

Meet Lightrun

The Leader in Developer Observability

Lightrun is a revolutionary **Developer Observability Platform** allowing developers to **dynamically instrument** logs, traces, and metrics to debug live applications **at runtime** straight from the IDE.

- ✓ **No redeployments**
- ✓ **No restarts**
- ✓ **No code changes**
- ✓ **No need to reproduce issues in local/test env.**



TRUSTED AND LOVED BY DEVELOPERS GLOBALLY



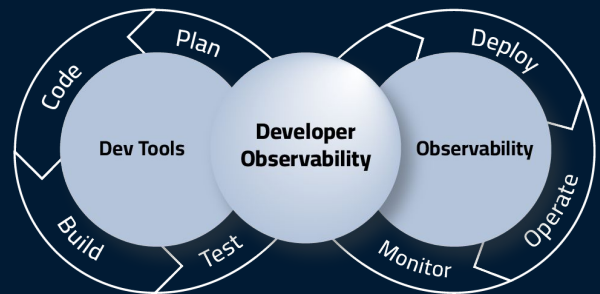
BACKED BY



The Solution

Shift Left Observability

Ops-free, Real-time, Dev-native Observability



- Quality
- Time to Market
- Productivity & Velocity
- Dev Experience
- MTTR
- Costs Optimization

Outcomes



Revenue at Risk Protection



Incident Management Process Efficiency



P1s & Incident Volume



Developer Productivity and Debugging Time



Observability Costs

Use Cases

Troubleshoot Production Incidents in minutes

Debug performance bottlenecks, failed app transactions, misbehaving caches /APIs, DB persistence issues and more

Reduce Logging costs by up to 40%

Dynamically add logs as and when you need them, slashing logging volumes and costs across the board.

Debug Cloud-Native Apps From the IDE

Avoid cascading failures in live applications while adding real-time logs, capture breakpoint-grade telemetry and instrument metrics in multiple pods or multiple clusters simultaneously - no service mesh or port-forwarding magic required.

Debug Code-level User Specific Issues

Explore specific user issues and track specific user execution flows in real-time without interfering the user experience.

Runtime CVE Prioritization - Eliminate CVE false positives

Reduce false positives, effectively prioritize runtime vulnerabilities, and improve the speed and security of development processes

Validate Progressive Delivery Rollouts

Know which code block is executed for each user. Easily & conditionally isolate specific execution paths - no unnecessary logs left behind.

Lightrun Architecture

A The developer's experience is of communicating directly with the running code, though there is no direct connection between developer and deployed software.

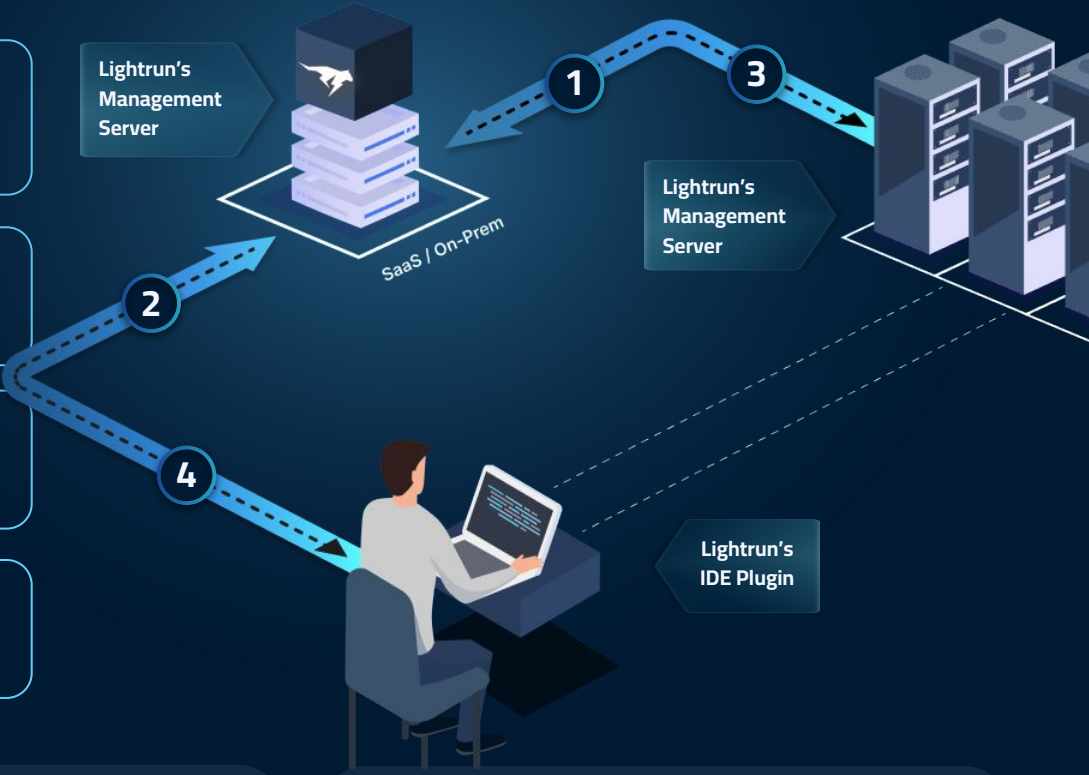
B The developer iteratively inspects the running code, while all communication is relayed safely via the Lightrun Server

1 Each service being monitored includes a Lightrun Agent. The Agent polls the Lightrun Server for requests via secure websocket

2 Developers use an IDE plug-in to request information about the running code

3 The developer's request is relayed along the channel opened by the Agent

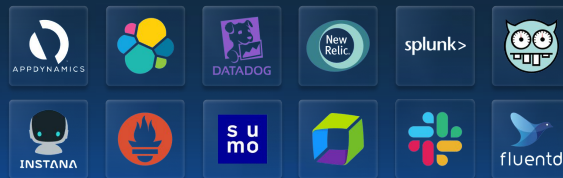
4 The requested information, received from the running Agent, is returned to the developer's IDE, or the application's stdout.



Lightrun is exposed to your developers as a native, familiar IDE plugin



Lightrun information can be piped anywhere your IDE, various integrations or local files



Environment Agnostic

Lightrun operates everywhere and anywhere: on-premise, in the cloud (**AWS, GCP, Azure**), for microservices, for serverless, K8s, and more. Debug in any environment across any infrastructure.



Security and Privacy

Lightrun assures organizations the security and privacy of their code by being ISO-27001, SOC 2, GDPR and HIPAA-compliant. In addition, Lightrun provides enterprise-grade controls out of the box: encryption, authentication, RBAC, SSO, audit trail and privacy blacklisting.

Source Code Incompatibility

Lightrun eliminates source code incompatibility by comparing file signatures between source and runtime.



Stability and Minimal Footprint

Lightrun's footprint is negligible. The invocation requires a minimal footprint ranging between 10s to 100s of microseconds. To ensure overhead control, we use quotas to impose usage limits. A built-in sandbox prevents state modifications.

Try our playground: playground.lightrun.com