5.2) Installation Script Failure

For our start script, we chose to use the format provided on Piazza by having the two; launch.sh and install-my-app.sh scripts. Since the main idea of making a robust start-script is “making sure the steps in your installation script can be run more than once” we have analyzed and “hardened” our start-script to support recovery from an instance crashing during the installation script. We can analyze the “hardness” of our start script by the following points:

1. Installing tomcat8 once again does no harm
2. Copying from s3 storage does no harm since we over-write the old file
3. Wget creates folders, but if a file with the same name already exists it does not over-write it rather creates a file with name.(1), name.(2) etc. So we need to make sure that no such file name exists in our system (as the one that wget will create), thus to play it safe we remove all the directories that could already exist before calling wget for IP, DNS and AMI-value. As our script might have crashed (due to a power outage so forth. After having created files, so there might be conflicts). We are aware that an extra file with name.(1) extension would do no harm to the system; however we don’t want to pollute our environment with files we don’t need.
4. In our simpledb code, we have set the Replace=true flag to true, so that even if a some values were corrupt we can just overwrite them, making sure that the simpledb is stable.
5. Since our system needs to register all the other instances by reading their values from the db, our server waits in a while loop while the number of attributes in the db is not equal to number of instances multiplied by number of attributes per instance in the simpledb.. Thus all our servers make sure that the “whole” system is stable before starting, they idle while others have yet to register to simpledb properly.
6. When we query our database to collect all the metadata for the other instances we write (re-direct) to a file. Since we are not appending to a file but rather over-writing it; this should also work.

Although it is mentioned in the spec that it could be of our interest to use the installation script for reboot with a simple “sudo su” hack(as we need to run with root privileges when we ssh in), we chose to write an alternative reboot.sh script since the reboot task also has the job of incrementing the reboot count, while the start script simply has the task of assigning 0 to the reboot count for the particular instance. Our reboot script makes the assumption that our instances have started using our start script.