Project Proposal: GDPR Compliant Key-Value Stores

Archita Agarwal
aa12

Marilyn George
mgeorge5

Abstract

We propose to prototype a design for a GDPR-compliant key-value store for a social network application, with emphasis on the right to user access.

1 Introduction

We propose to design a key-value store compliant with the right of access by the data subject (Art. 15) [1]. Our target application will be a social media network, and the data in our key-value store will be along the lines of Facebook's TAO [2]. For the scope of this project, we will consider data generated by users (friend lists, friend groups, posts) and posts (likes, comments).

2 Design Sketch.

We plan to implement a per-user sharded design [3]. Each user will have a separate user-specific key-value store. All the keys associated with that user will be stored in this key-value store. We note that keys could be associated to more than one user, as in a post or a like or a comment. In our current design we plan to add these keys into to key-value stores for all users associated with them, while keeping track of the following types of associations for user roles:

- Owner-Viewer: An Owner owns the data, for instance a post. A Viewer can interact with this data, like or comment on it. When the Owner invokes the right to access it receives the data as well as all the interactions associated with the data. When the Viewer invokes the right to access it receives only the information associated with the interaction, but not the data itself - this belongs to the Owner. This is similar to the unidirectional edges in TAO.
- Owner-Owner: Both users own the data for this association. For instance, a friendship or group memberships. When the right to access is invoked by either of them they should receive the data. This is similar to TAO's bidirectional edges.

We ignore Viewer-Viewer associations currently as being out-of-scope of the right to access. We note here that the appropriate associations could handled by the application outside the key-value store - only relevant data need be stored in each user's shard. However, we believe keeping track of the associations will enable us to extend our design to incorporate the rights discussed below.

Extensions. Time permitting, we will add encryption for data at rest to comply with the security of data processing (Art. 32) within our key value store. We also wish to incorporate the right to data portability (Art. 20) and the right to erasure (Art. 17) [1] by extending our current design.

Implementation. We plan to implement this design using memcache-d and study the performance of the new design in comparison to unaltered memcache-d to analyze the cost of compliance to this right.

References

[1] Regulation (eu) 2016/679 of the european parliament and of the council (general data protection regulation). https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32016R0679#d1e3722-1-1. Accessed: 2019-10-10.

[2] Nathan Bronson, Zach Amsden, George Cabrera, Prasad Chakka, Peter Dimov, Hui Ding, Jack Ferris, Anthony Giardullo, Sachin Kulkarni, Harry Li, et al. {TAO}: Facebook's distributed data store for the social graph. In *Presented as part of the 2013 {USENIX} Annual Technical Conference ({USENIX}{ATC} 13)*, pages 49–60, 2013.

[3] Malte Schwarzkopf, Eddie Kohler, M. Frans Kaashoek, and Robert Morris. Position: Gdpr compliance by construction. In *To be part of Poly'19: Towards Polystores that manage multiple Databases, Privacy, Security and/or Policy Issues for Heterogenous Data*, 2019.