GeoGuard

Daniel Ingersoll
Daniel Kimmelmann
Raul Moncayo
Alexander Nguyen
November 27 2015

Working Prototype Known Problems Report

What doesn't work, what causes it, where does it break, and how to fix it

Known Problems

Passwords are overwritten. This is caused because we do not check the input when inserting the data in insert.java. This can be fixed by checking the user input against the existing database for any matchings and warning the user.

Passwords can be any string including blank and special. This is caused because we do not check the input when inserting the data in insert.java. This can be fixed by checking the user input before inserting. While we have not been able to break it ourselves, we think it might be possible to do some damage with the right string. Additionally the passwords screen gets messed up with extremely long passwords dominating the page.

Passwords can be geotagged without any data turned on and still say they were inserted with a location. This is caused because we do not check for a network connection before inserting the data in insert.java. This can be fixed by first checking for a network connection before inserting.

Passwords can be inserted without any data turned on and still say they were saved but weren't pushed to the online database and therefore on logout and login the password will disappear. This is caused because we do not check for a network connection before inserting the data in insert.java and we do not have any way of holding the data until a connection appears. This can be fixed by first checking for a network connection before inserting and giving a warning or by making a queue structure to hold data temporarily until a network connection appears.

Passwords are not actually deleted when there is no network connection. This is the same issue as insertion without a network connection and can be solved with a queue that waits or flat out disable editing when there is not network.

The passwords screen not updated correctly. When reloading the page, passwords will disappear when leaving a tagged location but not reappear when re entering that location. This can be fixed by logging out then logging back in. We suspect there is something being messed

up with the location and something is being stored and not updated correctly because the fact that logging out and in fixes the issue. The problem is likely in the onCreate() method in localpasswords.java because it can be fixed by logging out then back in.

Multiple people can login to the same account simultaneously and data will not be saved correctly. We have not really tested multiple users signing in to the same account but the way we handled updating the online database, there will be some errors. We update the server by sending the entire file containing the password data. So if person A inserts a and then person B inserts b, both logged into the same account, the next login that fetches from the database will only see data b and data a will be lost. This could be fixed by limiting one person access to the database at a time. Additionally person A would not see data b until they logged into another account and then into the profile again. This is because we only fetch from the database when logging in and the last account differs from the one being logged in. This is to allow for offline access to data. This could be fixed by updating the profile on login if the user has a network connection.

Extremely large files will break the system, which is created by adding lots of passwords. For encryption we read the entire file into an array so theoretically a large enough file will not be read properly. Most phones have enough ram that this shouldn't be an issue because the files should remain relatively small since all we store is text. To fix this we could encrypt files in chunks but we ran into other issues mainly buffer and padding errors when doing that.

Clicking the insert button too fast will create a bunch of messages and they keep going despite closing app. This is caused because every insert creates a toast notification and they go for as long as specified. It does not break but is an annoying feature. To fix this we could have some logic to close the current toast in insert.java.

Homebase can be reset anytime. This also is not a serious error but it should be noted that homebase can be reset anywhere so the whole security by locations notion is completely negated. We could fix this by asking for the password again for confirmation. This could also be a good feature though if the user needs to see all their passwords at a given time.

The homebase does not appear immediately after setting it. The user must logout then log back in. We think this is because the main screen is only ever loaded once and going back just brings up an old state. To fix this we could have an update main screen function that would rebuild it when a user sets the homebase. The error is located in MainScreen.java

Rotation on the Passwords and Homebase screens blocks the geotagged passwords. This can be fixed by creating a more dynamic layout. There is a scrollable layout and that would fix the issue. We could also fix it by disabling rotation. The issue is located in the xml file.

Rotation on the Sign In and Sign Up screens blocks the enter button. This can be fixed by creating a more dynamic layout. There is a scrollable layout and that would fix the issue. We could also fix it by disabling rotation. The issue is located in the xml file.

Creating account sometimes does not work. The spinning creating account icon shows up and disappears but the main screen is not launched an an account is not made. Pressing the enter button a second time works. Could possibly be a timing error and talking to the database takes too long so it times out. The error is somewhere in signup.java. The error is hard to reproduce so it is hard to tell what exactly the error is.

After homebase is set, the user has to logout/login in order for changes to take effect. It breaks in mainscreen.java probably because something is set in onCreate() wrong. Another place it could be going wrong is location services which is also causing problems elsewhere so fixing the location methods might fix the issue.
Sometimes just closing the app isn't enough to purge all data, user must logout/login to reset their distance to their homebase. This could again be an issue with onCreate() or the location services which fixing might fix the issue.

With no network services it seems like the geolocation features do not work properly. A new location cannot be fetched but things do not disappear when this happens. The problem can be resolved by using a new canGetLocation() function and implementing the case when this is false to clear the fields instead of not doing anything. The problem can be seen in insert, homebase, passwords and the main screen. The root cause is probably in tracker.java.

Rotating the application on the login screen during the login spinning popup crashes it. Error is in unlock.java. The error is because of an illegalArgumentException when dismissing the popup because the view changed. A way to fix this would be to lock the view on a sign in.

Passwords names are displayed in all caps but the name the name is case sensitive so Test and TEST appear as two different entries both named "TEST." This could be a little confusing for the user or could be a feature to allow for multiples of the same name. To fix this we could display the name differently. They are generated in LocalPasswords.java and that is where the problem is.

Encryption is only on the online files so it could be more secure. To fix this we would have to encrypt local files too but this also could slow down the app so it was a design choice. Files are private on the app anyways.