Danger Log for HW1

YiHao Hu, YiFan Xiao

01/22:

Potential Danger:

Unregistered users or users who have not logged in can access the carpool app.

Solution:

Use user authentication to secure the main portal page, before entering the main page (or the app), check if the user is logged in, if not, user will be redirected to the Login Page.

01/23:

Potential Danger:

A non-driver user might access the driver page to claim rides.

Solution:

Before entering the driver page, check if the user has selected to be a driver, and check if the vehicle information is complete, if not, user will be redirected to the user profile page to fill out the related information.

01/23:

Potential Danger:

We found that the driver can potentially search and claim the rides that their vehicle cannot accommodate. Also, the more forms we open to user, the more attack surfaces are available.

Solution:

For each driver, we filter all available rides that a driver is eligible to claim for the driver. The result shows right up on the driver login page.This way we pretend drivers to do crazy things on the web app.

01/25:

Potential Danger:

We found that user might attack the website through various web forms via Cross Site Request Forgery.

Solution:

For each web form that is of “POST” type in the html file, we insert “{% csrf_token %}”, which is a Django built-in feature for CSRF protection, to prevent this potential attack.

01/26:

Potential Danger:

We found the driver might potentially claim a ride that is requested by himself.

Solution:

We add criterias to the filter so that driver can only claim the rides that fits his vehicle type while he is not the ride owner. Also, a driver of a ride cannot be a sharer of that ride (such result won’t show up in the available ride on driver page).

01/30:

Potential Danger:

In the requirement log, it requires all rides can have access to edit the ride detail, including ride owner and the sharers.

However, with this requirement, we found that the sharer that join the ride may gain the access to edit to the current ride, including changing the destination or the arrival time of the ride, which is not ideal since the owner should be the only person that can alter the destination and the arrival time.

Solution:

We only allow ride's owner to edit the ride's detail. When the user is the sharer of the ride, we will restrict its access to the detail of the page, and do not present a edit button for the sharer to access the edit function.

02/01:

Potential Danger:

We found that if the user has joined the ride as a ride sharer, and after that if she logged in as a driver, she is still able to search the ride she already joined as ride sharer.

Solution:

We adjust the search result with the restriction that the driver cannot be a part of the ride sharer or ride owner if she wants to claim a ride. So the search result will eliminate those results that list current user as ride sharer or ride owner.

02/01:

Potential Danger:

We found that unauthorized users can enter urls that are used to confirm the ride, and claim the ride without permission, even though they are not matched with the rides (originally the rides that are not matched with them will not show, but they can randomly guess the ride id and attempt to claim that ride).

Solution:

Before the ride is confirmed, we check whether the requested user to be capable to claim the ride (including whether the user is driver, whether the user's vehicle is able to claim the ride). If not, we will deny the claim request, and the user will be taken back to the home page.

Special Feature:

1. Easy to use User interface.
2. Automatic search available rides for drive to claim, saving the time to manually input the parameters that is already stored in the system, also avoiding the situation that driver searches for rides that they are not eligible to claim
3. Easy switch between rider and driver mode (use links on the nav bar).