CS 495: Capstone Progress Report 4
Karen Dana
4 November 2014

Since we forked our Capstone project, this will require me to set up my own server on which to store the database. I will be working on using Hypertext Transfer Protocol Secure (HTTPS) to transfer data between the Android mobile application and the database on the server. I will also be using temporary-access tokens that will be issued by the server. The user will have to enter his credentials to get a token, which will be attached with every request the user sends to the server.

For the operating system on the server, I will use CentOS, because it is stable and manageable, which should remove the worry of updates causing problems [1]. CentOS is based on commercial releases of Red Hat Enterprise Linux, but it is free and is managed by the community instead of the corporation [2]. It does not have company support, so all of the support comes from the community. Since it is such a large base, this should provide enough help if any errors or problems arise.

To handle the web server, I will use Apache HTTP Server 2.2 [3]. It is included with Red Hat Enterprise Linux along with service modules to enhance its functionality [4]. Apache HTTP Server is the most popular web server right now, so there will be plenty of support for it [3].

To create tokens, I am going to use the built-in account manager system from the Android API that provides a standard way to authenticate users. This will simplify the token process and handle access-denied scenarios. Tokens will have a time period for which they are valid, and must be renewed after it expires. The phone will request an authentication token from `AccountManager` which will in turn ask the `AccountAuthenticator` of the matching account type if it contains a token for the current user. If there is a token, it is returned. If there is not one, it will display the `AccountAuthenticatorActivity` which will prompt the user to log in. After the user logs in, the server will send back a token which will be kept in the `AccountAuthenticator` for future reference. Finally, the `AccountManager` will return the token to the application [5].

[1] http://www.centos.org/
[2] https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/index.html
[3] http://httpd.apache.org/
[4] http://www.centos.org/docs/5/html/Deployment_Guide-en-US/ch-httpd.html
[5] http://udinic.wordpress.com/2013/04/24/write-your-own-android-authenticator/

Tuesday, November 18, 2014

The server will be set up, running CentOS as the operating system, with Apache HTTP Server 2.2. The mobile application will be able to store values in the database. Users will be able login which will send their credentials to the database over the network where they will be encrypted. Authentication tokens will be implemented.

Tuesday, December 2, 2014

The information gathered from user profiles that is stored in the database can now be queried for likes; a user can choose a like, and the database will return a list of the current users who also have that same interest. Default groups will have been completed.