CS408 Computer Science Project
Fall 2016

김현승 HyunSeung Kim 20120048
Schneider Léo 20166434
Thévenet Pierre 20166488

Project Presentation

General Idea

Our project consists in developing an instant messaging application for web and android platform, allowing two people to discuss whether they are online or not, and allowing them to communicate without recording message in anywhere when possible.

Features

- P2P direct connection
- Video/Audio/Text chat by 2 or more people
- File/Location/Calendar sharing
- Mini-games
- Conference options
  - Memo writing by multiple users at the same time
  - E-mail notification to participants
  - Message hierarchy which allows viewing summary of conversation.

Problem Statement

Instant messaging services usually implies giving your conversation data to the service provider, with absolutely no control on it. Although this can practical for some reasons, this is sometime useless and inefficient. Whattsapp, KakaoTalk and Telegram for example forces the user to send the messages to their server, this could be avoided. Indeed, when two people are connected, these two may want to talk directly to each other without allowing an eventual eavesdropper from the provider to spy on the discussion.

Nowadays there are services specifically designed to solve this privacy problem. Nevertheless they lack the convenience of the other ones, for example letting two people send messages to each other even when one is disconnected.

The problem is therefore to let the users choose between convenience and privacy, when they have the choice. If one person is disconnected, the other user should be able to decide either to wait until he connects, or to send him a message that will be stored in the provider's server to permit the message to be delivered. And when these two people are connected, they should be able to avoid sending their messages via the provider's server, and therefore having a private conversation.

Therefore we think that a service that not only encrypts messages but also avoid unnecessary data collection is an interesting project.

Technical Challenges

- Designing efficient, reusable, maintainable software architecture.
- Database architecture and use of efficient SQL requests.
- Mechanism that facilitates concurrent memo writing by multiple users.
- Peer authentication.
- Resolving NAT shadowing.
- Designing a convenient user interface.