



# Rookout helps Behalf increase dev velocity

Adding non-breaking breakpoints on the fly drastically cuts down troubleshooting-related dead time

## The Behalf debugging challenge

Behalf develops microservices in Java and Python. At each stage, from development through staging, production, and rollout, it faces distinct debugging challenges:

**In development**, Behalf uses a hybrid debugging approach, which works well for individual microservices and even for end-to-end flows. The challenge arises when tracing a local microservice issue into components with which it may interact.

*"You trigger an event, and nothing happens. You trigger an API call and don't get the expected results. You pull in the developer of the other microservice and now two people are spending time and effort to dive into logs, spin up a new cloud environment, and check how both microservices behave."*



*Rookout has empowered my team to keep up the pace at which they develop, at every stage.*



**Production and staging** pose even thornier debugging challenges: Reproducing issues in a local environment can take Behalf developers hours and require pulling other developers away from their individual tasks. Adding a log line to the production or staging environment involves going through a release cycle. Even attaching to a remote process on a dev environment means more wait-time, context switching, and changing configurations. And despite all these efforts, chances of reproducing the bug are low since latency and other conditions are different from production or staging.

Finally, there are the challenges involved in **tracking new features** to quickly surface any issues they cause. No matter how systematic and thorough Behalf devs are when choosing what should be measured, unexpected issues almost always arise. Measuring data relevant for those unexpected issues means adding more log lines, which in turn means waiting for the next release or worrying about a hotfix.

## Getting started with Rookout

*"Throughout the deployment process and beyond, the Rookout team has been extremely responsive and cooperative. The training we got was excellent, and the fact that we had a slack channel where Behalf developers could directly communicate with the Rookout team was just fantastic."*



The main focus of the Behalf dev team - besides quality, of course - is to operate at high velocity. Every interruption slows the dev team down, and the longer it takes to handle the interruption, the more the team's velocity falls. Behalf was seeking a way to reduce idle time during troubleshooting.

Rookout enabled the Behalf dev team to quickly get the data they need by adding non-breaking breakpoints on the fly, in both production and staging environments. As Nadav Sharir, VP Engineering at Behalf put it:

*"Rookout has empowered my team to keep up the pace at which they develop, at every stage."*

Behalf keeps extensive logs and searching through them remains an important method for troubleshooting.

*"With Rookout, we can add log lines and stream them to Sumo Logic without a hotfix - a capability that will make our logs even richer and more valuable going forward."*

## Results

During development, Rookout cut the time it took to debug interacting microservices from an average of 1.5-2 hours for each developer involved to just a few minutes.

Similarly, in production, with a click of a button Rookout enabled Behalf developers to see all the data they need to resolve issues - with no context switching, worrying about hotfixes, or getting other developers involved. What is especially valuable is that now there is no need to wait for the bug to reproduce, or worry that it will not.

During feature rollout, Rookout allowed developers to fetch additional data points without waiting for a release. Since they could play with the data, they could quickly determine what they want to measure and how to format it. This enabled the developers to add the final measurement code in the next release, without a protracted trial and error process.

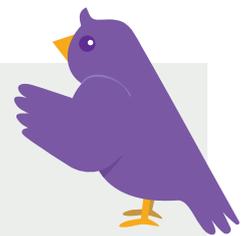
## The bottom line

*"Rookout gives me and my team the peace of mind to release more frequently, as I know that if there is a problem we will quickly fetch the data we need to troubleshoot it. It helps our developers maintain velocity while staying effective and happy."*

Nadav Sharir, VP Engineering



*During development, Rookout cut the time it took to debug interacting microservices from an average of 1.5-2 hours for each developer involved to just a few minutes.*



### A few words about Behalf

Behalf is a digital platform that facilitates payments by extending net terms and financing to businesses.

Behalf empowers commerce for large and small businesses with on-demand cash flow solutions. It helps businesses where they need it the most by increasing their purchase capacity for the things they need to grow.