How Komodor Helped Lacework Reduce MTTR by 70% and Save Christmas

LACEWORK

Company Size: 501 - 1,000 employees

Industry: Cloud Security

Komodor Installation:
30+ clusters (4 prod) /
3,700+ AWS-hosted services

About Lacework

Lacework is one of the world's leading cloud security providers, offering automated threat detection and deep visibility solutions for AWS, Azure, GCP, and Kubernetes environments. Disruptors and global industry leaders like Snowflake, VMware, and Drift depend on Lacework to drive revenue, improve business velocity and consolidate point security solutions into a single platform.

The company is led by Jay Parikh (ex-Facebook and Atlassian) and David Hatfield (ex-Symantec and Gartner) and is widely considered to be at the forefront of innovation, tech, and dev culture.

The Challenge

Lacework is in a hyper-growth stage, fueled - amongst other things - by an accelerated adoption of Kubernetes. To support the growth of its business the company's development team of several hundred was running a highly complex k8s environment with dozens of clusters and growing.

Despite having a highly efficient architectural setup for deployment and best-of-breed stack, including tools like Scalyr, Grafana, ArgoCD, and PagerDuty, Lacework's teams were struggling with some acute inefficiencies when it came to incident management.

For instance, the SRE and on-call teams lacked visibility of manual deployments and configmap changes, hindering their ability to quickly identify root causes, based on a complete picture.

With **Komodor** Lacework was able to:

Cut down MTTR by ~70%
with newfound visibility of node issues.

Drive X4 increase
of dev participation in on-call rotation

Reduce OOM errors by 90%
in production

Save 1 Xmas dinner

Also, using multiple tools actually created delays, requiring responders to scour through heaps of irrelevant data every time they were investigating a problem. This created confusion and unnecessary delays, impacting the responders' ability to quickly resolve issues with confidence.

What this added up to was that the average MTTR for incidents, and the escalation rates, were above what the company was willing to accept. To improve on the situation Lacework needed a dedicated tool that would provide visibility of all Kubernetes resources, streamline investigation and improve dev experience and participation, allowing it to double down on its commitment to shift-left troubleshooting.

Landon Orr
Staff Site Reliability Engineer at Lacework

"I showed Komodor to our build release teams, who manage our non-prod clusters and they were in awe. Hands down, this is the best troubleshooting tool we have. It pinpoints exactly what changed when it changed and who changed it. We have teams using it so much now, that it's starting to become part of their daily processes."

The Solution

Komodor platform has helped Lacework significantly reduce MTTR and improve overall troubleshooting efficiency with the combination of the following:

1. **E2E visibility:** Komodor provided a multi-cluster view of all k8s events and resources, including manual deployments and configmap changes. Distilling these into a single timeline view created a simple chronological story of all system changes, useful for expert and non-expert troubleshooters alike.

2. **Contextual insights:** Seeing all of the events across the system, in their chronological order, made it easy to know 'who changed what and when', gain the right context, and correlate between an issue and its root cause. This helped reduce investigation times, driving down MTTR for nearly all incidents.

3. **Improved efficiency:** Acting as a single source of truth (SSoT) Komodor allowed responders to conduct the entire troubleshooting process with just one tool, in the comfort of a simple and user-friendly UI.

In addition, Komodor offered easy secured access to production resources, no longer forcing responders to struggle with VPN and RBAC configurations to remotely pull logs. This helped boost response times, further cutting down MTTR.

4. **Opinionated monitoring:** Komodor's automation features provided digested root cause analysis along with easy-to-follow remediation suggestions, empowering more devs to participate in the troubleshooting process.

   As a result, the on-call pool expanded four times over (from under 10 to 40+), no longer putting the majority of the load on the shoulders of the US-based SRE team.

5. **Pinpointing systemic issues:** As more and more developers across the organization started using Komodor, they were able to expand their understanding of k8s troubleshooting and identify deeper issues.

   One of those issues was a problem with a dev cluster that kept failing due to inadequate memory resource limits. Fixing this single issue helped the team cut down the number of production out-of-memory (OOM) events by 90+%.

6. **Saving Christmas:** Komodor's platform was 'fire tested' during the holiday season, when it allowed the SRE team to solve an outage on Christmas eve in under 10 minutes, without even leaving the dinner table.

   After getting a notification, the Senior SRE on-call simply pulled out his phone, logged into Komodor, looked up changes across all clusters, instantly saw the issue, and proceeded to revert the change that was causing a problem in 30 pipelines.