Proposal to

AnonTalk: An Anonymous Communication Application for Mobile Devices

Yale University CPSC 490, Spring 2013

Danqing Liu

Abstract

The goal of this project is to build an anonymous communication application for the iOS mobile platform. Anonymous communication is in increasing need in “high-stakes” contexts, when users want to safely exchange sensitive information, and due to their portability, smart phones and other mobile devices are increasingly becoming the major communication tools people rely on. Therefore, there are situations where such anonymous mobile communication proves useful to users.

This project is an implementation of the Dissent communication protocol. Developed at Yale, Dissent is developed as an improvement on the security level of existing anonymous communication protocols, and to balance the trade-off between size and strength. It also aims to achieve scalability and resilience to slow or unreliable clients. With all its benefits, Dissent is still in its early research stage, and so this application partly serves as a practicality check for this novel protocol by implementing it for one of its major potential use cases.

Project Description

The app to be built is a chatting app. From the perspective of a user, it supports:

- Logging in so that the user is connected with a Dissent server or with an ad-hoc Dissent network.
- Joining a Dissent group. The user may either see a list of public groups that anyone can join, or join a private group that grants the user access.
- Posting messages to the Dissent group. The user should be allowed to choose to be identifiable or anonymous. For example, in some groups it may be the case that a journalist is seeking for information from sensitive people, and the journalist may want to be identifiable so not to be confused with everybody else, whereas other people want to stay anonymous.

The application is going to be a native iOS (Cocoa Touch) app, with its underlying communications handled by the Dissent protocol, which needs to be migrated to the iOS platform. The Dissent protocol should stand as an independent framework with its own APIs so
that apps that wish to adopt the Dissent protocol can do so easily by including the relevant files. This also goes along with Apple's Model-View-Controller structural recommendation for applications, and makes it easier to migrate the protocol to OS X desktop platform, which is largely similar to iOS.

The application will include one or more extra features which Dissent does not yet support but may be either important for good user experience or important use case. These include but not limit to:

- Push Notification integration. Due to the nature of Dissent, the user may need to wait for a significant period of time before posting the first message. Therefore, push notifications may be a useful tool to notify the user when the conversation can actually begin.
- File exchanges. So far Dissent is text-message only, although the underlying algorithm is generic and allows the transmission of binary data. It may sometimes be useful to share files anonymously, and so such functionality is desirable. However, it must be carefully thought out and experimented because even though the communication process is anonymous, the files themselves may not be. For example, a Microsoft Word file is likely to contain author information.

**Technical Challenges & Difficulties**

The major challenge is to migrate and re-implement Dissent protocol, which is currently only available on Linux, on Apple iOS (and more generally, Objective-C) platform. Dissent relies on several existing libraries which may or may not be compatible with iOS, and even if certain libraries are available, redundant components need to be carefully removed so that the application can stay slim, which is important for mobile apps.

Another challenge is to accommodate the limited processing power and battery life of mobile devices. Dissent is crypto-heavy, and it may be slow on smart phones and/or drain battery life very fast. Therefore, appropriate measures are necessary to make sure that it can actually exist in a stable and useable way on mobile devices. This is in fact part of the goal of this project, aka. proving that the protocol can live on mobile devices, which is one of its major potential use cases.

A last challenge is the interaction with the iOS device. The app needs to be user-friendly and intuitive to use, needs to interact with various resources available to the device, and most importantly needs to be absolutely safe. This means for example that on the local device, no identifiable data should be stored, and no information should be secretly sent out.
**Deliverables**

The main deliverable of this project will be a complete iOS application that achieves the functionalities outlined above. It should be a native Cocoa Touch application supporting iOS 6+ platforms.

There should be a core Dissent framework package for iOS and more generally, Objective-C platforms. Included as part of the main application, this framework should by itself provide all necessary APIs that handle anonymous communication requests through the Dissent protocol, so that other applications on iOS and OS X platforms can simply include this framework to adopt Dissent protocol.

If Push Notification Service proves to be necessary for the application, relevant server code that handles the communication between server(s) and target client devices should be included.

Finally there should be a report that summarizes the results of the project, including documentation for the core Dissent framework for future use.

**Milestones**

**Stage I.** Setting up the basic structure and logic of the app. This includes all the essential logic, views and UI elements of the application. This should be done by February.

(Also by the end of February I expect to understand most relevant parts of Dissent well so it can be implemented in the second stage.)

**Stage II.** Integrating Dissent into the application. As the current Dissent package runs on Linux, it needs to be migrated to the Apple iOS platform with all necessary supporting libraries and frameworks. By the end of March, basic Dissent framework should be migrated and implemented, so that the application can perform the most basic communications via Dissent.

**Stage III.** Functionality enhancements. After making sure that the basic Dissent protocol works on iOS, more work should be done to i.) optimize Dissent to make it work more smoothly on a mobile device that has limited processing power and battery life, and ii.) customize Dissent (or add layers on top of it) to satisfy more mobile communication needs and circumstances. This is the last stage of the project.