



Rookout Transforms Flytrex's Ability To Debug In Production



Rookout is most effective in locating, pinpointing, and understanding bugs that happen in production.



FLYTREX



Using Rookout allows us to still collect information from a live production environment as we want it and without deploying any changes, which is a major force multiplier for us when investigating issues in regulated environments.





FLYTREX

Rookout Transforms Flytrex's Ability To Debug In Production

Debugging in production drastically reduces the time spent debugging.

The Flytrex Debugging Challenge

Debugging in production, without the right tools, can be situations made out of the most well-crafted nightmares. That's exactly what Flytrex found out when they came across a bug within one of their drones - while it was in flight.

Flytrex faces a few core challenges when it comes to debugging their aircrafts. To begin with, their technology is extremely complex. Second, they need to abide by stringent FAA regulations. And third, when bugs happen in production, it means that they're dealing with a bug that can be found in an airborne system.

Their drones are complex entities that include software, hardware, embedded code, and operations. When previously attempting to debug them, Flytrex had found themselves trying a myriad of methods to debug their drones. This included reproducing their bugs in local or staging environments, reproducing changes that had logging, or redeploying changes to production with the aim to collect more information about the bug that needed to be fixed. All of these methods took a lot of time and, in most cases, proved to be nearly impossible. This was because a good amount of Flytrex's issues can only be resolved in the field, as that is where the set of relevant parameters happen. Therefore, reproducing these bugs locally wasn't the right solution.



Rookout is most effective in locating, pinpointing, and understanding bugs that happen in production.



Flytrex also encountered a unique challenge in being regulated by the FAA. FAA regulations stipulate that companies are unable to deploy to production without approval, as the production system is regulated by the FAA; as such, changing to a version that hasn't yet been tested, verified, and certified by the FAA is not allowed. Therefore, a search for a solution had to be done that would allow them to find the source of the bug quickly, assess the situation, and be able to rectify it or inform the FAA about it as quickly as possible.



FLYTREX

Rookout In-Flight: An example

There's no better way to illustrate how a live debugging tool works than when your code is running. Or in Flytrex's case, when a drone is in flight.

During an internal test, with their drone up in the air, an issue with session counting was found. This issue took place due to a surprising race condition, which is very difficult to catch. The bug itself wasn't even in the code, but rather in a library that was being employed. Rookout was used to pinpoint the bug and, while the drone was in flight and their people were in the field, were able to provide a workaround and ultimately a solution.

"Rookout gave us complete confidence in being able to handle any bugs we were faced with, especially in prod. This confidence was due to our ability to gain a full understanding of the bug we were encountering. We therefore knew how to work around it, as we understood the issue completely, and can conclude whether it was safe or not safe to fly in that particular situation, even if the actual fix came a few days later. We were able to quickly understand the source of the bug, thanks to Rookout", said Imri.

"When in a test, while our people are in the field, having a certain component or scenario not working is quite uncomfortable, to say the least. Rookout gave us the confidence that we needed in knowing that we understand what's happening in our systems, that it's something we can and will fix. The certainty that we know how to solve whatever issue we're facing helps us both internally, and with the FAA".



Using Rookout allows us to still collect information from a live production environment as we want it and without deploying any changes, which is a major force multiplier for us when investigating issues in regulated environments.



Results: How Rookout Helped

Initially, Flytrex believed that Rookout would be used predominantly to debug their code early in the software development lifecycle. They came to realize that it helped them do much more than that, though. "What we quickly realized was that Rookout is most effective in locating, pinpointing, and understanding bugs that happen in production", said Imri Goldberg, VP R&D at Flytrex. "For instance, if we find that one of our delivery drones has a bug, mid-delivery, and we want to immediately find out what's happening, we can."

"Personally, Rookout has saved me hours of work", said Imri. "The R&D team has been able to figure out what the problem is very quickly. This has been critical because 'when you have a failure or issue in the field, and you need to understand what the issue is and what bug it is, you can't be relaxed and wait for the answer to appear. Rookout has become our workaround solution. Instead of a full day of research, it takes a couple of hours or less."



FLYTREX

Using Rookout has saved the Flytrex team both lots of downtime and a significant amount of the time in which we had previously been blocking flights because of issues. Now, instead of wasting hours or even days of R&D time, we are able to deliver much earlier", says Imri.

With the adoption of Rookout, Flytrex has been able to go above and beyond their capabilities than they had before in two significant ways. The first is with the ability to collect information from production, live, without having to deploy new code or make code changes. They are able to do so on the fly as they see the bug happen and collect more information as they want it. As Imri said, "Using Rookout allows us to still collect info from a live production environment as we want it and without deploying any changes, which is a major force multiplier for us when investigating issues in regulated environments.

The second way is with the ability to pull information from their running code into systems such as logz.io, without having to deploy any changes. This gives them the ability to look at the information at their own leisure, without any stress or time constraints.



Now, instead of wasting hours or even days of R&D time, we are able to deliver much earlier.



A Few Words About Flytrex

Flytrex is a real-world drone delivery service that is fully autonomous, regulatory approved, and insured. They are part of the Federal Aviation Administration UAS Integration Pilot Program (FAA IPP) and have partnered with the North Carolina Department of Transportation to standardize backyard drone delivery.

Today there's a lot of demand for drone deliveries, especially because of COVID 19. People are looking for less human contact when doing deliveries and it's a faster and more efficient way.

Their Stack

Flytrex's solution is more than just a flying drone, it's actually considered an aircraft. Weighing at around 15kg, it flies over people's heads at a height of approximately 70 meters, and therefore has a significant focus on safety. Due to it being labeled as an aircraft, Flytrex has to be regulated by the FAA. The technology of their aircraft is made up of hardware, software, embedded code, and operations.



©2021 Rookout