CS460 Project: PAssword Manager with Remote Android Client (PAMRAC)

Fred Douglas

2016-05-07

1 Overview

My project was a password manager, primarily targeted at Android but supporting synchronization across devices. A user should ideally run their own server, but one server can also host multiple users; it's safe to assume the server is curious, or even malicious.

The interesting part of the project is a key recovery feature. Users can link up with friends (just by verifying a key fingerprint; I think it's basically acceptably convenient), and give out Shamir k-of-n secret shares of their master key. There are some details to make it safer and more controllable: a user can choose to exclude some information from being recoverable, a user can designate a few friends without whom recovery is impossible (i.e. the k shares must include one of these people), and keys are rotated periodically, to defend against theft of shares from honest friends (either the adversary must get enough shares within the rotation period, or the adversary is an actual coalition of your friends trying to hurt you).

2 Progress

The system's behavior is outlined by a Google Protocol Buffer specification, which describes every type of message clients and servers might send to one another, along with every type of object that might be stored in a file. I have been careful to keep all client-server interactions independent from each other, to minimize the complex/far-reaching effects to be considered when reasoning about security. This specification (pamrac.proto, in the top level directory) serves as a relatively compact full description of the system. It also dictates the logic of the server and clients, as they do all communication and storage in terms of the message formats it defines.

For this class, I implemented the server (in C++), a Java client, and the UI shell of an Android client. I also wrote an interface for converting password stores to/from PAMRAC, which other password managers could implement to become compatible with PAMRAC; I wrote a quick sample implementation that takes a directory full of plain text files.

I wrote all of the Java client in the last few weeks, and it is "done" but not quite working: Java is not loading the Shamir secret sharing library I intended to use, despite compiling it, and code that uses it, with no problem. I don't really understand why. I had earlier started by taking care of what I had expected would be the hardest part: a custom Android keyboard that can be programmatically reconfigured to input arbitrary strings into

a selected text field, with that keyboard being available system-wide. If I realized Java, even outside of Android, would be such a problem, I would have started out purely in C++, and not moved to Java until I had something working.