Raiya Haque

2.3 Reference Monitor Write Up

The vulnerabilities present in my reference monitor were mainly in the writeat function, specifically when attacks were trying to write past the end of the file. Several attack cases where a writeat would be called, undone, and then another writeat would be called at an offset past the end of file were able to break through my reference monitor. I did not think about this possibility while originally building my reference monitor as I thought that checking for a negative offset and file length would be sufficient to deal with any abnormal behavior. I fixed this implementation by adding a few more variables to help me keep track of the previous data and previous length of the most recently called writeat. At the beginning of every writeat, I would check whether the previous call was an undo() call by determining whether the pending data and pending offset were None, and I would check whether anything was actually undone by seeing whether my previous data was not None. I would update the length to subtract the previous data, so that essentially undid the previous writeat. To handle the EOF error specifically, I would raise the error only if the offset was greater than the length of the file. The length of the file was always updated accordingly, so the easiest way to deal with EOF was to check whether the offset exceeded the file's length. Updating the length properly was also something I didn't handle properly in my original reference monitor, and I fixed this by adding the length of the new data to the file length plus offset if that was greater than the current file length. I also did not handle the case of not creating a file if create was set to False and the file did not exist already. I added this implementation in the init function to handle that scenario.