The Feed: A Decentralized Photo Sharing Application

Oluwapelumi Odimayo with Michael J. Friedman

Princeton University Department of Computer Science

Objectives

The primary objects of this project are to provide users with a social networking application in which they can:

- Join the network at anytime provided that they have a verified namespace registered with the Blockstack platform.
- Completely control who is authorized to view data and content they put onto the network
- Participate in the network without placing trust in a centralized entity/"network-overseer".
- At any time completely remove their trace from the network without hindering the operation of the rest of the network.

Introduction

Decentralized applications built on top of blockchains[1] have garnered much attention in recent years for the security properties afforded to them by these peer-to-peer (P2P) networks. The rise in popularity of these applications can be attributed to growing security and privacy concerns associated with traditional web application models. However, this departure also led to a departure from a traditional user experience.

This project introduces The Feed, a decentralized photo-sharing application built using the Blockstack platform.[2] The Feed is an instagram-like social network that is easy for the end-user to setup and use. It eliminates the costly operations associated with other decentralized applications by reducing interaction to the blockchain by only using it as a login and authorization/ verification services. The design presented is such that the security properties of the Blockstack platform are used to provide users with a network in which they own/control their data and, more importantly, no longer have to trust a central node or network overseer to participate in the network.

Design

The design of The Feed consists of three major components:

- User Server - User owned server that handles requests.
- User Storage - User owned storage where they can store their data.
- Application - Runs locally and sends requests to user server.

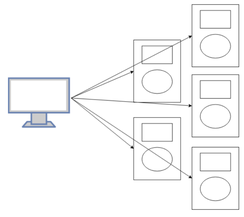

Figure 1: Current System Architecture

Important Consideration

Although this design eliminates the per-operation costs associated with other dapps, the cost of setting up and running user-servers does not scale well. The solution to this, and the original intent, is using Blockstack Multi-Reader Storage feature once it is released.

Implementation

The application, user servers, and user directory are all implemented using the Node.js Javascript framework. The user-server is deployed to Google Cloud App Engine. The API exposed by the user-server are HTTP request handlers that implement all requests as POST requests so that the body of each request can be verified using the authentication and permission system. The user storage is a simple MySQL (ClearDB)database deployed through the Heroku web service.

The user-directory is very simple in implementation. This also a simple Node.JS deployed with Heroku (just like the user-servers storage) that handles HTTP requests through an exposed API. Upon initialization the database contains a single table which contains the mapping between a user Blockstack ID and their server IP address.

Authentication and Permissions System

To authenticate the requester, the server will retrieve the public key associated to the user Blockstack ID and use this to verify the signed data. If the signature is valid, the server will then verify permissions to ensure that a non-owner is not trying to gain access to owner-only data. Once verified, the server will then decode the signed data and make sure that the request timestamp is within an appropriate range (for version 1 this is a few seconds). If a check fails at any point in this process, the server will respond with a failed result. Otherwise the server will send back the requested data.

Results

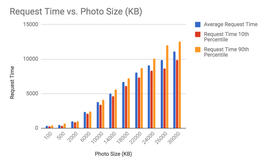

Figure 2: Request Times of Post Upload

Figure 3: Average Times for Photo Upload

Our results show the (time) cost of key operations involved in network participation.

Conclusion

The design of The Feed focuses on making reasonable performance tradeoffs to ensure that users within the network have as much control as possible over their data. Specifically, every user within the network runs the application locally, is provided with their own user-server, and controls access to their own user storage. For future work we plan to switch the system to Blockstack Multi-Reader Storage once it is released. Version one of The Feed can be found on github at https://github.com/michaeljfriedman/IBOB.

References

[1] Steem.
Steem: An incentivized, blockchain-based, public content platform.
In *SteemWhitePaper*, 2017.

[2] Jude Nelson Muneeb Ali, Ryan Shea and Michael J. Freedman.
Blockstack: A new internet for decentralized applications.
In *BlockstackTechnicalWhitepaper*, 2017.

[3] Jude Nelson Muneeb Ali, Ryan Shea and Michael J. Freedman.
Blockstack: A global naming and storage system secured by blockchains.
In *BlockstackTechnicalWhitepaper*, 2016.

Acknowledgements

Huge thank you to my partner Michael Friedman. Special thanks to Michael Freedman, our advisor, and the Blockstack team for providing the support necessary to complete this project.

Contact Information

- Email: oodimayo@princeton.edu
- Phone: +1 (207) 485 2523