[**https://drive.google.com/drive/u/0/folders/1mhvCwOolH2tBQM1hJx-hhvO8yZfKlHU-**](https://drive.google.com/drive/u/0/folders/1mhvCwOolH2tBQM1hJx-hhvO8yZfKlHU-)

1. Why do you consider your password requirements to be strong. Note that user friendliness is also important, so requiring a 25-character password is not a good idea. A brief analysis (for brute force attack) would be good.

I decided to have a password requirement of 9 characters made up of at least one upper- and lower-case letter, a number, and a symbol on the number keys. From our analysis in class, there are 52731073839781512 possibilities which could take over 600 days to crack by brute force.

1. How did you store your passwords? Document the file/entry structure.

Because we assumed that the only way to access the data files is through the application, I chose to store the actual passwords instead of the encrypted once. I did however check the encrypted password against the one the user input during the login process to authenticate the user.

1. How did you implement role-based access controls?

Since we didn’t have to worry about privileges on any files, I included the user’s roles in the password file that could be edited by the admin or human resources to change their access. I simply assigned a number to each job/role and then checked against that number for every operation the user tried to make.

1. List all events you are logging.

I logged every time someone logged in, viewed the admin or human resource pages, viewed their own personal info or the personal info of others, when a new user was added or removed, when personal information was changed, and every time a calculation was performed. For all of these actions, I also made sure to record who performed the action and who was affected by it.