InterMessage

Дмитрий Саютин
Александр Федоров

Санкт-Петербургский Академический Университет

22.02.2018

Мотивация проекта

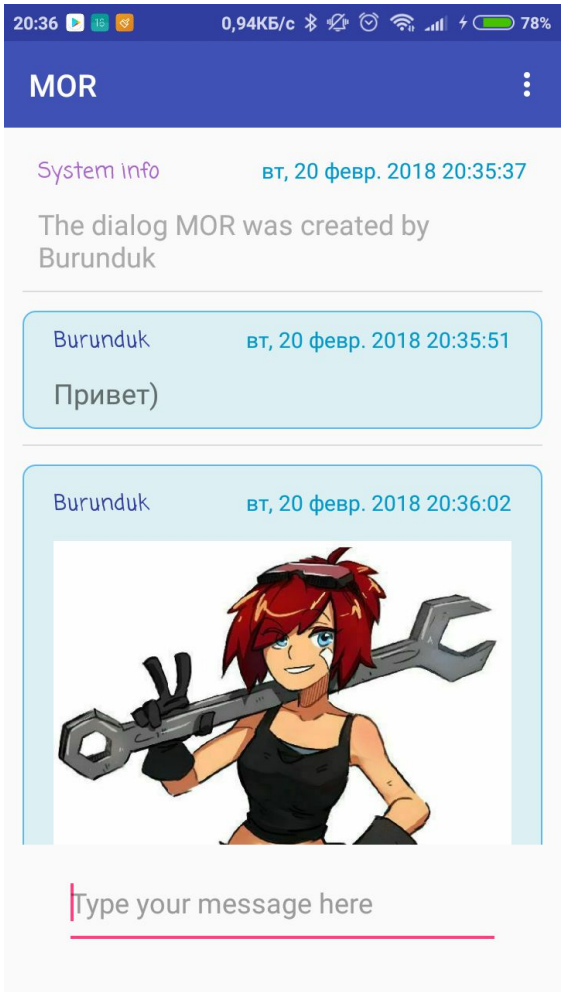
- Ситуация А:
  Вы студент и в вашем общежитии кто-то не заплатил за интернет, но сеть wi-fi все еще поднята. Как сообщить всему общежитию, что сегодня опять отключили горячую воду?

Мотивация проекта

- Ситуация А:
  Вы студент и в вашем общежитии кто-то не заплатил за интернет, но сеть wi-fi все еще поднята. Как сообщить всему общежитию, что сегодня опять отключили горячую воду?
- Ситуация Б: Вы грибник и поехали группой за грибами в лес, а там нет ни Интернета ни общей сети wi-fi. Но вы не промах и взяли с собой мощный wi-fi роутер. Тогда, чтобы сообщить другим людям, что вы нашли грибную поляну, вам теперь не нужно идти искать их в лесу, надеясь, что поляну за это время не найдут другие грибники.

Решение

- Децентрализованный мессенжер работающий в локальной сети



Аналоги

- Многие системы видео-звонков используют возможности peer-to-peer связи (например skype).

Аналоги

- Многие системы видео-звонков используют возможности peer-to-peer связи (например skype).
- Есть некоторое количество мессенджеров в локальной сети (в основном для ПК).

Аналоги

- Многие системы видео-звонков используют возможности peer-to-peer связи (например skype).
- Есть некоторое количество мессенджеров в локальной сети (в основном для ПК).
- А вот децентрализованных в локальной сети уже немного!

Аналоги

- Многие системы видео-звонков используют возможности peer-to-peer связи (например skype).
- Есть некоторое количество мессенджеров в локальной сети (в основном для ПК).
- А вот децентрализованных в локальной сети уже немного!
- Ближайший известный аналог — FireChat.

Отличия от FireChat

- В основном ориентирован на публичные комнаты, а не приватные диалоги.

Отличия от FireChat

- В основном ориентирован на публичные комнаты, а не приватные диалоги.
- У него только сообщения и картинки.

Отличия от FireChat

- В основном ориентирован на публичные комнаты, а не приватные диалоги.
- У него только сообщения и картинки.
- Пытается комбинировать связь через интернет и локальные взаимодействия (в частности при первом запуске нужно подтверждать почту!).

Отличия от FireChat

- В основном ориентирован на публичные комнаты, а не приватные диалоги.
- У него только сообщения и картинки.
- Пытается комбинировать связь через интернет и локальные взаимодействия (в частности при первом запуске нужно подтверждать почту!).
- Умеет передавать сообщения еще и по bluetooth.

Отличия от FireChat

- В основном ориентирован на публичные комнаты, а не приватные диалоги.
- У него только сообщения и картинки.
- Пытается комбинировать связь через интернет и локальные взаимодействия (в частности при первом запуске нужно подтверждать почту!).
- Умеет передавать сообщения еще и по bluetooth.
- Использует чужие устройства для передачи сообщений.

InterMessage

- Работает по WiFi в локальной сети

InterMessage

- Работает по WiFi в локальной сети
- Полное End-To-End шифрование (RSA + AES)

InterMessage

- Работает по WiFi в локальной сети
- Полное End-To-End шифрование (RSA + AES)
- Автоматические цифровые подписи у каждого сообщения

InterMessage

- Работает по WiFi в локальной сети
- Полное End-To-End шифрование (RSA + AES)
- Автоматические цифровые подписи у каждого сообщения
- Поддержка различных типов сообщений (например картинки)

InterMessage

- Работает по WiFi в локальной сети
- Полное End-To-End шифрование (RSA + AES)
- Автоматические цифровые подписи у каждого сообщения
- Поддержка различных типов сообщений (например картинки)
- Другие устройства могут помогать с доставкой сообщений, но только внутри одного чата.

Шифрование

- У каждого человека есть публичный и приватный RSA-ключ.

Шифрование

- У каждого человека есть публичный и приватный RSA-ключ.
- Публичный ключ общеизвестен и является внутренним идентификатором в системе.

Шифрование

- У каждого человека есть публичный и приватный RSA-ключ.
- Публичный ключ общеизвестен и является внутренним идентификатором в системе.
- Как следствие, например, невозможна MITM-атака (система сразу обнаружит попытку подмены адресата).

Шифрование

- У каждого человека есть публичный и приватный RSA-ключ.
- Публичный ключ общеизвестен и является внутренним идентификатором в системе.
- Как следствие, например, невозможна MITM-атака (система сразу обнаружит попытку подмены адресата).
- Так как с помощью RSA можно шифровать только сравнительно короткие сообщения, то мы дополнительно используем AES.

Шифрование

- У каждого человека есть публичный и приватный RSA-ключ.
- Публичный ключ общеизвестен и является внутренним идентификатором в системе.
- Как следствие, например, невозможна MITM-атака (система сразу обнаружит попытку подмены адресата).
- Так как с помощью RSA можно шифровать только сравнительно короткие сообщения, то мы дополнительно используем AES.
- С помощью RSA шифруется AES ключ и дальнейший обмен данными происходит через AES.

Сеть: логический уровень

- Проблема: так как нет центрального сервера, то нет общей истории сообщений, на каждом клиенте сообщения могут приходить в разном порядке.

Сеть: логический уровень

- Проблема: так как нет центрального сервера, то нет общей истории сообщений, на каждом клиенте сообщения могут приходить в разном порядке.
- Нужно как-то научится синхронизировать данные.

Сеть: логический уровень

- Проблема: так как нет центрального сервера, то нет общей истории сообщений, на каждом клиенте сообщения могут приходить в разном порядке.
- Нужно как-то научится синхронизировать данные.
- Наблюдение: если кто-то отправил нам в чат несколько сообщений, то вряд ли разумна ситуация если более ранее сообщение не доставилось, а более новое доставилось.

Сеть: логический уровень

- Проблема: так как нет центрального сервера, то нет общей истории сообщений, на каждом клиенте сообщения могут приходить в разном порядке.
- Нужно как-то научится синхронизировать данные.
- Наблюдение: если кто-то отправил нам в чат несколько сообщений, то вряд ли разумна ситуация если более ранее сообщение не доставилось, а более новое доставилось.
- Следовательно состояние нашего чата прекрасно описывает некоторый массив целых чисел — сколько сообщений у нас есть от соответствующего человека.

Сеть: логический уровень

- Проблема: так как нет центрального сервера, то нет общей истории сообщений, на каждом клиенте сообщения могут приходить в разном порядке.
- Нужно как-то научится синхронизировать данные.
- Наблюдение: если кто-то отправил нам в чат несколько сообщений, то вряд ли разумна ситуация если более ранее сообщение не доставилось, а более новое доставилось.
- Следовательно состояние нашего чата прекрасно описывает некоторый массив целых чисел — сколько сообщений у нас есть от соответствующего человека.
- Осталось понять как эти массивы синхронизировать чтобы не посылать бесполезные пакеты.

Сеть: логический уровень

- Давайте для каждого человека в нашем чате поддерживать их массивы, в том виде как мы их виде в последний раз.

Сеть: логический уровень

- Давайте для каждого человека в нашем чате поддерживать их массивы, в том виде как мы их виде в последний раз.
- Тогда мы более-менее легко можем поддерживать список людей, у которых хотя бы в одном чате недостаточно новый список сообщений.

Сеть: физический уровень

- С физической точки зрения мы используем UDP броадкасты и прямые соединения по TCP.
- Когда человек впервые входит мы посылаем UDP броадкаст, с помощью которого другие клиенты во-первых понимают где в случае чего нас найти, а во вторых мы посылаем список людей, которым мы хотим доставить сообщения.

Сеть: физический уровень

- С физической точки зрения мы используем UDP броадкасты и прямые соединения по TCP.
- Когда человек впервые входит мы посылаем UDP броадкаст, с помощью которого другие клиенты во-первых понимают где в случае чего нас найти, а во вторых мы посылаем список людей, которым мы хотим доставить сообщения.
- Если кто-то видит себя в таком списке, то он начинает процесс обновления сообщения.

Сеть: физический уровень

- С физической точки зрения мы используем `UDP` броадкасты и прямые соединения по `TCP`.
- Когда человек впервые входит мы посылаем `UDP` броадкаст, с помощью которого другие клиенты во-первых понимают где в случае чего нас найти, а во вторых мы посылаем список людей, которым мы хотим доставить сообщения.
- Если кто-то видит себя в таком списке, то он начинает процесс обновления сообщения.
- Плохая новость: на каждое новое сообщение в чате запускается достаточно много новых синхронизаций.

Сеть: физический уровень

- С физической точки зрения мы используем `UDP` броадкасты и прямые соединения по `TCP`.
- Когда человек впервые входит мы посылаем `UDP` броадкаст, с помощью которого другие клиенты во-первых понимают где в случае чего нас найти, а во вторых мы посылаем список людей, которым мы хотим доставить сообщения.
- Если кто-то видит себя в таком списке, то он начинает процесс обновления сообщения.
- Плохая новость: на каждое новое сообщение в чате запускается достаточно много новых синхронизаций.
- Хорошая новость: после любой синхронизации хотя бы один массив сведений у какого-то человека увеличивается.

Сеть: преимущество такого подхода

- Таким образом, если A и B находились в одной сети, чатились, а потом B уехал за границу к C, то если в общем чате от A, B, C были какие-то сообщения от A, то они доставятся и до C!

Сеть: технологии

- `TCP`, `UDP`
- Неблокирующие операции.
- Мессенджер порождает ровно один дополнительный поток, обслуживающий и клиентские запросы к мессенджеру, и всю сеть.

База данных

- Все сообщения и информация о пользователях сохраняется на устройстве с помощью SQLite.

База данных

- Все сообщения и информация о пользователях сохраняется на устройстве с помощью SQLite.
- Еще есть локальное кеширование для того, чтобы не нужно было перезагружать сообщения, например при повороте экрана.

Поиск людей в округе

- Проблема: откуда брать людей, которых можно добавить в диалог?

Поиск людей в округе

- Проблема: откуда брать людей, которых можно добавить в диалог?
- Решение: Давайте предлагать всех людей, которые сейчас находятся поблизости
- Тогда легко добавлять группу новых людей.

Распределение труда

- Дмитрий Саютин — сеть, шифрование, логика работы приложения.
- Александр Федоров — интерфейс, база данных, а также взаимодействие между собой частей программы

.end()

Вопросы?