Master Thesis: Open Specification of a user-controlled Web Service for Personal Data

G. Jahn

November 1, 2016

Abstract

TODO

Contents

1

Introduction

1.1 Motivation

- discriminate humans related to certain data, by interpreting results of deep learning and

  using neuronal networking and trying to extract information out of data haystacks
- correlation is no proof of causation

1.2 Purpose & Outcome

- Personal Data Store (aka. Service, Space, Vault, Cloud, Management/Manager), VRM (Vendor
  Relation Management) aka “CRM upside down”
- Open Source

1.3 Glossary

- serverless: https://auth0.com/blog/2016/06/09/what-is-serverless/
- digital footprints: TODO
- owner: person who controls (and probably hosts) the data service containing her personal data

2

Fundamentals

2.1 Personal data in the context of Big Data - Movement

- difference between users *profile/account data* and their *meta data*?

2.2 Personal data as of the Law

- who is the owner in what situation or under what circumstance?

- difference between “Personal Data” and the owner of that data?

2.3 Personal Data as a Product

- individuals then get in role of selling/offering it’s own data to those who were previously
  collecting them

2.4 Digital Identity

- identity defining data (e.g. history of personal ID card)
- with such a system a human being is represented by a non-physical abstraction of herself.
  Which essentially is a list of attributes, that are at least for legal and civil
  administration purposes important. Their values in total are unique and representing the
  corresponding human. Certain attributes hold unique values within it’s own context, for
  example the *social security number*.
- Thus it’s not necessary to know the values of all attributes in order to identify it’s
  owner
- therefore its imported to not see it as a reduction of a living individual

to some bits and

bytes

- what will happen with her data service after a person died?

2.5 Related Work

TODO

2.5.1 Existing Systems

- openPDS/safeAnswer [http://openpds.media.mit.edu/]
- TAS3 aka ZXID aka Synergetics (lead arch Sampo Kellomäki also Co-Authored openPDS papers)
- Higgins [https://www.eclipse.org/higgins/]
- Hub-of-All-Things [http://hubofallthings.com/what-is-the-hat/]
- ownyourinfo [http://www.ownyourinfo.com]
- PAGORA [http://www.paoga.com]
- PRIME/PrimeLife [https://www.prime-project.eu, http://primelife.ercim.eu/]
- databox.me (reference implementation w/ the “solid” framework)
- Microsoft HealthVault
- Industrial Data Space (german research project mainly driven by Fraunhofer Institute)
- Polis (greek research project from 2008) [http://polis.ee.duth.gr/Polis/index.php]
- MyData [https://mydatafi.wordpress.com/]
- Meeco (killing the ad provider middle man) [https://meeco.me/how-it-

works.html]

- RESPECT network [https://www.respectnetwork.com/]
- aWise AEGIS [http://www.ewise.com/aegis]

2.5.2 Existing Organisations

- Kantara Initiative (former “Liberty Alliance”) [https://kantarainitiative.org/]
- Open Identity Exchange [http://openidentityexchange.org/resources/white-papers/]
- Qiy Foundation [https://www.qiyfoundation.org/]

2.6 Application

2.6.1 Use Cases

TODO: should be more general, differently from each other

2.6.2 User Stories

- package shipment after buying sth online
- social network accessing arbitrary profile data
- making an online purchase
- credibility (requesting credit permission) validation by a certain financial institution:
  accessing arbitrary data
- patient/health record

- care (movement) data

3

Concepts

NOTE: here we discuss a variety of possibilities –> conceptual work

3.1 Data Ownership

- user-centric, full control

3.2 Identity Verification

- maybe go with a CA
  - revoking the cert which provides the authenticity of the individual's digital identity

    should only be possible with a two-factor secret. One part of this secret is owned by

    the CA and the other half has the individual behind the personal API

3.3 Authentication

- NOTE: does not mean this tool authenticates it's owner against third party platforms like

  OpenID does. but it could play the role of the 2n factor in a multi-factor authentication

  process (if the mobile-device-architecture was chosen)
- refers primarily to the process of a data consumer (third party, which needs the data for

  whatever reason) verifies her admission to request

3.4 Data Reliability

- is this data (in this case identity) certified or not (results in higher value)

3.5 Exposing Data

- pure/plain data request/resonse
- remote computation/execution (assuming there is no client for the consumer)
  like https://webtask.io/

3.6 Encapsulation

- containerization (coreos, rkt, mirageos aka unikernal)

3.7 Open Standards

- what standards can be used for which tasks

3.8 Hosting

- DYI
- Usability

4

Architecture

NOTE: based on the previous discussion, here we show our results and solutions

4.1 Requirements

4.1.1 Components

4.2 Possible Directions

- cloud or local storage
- which components can go where

4.3 Overview

- distributed architecture (e.g. notification/queue server + mobile device for persistence
  and administration)

4.3.1 $componentA

4.3.2 $componentB

4.4 Security

- the downside of having not just parts of the personal data in different places (which is
  currently the common way to store), is in case of security breach, it would increase the

possible damage by an exponential rate

Thereby all data is exposed at once, instead of not just the parts which

a single service

has stored

4.4.1 Environment

4.4.2 Transport

4.4.3 Storage

4.4.4 Authentication

- how should consumer authenticate?

4.4.5 Attack Scenarios

4.5 Plugins

- but for what? and not harm security at the same time

5

Data

- keep in mind to make it all somehow extendible, e.g. by using and storing

  corresponding schemas

5.1 Modelling

5.2 Categories

5.3 Types

5.4 Flows

5.5 Persistence

- database requirements

5.6 History

- data versioning
- access logs

5.7 Management and Administration

5.7.1 Access & Permission

- data needs to have an expiration date

5.8 Consumption

- how data will get into the system
- hwo is the user able to do that, and how does it works

5.8.1 Manually

5.8.2 Automatically

6

Interfaces

6.1 Internal

6.2 External

- should there be a way to somehow request information about what data is available/queryable,

  or would this be result in spam/crawler and security issues (also a question for the topic of permissions/sensibility level of certain data)

7

Conclusion

7.1 Ethical & Social Impact (TODO: or “Relevance”)

7.2 Business Models & Monetisation

- possible resulting direct or indirect business models
- owner might want

7.3 Challenges

7.4 Solutions

7.5 Future Work

7.6 Summary

- main focus
- unique features
- technology stack & standards
- resources