2 Applying Proxy Re-Encryption

The core part of the project relates to how you encrypt and distribute data without negatively impacting the ownership of the data. Furthermore, other parts of the project specification need to be maintained. Proxy reencryption will therefore provide the backbone of the project, albeit in various states.

We start with taking parties P_a and P_b. These are two separate parties who wish to use the implementation. P_a wants to share some data with P_b. P_a encrypts data D_a with their public key such that the data is now only readable by P_a using P_a's private key, E_a. The data is then transmitted through to the backend where it is stored as a master copy. When P_a wants to retrieve the data from online, the private key which corresponds to the data is required.

TODO: Much more detail required here

[1] E_x represents the private key of a party x, e_x represents the public key of a party x.

2.3.3 ZeroDB

TODO: Is this relevant?

3 Time-based Access For Encrypted Data

TODO

3.1 Layered Data Access

TODO

4 Implementation Considerations

Since the core requirements of the project are architecture-independent to some degree, a simple solution which is not fully decentralised is a good option to begin with.

At current, a potential implementation utilises the following technologies:

- AWS S3 (file/blob storage)
- AWS EB (scalable application endpoints)
- AWS RDS (SQL database)
- Ethereum (Blockchain-based decentralised applications)
- Electron (desktop application wrapper)

Whilst centralised, Amazon Web Services (AWS) provides a solid and stable platform for the deployment of the application. Well-tested and production-ready software such as PostgreSQL, Amazon S3, and Amazon Elastic Beanstalk (running docker) not only provide a durable and dependable deployment solution, but also allow the development and production environments to be mirrored such that development need not take place online.

The application would be distributed such that it could run on desktop operating systems (Mac and Linux (Debian) initially[2]). This means that the backend services of the project would be distinct from any front-end, whilst a public API offers the opportunity for third parties to contribute data and encourage user adoption.

The project architecture could be modelled, to some extent, by the following:

[2] Deployment of other packages may incur more overhead unnecessary at this point

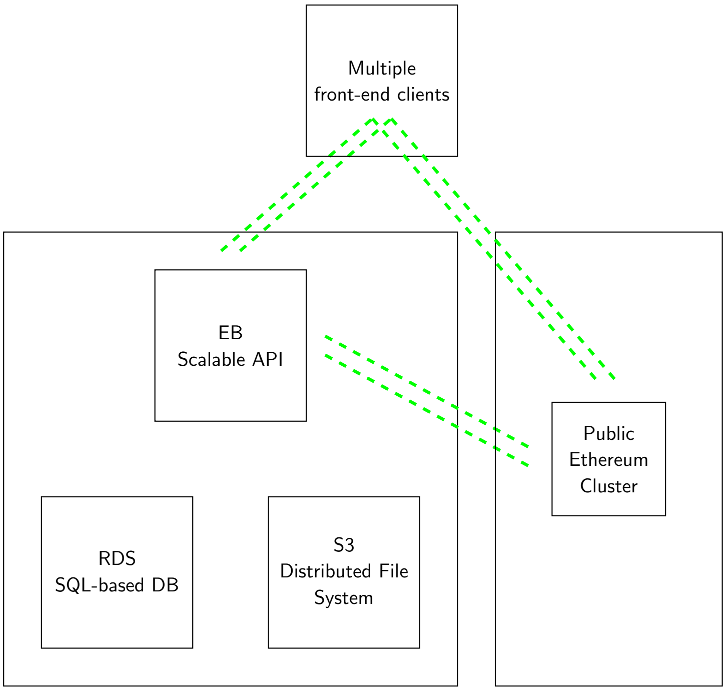

Figure 2: Potential architecture for proof of concept. Shows links between different systems using green, dashed lines. Each inner box referring to a specific service within a location (public or private). All links are made over public infrastructure.*The Electron application that will be built as part of this proof of concept is assumed to be one of many front-end clients.*

5 Project Plan

Over the course of the project, there will be several stages and for each of these stages an associated key milestone. Throughout this process, a mixture of research, discovery, implementation, and user testing must occur in order to give the project a well-rounded and well-evaluated outcome.

Below, I explain the desired timeline of the project and indicate when I intend to reach specific milestones.

10th February	Complete Interim Report (Draft) At this point crucial research should have been undertaken to understand the viability of the project and to be able to assess it's preliminary chances of success.
17th February	Prototype architecture Whilst working on preliminary architecture, the implementation of core features should have begun. By this point a more structured and well understood architecture should be written.
3rd March	Evaluate and consider second marker's comments and advice After discussions with the second marker to the project, evaluate the points of concern and any advice on how to achieve the greatest success with the project. Move forward with this in mind and adjust any proposed plans or process as necessary.
5th March	First working prototype A basic working prototype must include the ability to read and write data at a basic level
7th May	Second working prototype A second working prototype should be made by this point. It must include basic desktop functionality and the ability to use multiple accounts to access 3rd party data (at different levels).
15th May	Project Health Checkup Given the length of time between here and the working prototype having been demonstrated to the supervisor in March, by this point some evaluation should be done in terms of reconsidering what the final goals are for the project in the remaining month.
19th June	Final Report Due
26th June	Submission of Project Archive

Table 1: Plan for project (given progress thus far)

6 Evaluation Plan

6.1 Demonstrable core functionality

Below are lists of functionalities that are required for the project to have achieved success. In the case where the user is expected to be able to give multiple inputs in an either-or fashion, partial success is still achieved by implementing a subset of those inputs.

6.1.1 Secure Distributed Storage

- Store encrypted data only, such that readable by the primary data owner only
- Data input and output should never be decrypted - this should be provable

6.1.2 Secure Layered Access

- Allow the use of different access layers across a dataset
- A party P_a that is a member of a data-layer access group D_b, but may have extra (superceding) permissions will have access that is an extension of a party P_b who is only a member D_b.
- Disallow a party to see data exists if they do not have read access

6.1.3 Time-based Access

- Allow any user to request data from any other user who they can identify
- Allow nominated 3rd parties to access data upon the successful granting of a request
- Granted access to data is time-dependent using one of two inputs:
  - Set remaining time period
  - Set access termination date

6.1.4 Transparent and Public Logging

- For every access of a file (through the system), a record is written to a public ledger
- All records of user access must be encrypted such that the primary data owner is the only party that can read them
- The collapse of the system would not stop a user from viewing the logs for their data

- The use of a public ledger does not cost the primary data owner anything

6.1.5 Secure Access and Access Management

- Unauthorised access to a user's account (maliciously or otherwise) does not allow reading a user's data
- An actor must not be able to write to the access system such that they gain unauthorised access to data

6.2 Experimentation and Validation

In order to verify that the above functionalities have been met, a series of experiments will need to be performed. These will include but are not limited to:

- Create two users. Use the first to request data from the second (given a username or other identity parameter).
- Simulate the use of a security hierachy and observe whether the system is able to handle this as one would expect.
- Validate that data for which access has been granted is accessible with the correct access permissions (multiple tests required)
- Validate that data for which access is given in a time-sensitive manner is no longer available once this time period expires
- Attempt to write transactions to the ledger the application uses to gain access to the user data
- Ensure that under single-user and multi-user loads, access is correctly implemented
- Simulate a malicious attack on a user's account and attempt to retrieve their data. Record what user security information is required as a minimum to access any part of the user's secure data.

6.3 User testing and evaluation

Qualitative user data will be assessed using the front-end of the application created for demonstration purposes. It is important that users who would currently access and update such systems do not feel pain in using the developed prototype. It is also important that the user experience is similar to that expected by potential users. It is my intention to use members of the college community of varying technical abilities and select members of the public who work in relevant industries to test the useability of the application and give feedback to improve the user experience.

I will attend the Wearable Technology Show[3] where I will try to get as much market data as possible on the viability and demand for such an application in the market place. This will be largely from wearable technology providers who, for the health care case study, would likely provide the infrastructure for data input from end users.

[3] London, UK-based event taking place on 7-8 March 2017 `http://www.wearabletechnologyshow.net/home`

7 Bibliography

Green, Matthew and Giuseppe Atheniese (2006). "Identity-Based Proxy Re-Encryption". In: URL: http://eprint.iacr.org/2006/473.pdf.

Orwell, G. (1949). *1984: A Novel ; Revised and Updated Bibliography*. Signet Classics. ISBN: 9780451524935. URL: https://books.google.co.uk/books?id=EevFmgEACAAJ.

The Bank Of England (2002). *A brief history of banknotes*. URL: http://www.bankofengland.co.uk/banknotes/Pages/about/history.aspx (visited on 01/14/2017).

Wilbanks, John T. and Eric J. Topol (2016). "Stop the privatization of health data". In: *Nature* 535.7612, pp. 345–348. DOI: 10.1038/535345a.

8 Appendix

List of Figures

1 Example message journey through a proxy re-encryption scheme. 4

2 Potential architecture for proof of concept. Shows links between different systems using green, dashed lines. Each inner box referring to a specific service within a location (public or private). All links are made over public infrastructure.*The Electron application that will be built as part of this proof of concept is assumed to be one of many front-end clients.* 7

List of Tables

1 Plan for project (given progress thus far) 8