Isaiah Narisma

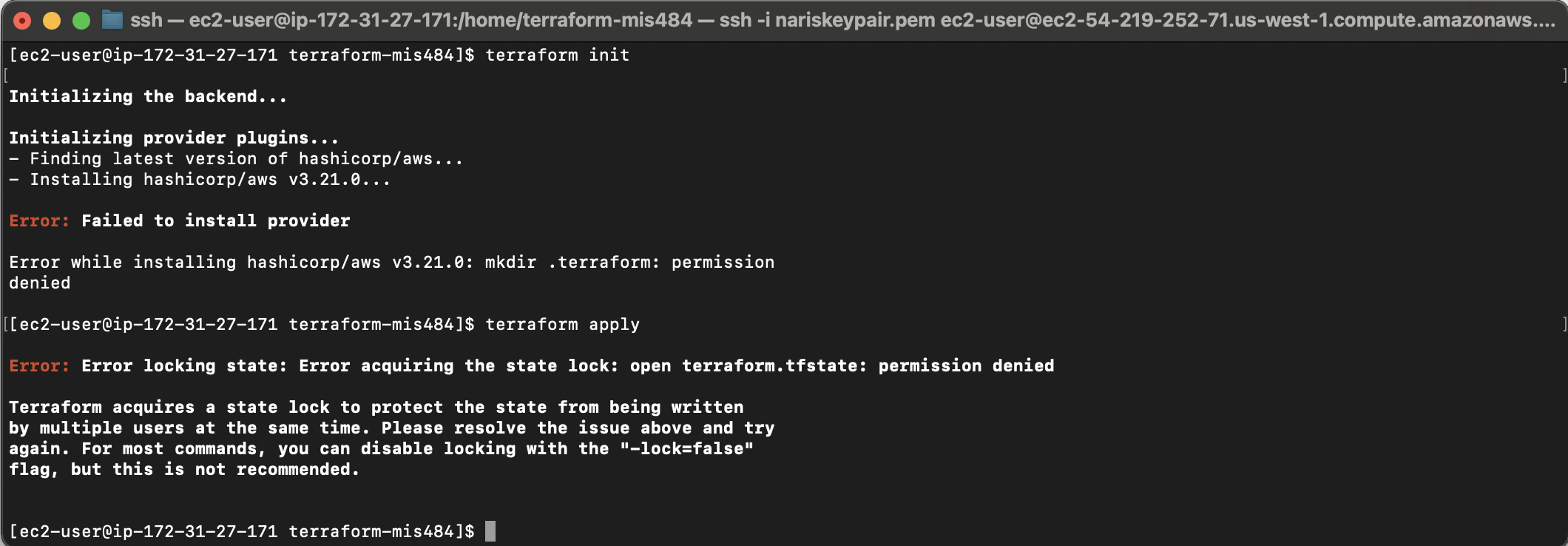
Professor Underhill

MIS 484-6 (01) 42896

14 December, 2020

GCP Automation Cloud Deployment Reflection

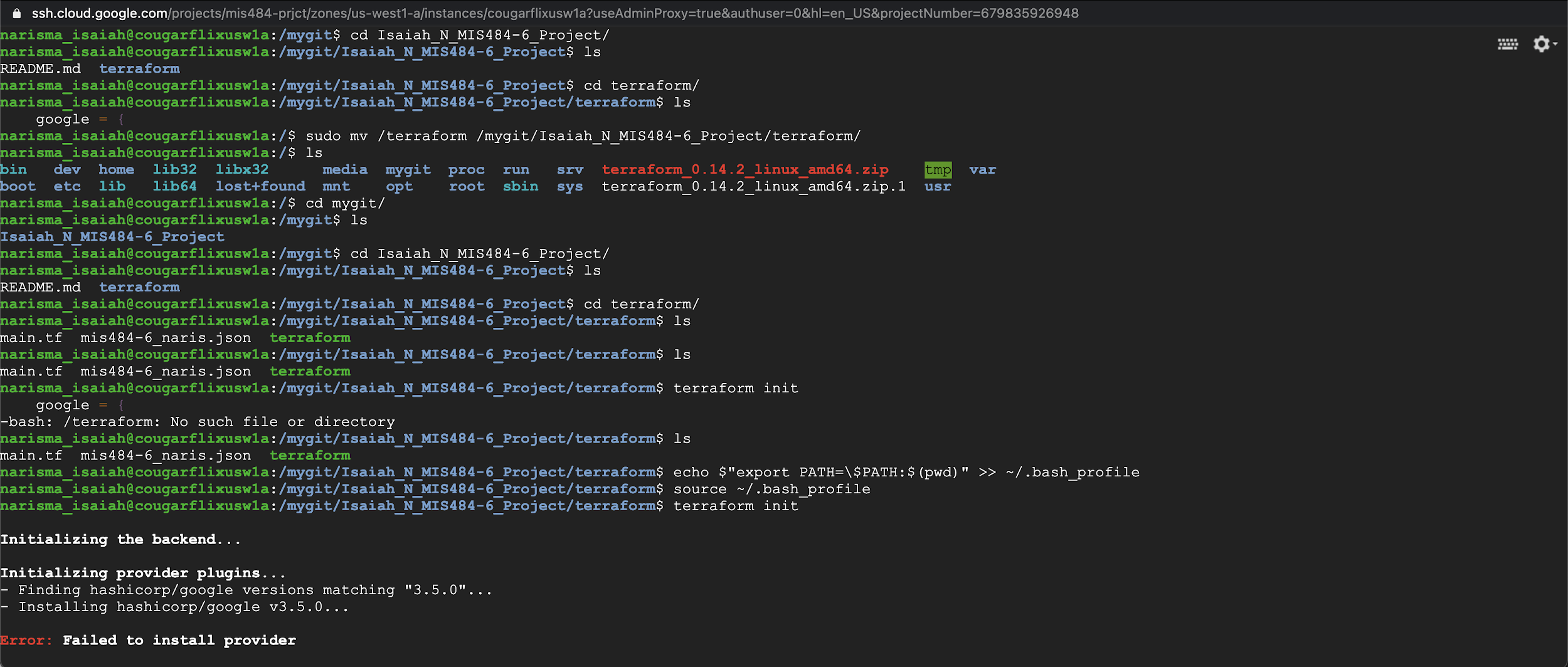
So when I heard that I needed to switch to GCP from AWS I was initially shocked at first because I thought that we would be able to do this project in AWS. I had already put so much time into AWS at that point, and switching to GCP would have been cutting it close. But then I realize that I already had the resources in might GitHub repository that I would just be able to perform a `git clone`.



The next hurdle was essentially just figuring out how to download everything on GCP because in AWS the syntax for downloading certain package managers were a little bit different, likeI did not realize that GCP instances do not already come with certain packages such as wget or unzip, so I had to do a little bit of research to figure out how to download wget using apt and unzip using the same package manager.

Another issue that I’m not sure if it was much of an issue rather than an inconvenience, whenever I would have to download GitHub or install git, the git installation would take an absurd amount of time. I would time it and sometimes it would take upwards of 10 to 15 minutes to install. Also, a lot of students came to me with transferring SSH keys to the VM problem that they were having earlier on in the semester, where they would not be able to login SSH into their instance and it will just take a really long time and would just perpetually lag. That is currently a problem I am experiencing myself with all of my instances after the project is complete.

So the next biggest problem I had in this section was performing terraform init command I would just get error after error, after error, where it would say initializing the back end, initializing provider plug-ins, and begin installing, however, it would eventually fail in an error saying citing ‘failed to install provider’ and it was just incredibly frustrating and I had that problem probably five times I had it in AWS and I had it 3 to 4 times in GCP and all different instances.



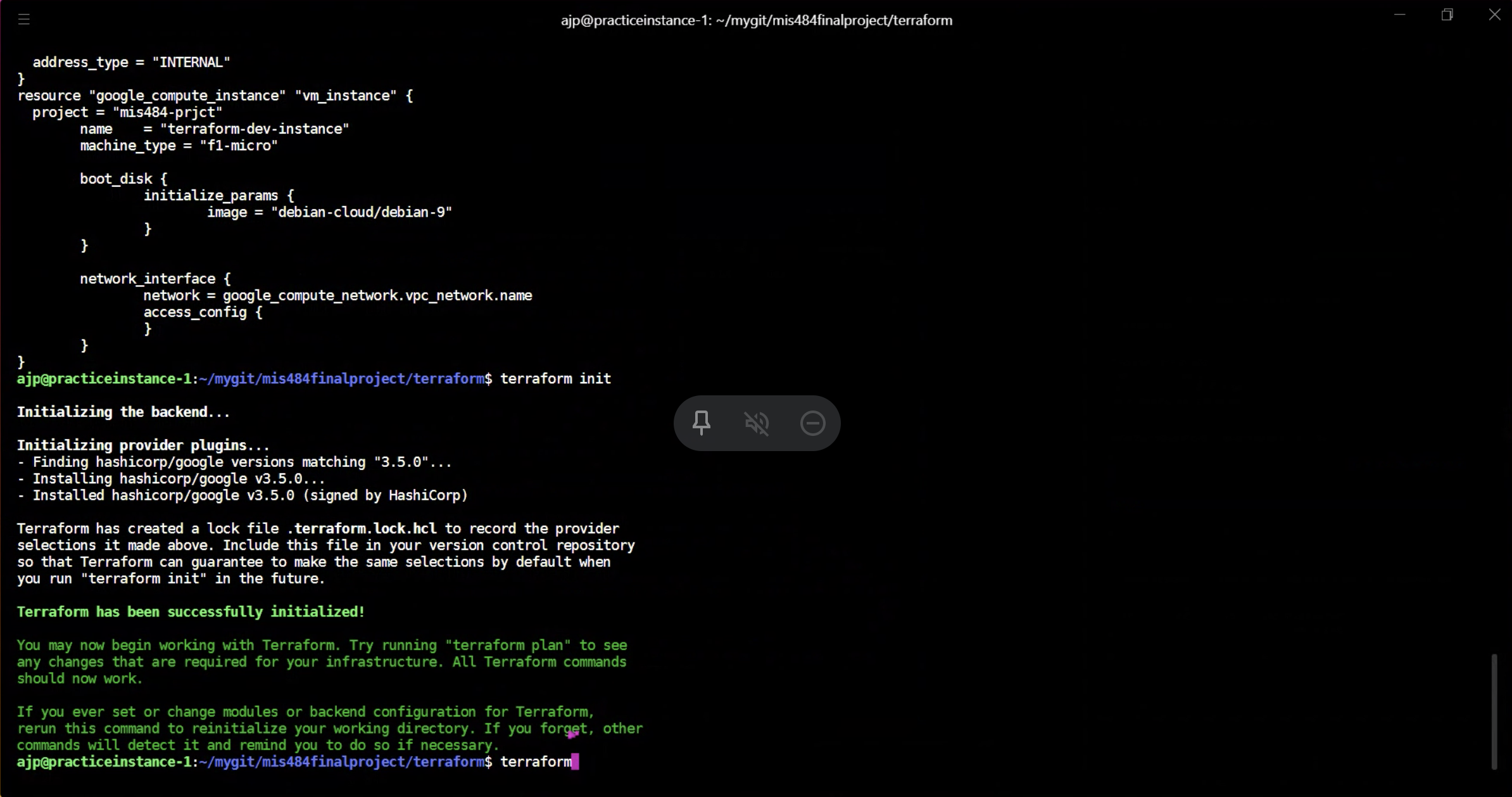
So I contacted a few of my teammates and I reached out to Michelle, which she told me that I should move my terraform inside my terraform directory in my gate, and I did that however the problem still remained. I also reached out to Alan and he recommended that I copy all of my dad Jason on file code into the doggie ignore file and commit and re-run it, I did that and still the same problem Allen furthermore also suggested that I make some edits to my ~/.profile to try to fix it however no luck.

WORKAROUND:

So the eventual workaround that we had to come up with after hours of troubleshooting and researching; was that Allen would share his screen and I would essentially use his instance to provision an instance in my GCP using my resources from my GitHub Repo.

Allen was kind enough to stick with me later that night and help me out with this. We were screen-sharing his screen on Google hangouts while I told him what to do, and the first thing that we did was connect my GitHub repo to his instance, using the same procedures as if I would connect to an instance. From there, we were able to perform a `git clone` and take all of my resources.

Allen and I proceeded to check all of our /.profiles and make sure the pathing was correct. We also moved my `terraform` into my `terraform` directory within my repo directory and made it work.



All in all this section of the project was incredibly frustrating; however, we made it work in the end