Andrew DiBiasio

Amanda Girard

Computing, Society, and Professionalism

30 August 2023

Homework 1: Therac and Boeing

In both the Lion Air Flight 610 of October 29, 2018, and the Ethiopian Airlines Flight 302 of March 10, 2019, a Boeing 737-MAX aircraft crashed due to technical malfunctions with the software behind the Maneuvering Characteristics Augmentation System (MCAS). The MCAS was intended to address uneven weight distributions on the aircraft, stemming from their engine placements [2]. As such, they would cause the aircraft to pitch down and ultimately mitigate the aerodynamics issue if the AoA reading, a measure of aerodynamic imbalance, became too high. This system would read the current AoA from one of two sensors, which many considered a design flaw, as it only collected data from a single sensor. In Lion Air Flight 610, a malfunction in the AoA sensor caused the MCAS to point the nose of the aircraft down, causing the plane to plummet and eventually crash [2]. A very similar issue happened with the Ethiopian Airlines Flight 302 when a series of commands from the MCAS caused the aircraft to drive the nose downwards, once again causing the aircraft to plummet [1].

One causal factor from section 4 of Nancy Leveson’s article “Medical Devices: The Therac-25” that is applicable in these scenarios is Overconfidence in Software. It is common for engineering teams to simply assume the software they are given works as-is. Due to this, many people end up overlooking potential errors. In the case of the Therac-25 medical device, the safety analysis did not include any software checks [3]. Similarly, in the case of the Boeing 737-MAX, there was a large amount of overconfidence placed in the reliability of the MCAS software. They wrongly assumed that the software would be capable of handling any extreme scenarios or malfunctions of external parts, such as the AoA sensors. Furthermore, because of this overconfidence, pilots were not trained in the proper procedures to follow in the instance of a software malfunction. This is what ultimately led to the crash of these aircrafts, as when a software malfunction did occur, the pilots were unable to interfere with the MCAS, and the plane continued to plummet until it crashed.

Another Leveson causal factor is the Inadequate use of Software Engineering Practices. As was later found, the MCAS software was only designed to trigger the nose-down command after reading data from only one sensor [2]. However, this is a major design flaw, as these types of commands should not be susceptible to a single point of failure. This makes it clear that Boeing did not put enough consideration into the necessary failure prevention measures and did not conduct a comprehensive testing of the system to identify potential flaws of the MCAS software. Similar inadequate software engineering practices were also found in the Therac-25 incident. During this issue, the Therac-25 lacked the necessary safety checks on the software component of the device. While these share similarities in malpractice of their software, the Boeing 737-MAX is slightly more complicated, as the testing of the MCAS involves both software and hardware. So, this added complexity is a possible reason as to why it was more difficult to test the MCAS software compared to the Therac-25 device.

The final but one of the most prominent Leveson causal factors is the Inadequate Investigation or Follow-up on Accident Reports. Clearly, one can see that Boeing did not perform an adequate amount of testing on the malfunction of the MCAS system following Lion Air Flight 610. Despite taking place nearly half a year prior to the Ethiopian Airlines Flight 302, the MCAS once again malfunctioned due to a faulty AoA sensor [1]. After the initial crash, there were clear indications of bugs within the MCAS system. However, there evidently was a lack of investigation and follow-up on the root cause, as the same issues caused another flight of passengers to lose their lives. A similar incident happened with the Therac-25 medical device. For instance, the Therac-25 radiation therapy machine had an outstanding history of providing overdoses to its patients, attributed to software-related issues [3]. However, over this multi-year long span, the issues within the Therac-25 continued to overdose its patients, causing harm to many. Ultimately, the lack of effective investigation and follow-up on the accidents reported on the Therac-25 led to the same issues occurring again over several years.

Ultimately, the Boeing 737-MAX and Therac-25 malfunction incidents both feature key causal factors outlined in Nancy Leveson’s analysis of the Therac-25 case. Due to their development team’s overconfidence in software, inadequate use of software engineering practices, and inadequate investigation or follow-up on accident reports, these technologies experienced major software malfunctions that caused harm to their intended audiences. The tendency for people to place too much confidence in software is not unique to a particular field, so when both Boeing and the developers of the Therac-25 exhibited an overreliance on software, they exposed themselves to potential software malfunctions. Furthermore, by failing to conduct the necessary software engineering practices, as seen with Boeing’s lack of rigorous testing of their MCAS and the lack of safety checks for the software being the Therac-25, these teams allowed for the possibility of errors. Lastly, by failing to properly investigate and follow-up on reported accidents, they also allowed for life-threatening malfunctions to persist and continue to harm others.

Works Referenced

[1] “Ethiopian Airlines Flight 302,” Wikipedia, https://en.wikipedia.org/wiki/Ethiopian\_Airlines\_Flight\_302#:~:text=Repetitive%20and%20uncommanded%20airplane%2Dnose,probable%20cause%20of%20the%20accident. (accessed Aug. 21, 2023).

[2] “Lion Air Flight 610,” Wikipedia, https://en.wikipedia.org/wiki/Lion\_Air\_Flight\_610#:~:text=A%20malfunction%20in%20the%20captain,causing%20the%20plane%20to%20crash. (accessed Aug. 21, 2023).

[3] N. Leveson, Medical Devices: The Therac-25, https://www.cs.ucf.edu/~dcm/Teaching/COP4600-Fall2010/Literature/Therac25-Leveson.pdf (accessed Aug. 21, 2023).