



Rookout transforms how Backblaze engineers develop



Using Rookout lets us just look under the hood and see what's going on. It saves us a lot of time.



BACKBLAZE



Putting debugging statements at various areas in the code let me see just where that code was hit, so I could find the real issue much more quickly.





Rookout transforms how Backblaze engineers develop

Accelerating identification of root causes makes solving code issues faster and easier

The Backblaze debugging challenge

Backblaze has over 2,000 instances in its production environment. Strict limitations on the frequency and timing for pushing new code -- including adding log-lines -- meant that pinning down root causes of issues was a lengthy and distracting process for the company's engineers.

Reproducing issues in their staging environment was often hit-or-miss and in many cases, they didn't have the log information they needed. So they had to add log lines to their code and wait, sometimes for days, until the log code could be pushed to production.

Backblaze was seeking ways to make the state of its system more visible, so they could avoid the distractions and delays involved in getting log info.



Using Rookout lets us just look under the hood and see what's going on. It saves us a lot of time.



Rookout helps Backblaze engineers quickly and efficiently gather information about exactly what's going on in their dev and integration servers so they can find out what's causing bugs without hassle or delays.

Getting started with Rookout

Deploying Rookout across Backblaze's 2,000+ instances in staging and production was fast, friction-free and had negligible performance impact. Once the company's engineers clicked-in to the Rookout concept, they were quickly able to adapt to the UI and master its use.

Rookout created a new way for Backblaze engineers to flush out issues in staging, before they get to production. Now, with Rookout, they can easily add ad-hoc variable requests to their staging code, play out scenarios that they are interested in investigating, and immediately see how the code is behaving as each scenario unfolds.

As Adam Feder, Principal Engineer, Backblaze, explained:

"Before, when debugging, a developer would have to compile, build, push -- you know the drill. And all that took quite a while even in our staging environment, especially since we share the environment with QA and therefore try not to bounce it constantly".



The results

The Backblaze dev team uses Rookout to expedite debugging issues in both staging and production, for ongoing data collection and troubleshooting as well as for edge cases. Here's what Marjorie Ready, Senior SW Engineer at Backblaze, had to say about debugging with Rookout:

"When I debug, it's mostly in the staging environment. And it's been fairly laborious because in the past, we had to put in log lines, which had to be approved, then merged into master, and only then pushed out to the staging server. So it could take at least an hour, if not more, to get things in place to do even preliminary debugging. And then if it didn't quite hit the thing you were trying to hit, you'd have to add, rinse and repeat, often multiple times. Typically, what I found was that the problem that I thought that I had was not the problem that I really had."

"Putting debugging statements at various areas in the code let me see just where that code was hit, so I could find the real issue much more quickly and that has made Rookout especially useful and helpful!"

"I can envision cases in which it might have taken me several hours of adding log lines and pushing to staging to find errors that I found in minutes with Rookout. I was also pleasantly impressed with the responsiveness of the Rookout team to our questions and requests. Their level of support exceeded our expectations."

The bottom line

Rookout enabled Backblaze to quickly identify and pinpoint issues in both staging and production, significantly reducing both debugging time and dev frustration. In fact, instead of using Rookout a couple of times each week, as expected, Backblaze engineers are finding that they can get much more done by turning to Rookout several times daily.



Putting debugging statements at various areas in the code let me see just where that code was hit, so I could find the real issue much more quickly.



A few words about Backblaze

With almost an exabyte of customer data under management, Backblaze is a leading provider of cloud storage. They offer two primary products:

- **B2 Cloud Storage:**
An infrastructure-as-a-service (IaaS) storage solution that is similar to Amazon S3, but offered at ¼ of the price.
- **Computer Backup:**
The industry's leading backup solution for computers & laptops.