Decentralized Messaging

Introduction

The objective of this project was to develop a decentralized messaging UI component for Web3. The broader intent was to find a way to overcome some of the inefficiencies that plague the current messaging systems. In the solution, the messaging component was developed as an npm library, which can be included in any web application that needs decentralized messaging features. The messaging protocol is end-to-end encrypted, peer-to-peer and uses ENS as a central registry for storing contact information. This messaging component also allows anyone to send messages from one ENS name to another easily.

Client Details

Name: Confidential   **|**   **Industry**: Software   **|**   **Location**: Europe

Technologies

Ethereum, Solidity, React. js, TypeScript, Node. js, Ethers.js, Rainbow Wallet, Wallet Connect, Metamask, Gnosis Chain

Project Description

The Current messaging systems (such as instant messages, chats, emails, etc.) are fraught with inefficiencies. For instance, email services lack efficient spam protection methods other than blocking and censoring. Also, Web2-based messaging services (like WhatsApp, Signal, etc.) function in silos, and therefore, cross-app communication is not possible. The objective of the project was to establish a common base standard for Web3-technology-based messaging. Further protocols and

Decentralized Messaging

applications can be built on top of it to create a silo-free, secure, interoperable and decentralized messaging ecosystem.

A system such as this not only allows users to exercise full control over their data and messages but also to choose the messaging app that best suits their needs and preferences And all this comes without the compulsion of having to be limited to a particular ecosystem.

Some of the key aspects of this Web3 messaging protocol are:

- Key-based identity
- Self-sovereign
- Decentralized
- No single point-of-failure
- Encrypted
- Privacy (information and metadata)
- Interoperability
- Anti-spam

The essential parts of the Web3 messaging system include:

- Central registry: Contains encryption and delivery information
- Delivery Service: Decentralized Network of delivery service nodes
- Protocol Layer: The protocol is peer-to-peer, end-to-end encrypted, with scalability, privacy and interoperability
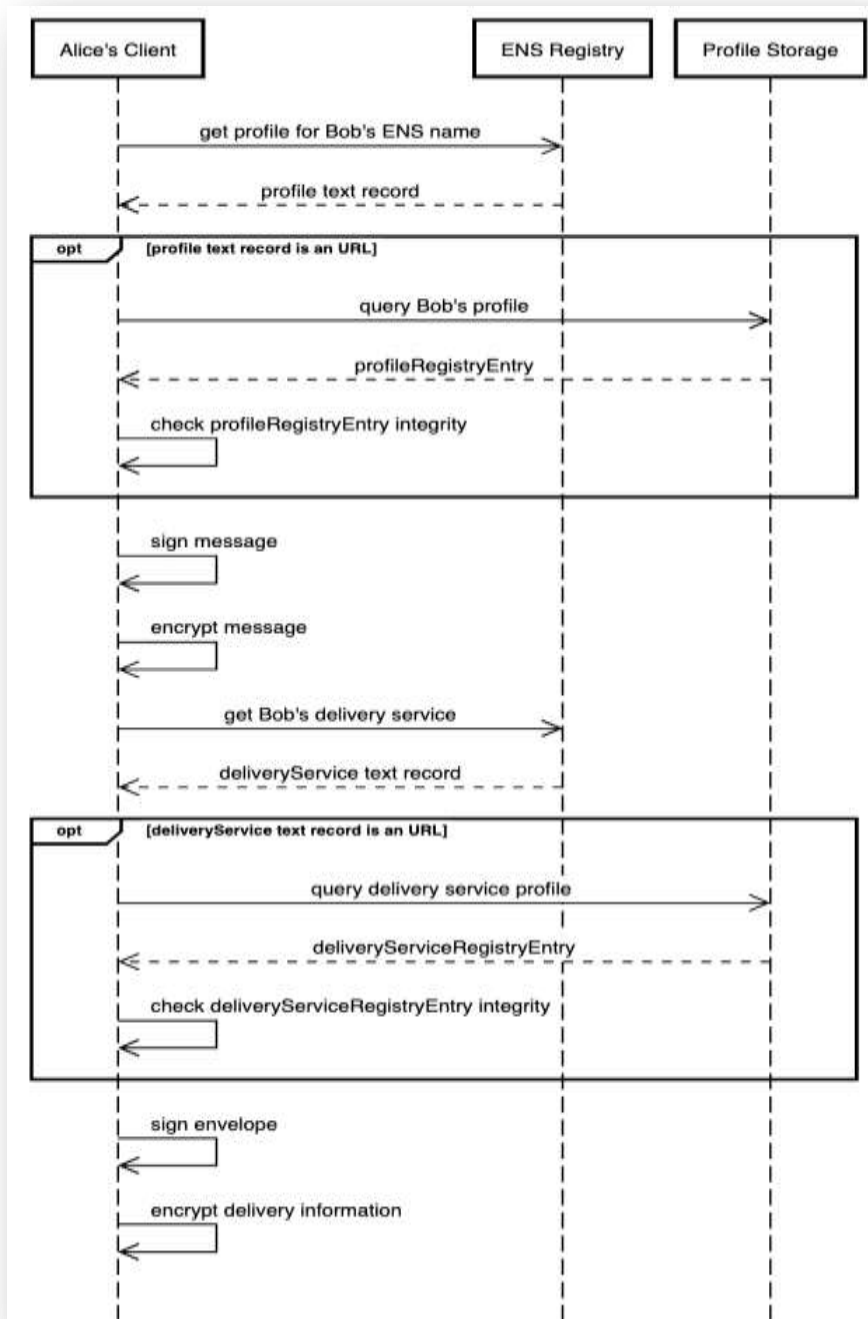
Decentralized Messaging

Architecture Diagram
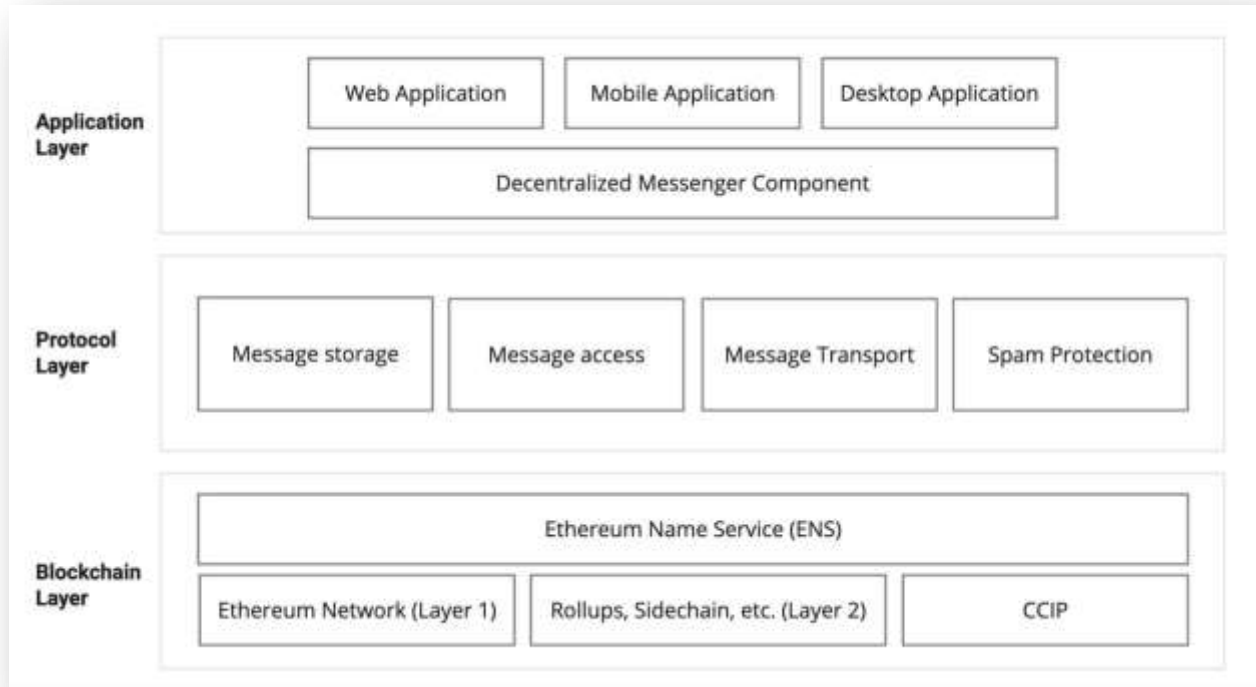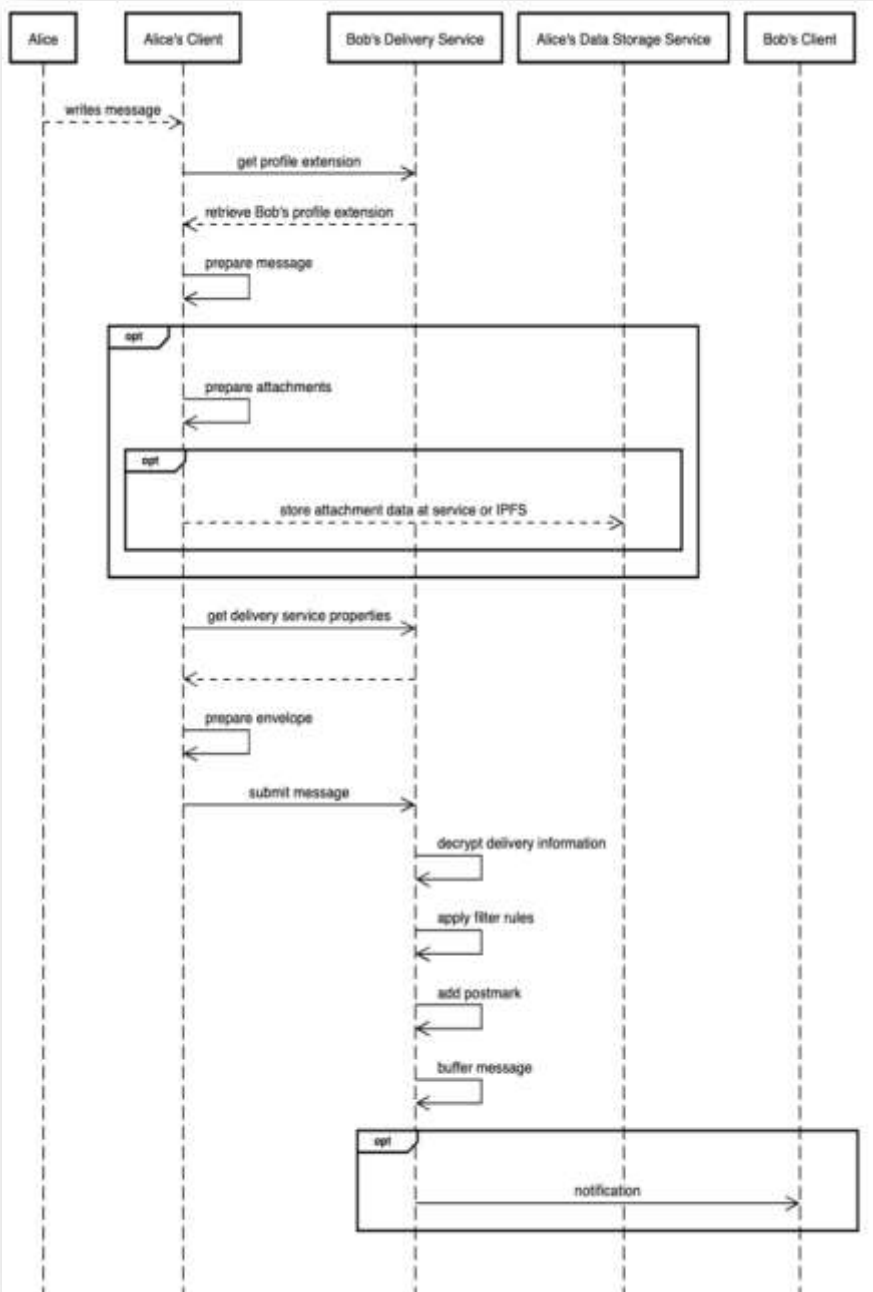
Decentralized Messaging

Sequence Diagram 1

Create message and envelope

Decentralized Messaging

Sequence Diagram 2

Message processing at the delivery service

Decentralized Messaging

Screenshots