Cloud | Containerization

Case Study

gslab

Modernizing an application delivery platform using containerization

An app delivery platform reduces deployment time and improves scalability through modernization & cloud migration

Executive Summary

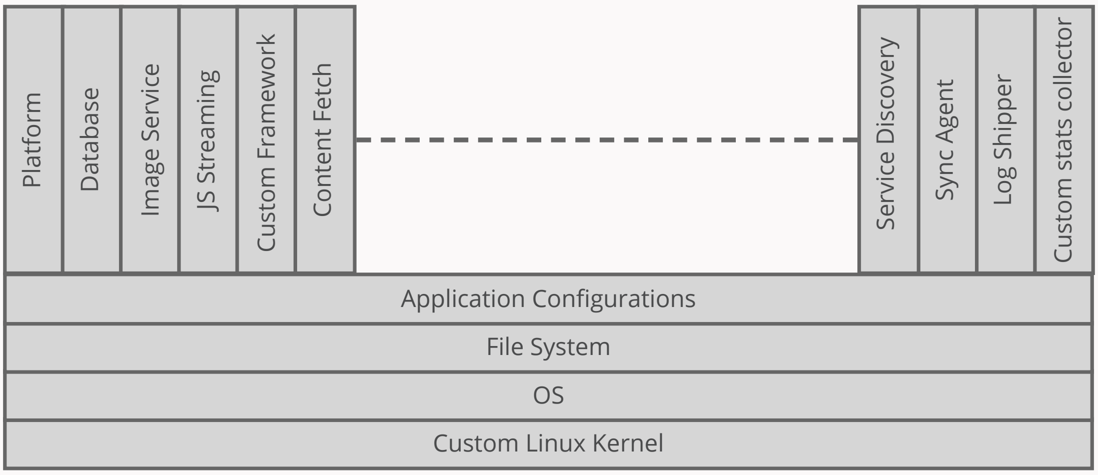
Today, consumer expectations & the technology landscape are both changing at a rapid pace. Some of the products based on older architectures lose out on benefits such as speed, cost savings and scalability offered by the latest innovations. The issue can grow to become such a big problem that it starts impacting product reach, adoption, and usage. To avoid these pitfalls, our client decided to modernize their monolithic app delivery & optimization platform. We built a completely containerized & scalable solution which is future proof for the time to come.

Overview

When a web application is served, the faster it loads and the more personalization it offers, the better the experience turns out for the end user. Our client ran a platform leveraged by several web applications. It used artificial intelligence & machine learning to analyze the behavior of those applications and their consumers to optimize HTML, JavaScript, images, video and other application components. The primary goal of the platform was to improve application performance, consumer experience and security. The core product was built for bare metal infrastructure and was monolithic in nature. Each component was installed one after the other on a single operating system. The product was mostly installed on custom built public clouds. Each of these deployments was customized to geographic requirements across the globe. Deployment, management and maintenance was too expensive for this infrastructure. There was a need to modernize the application to improve the speed of deployment. This would enable the end customer to quickly get up and running.

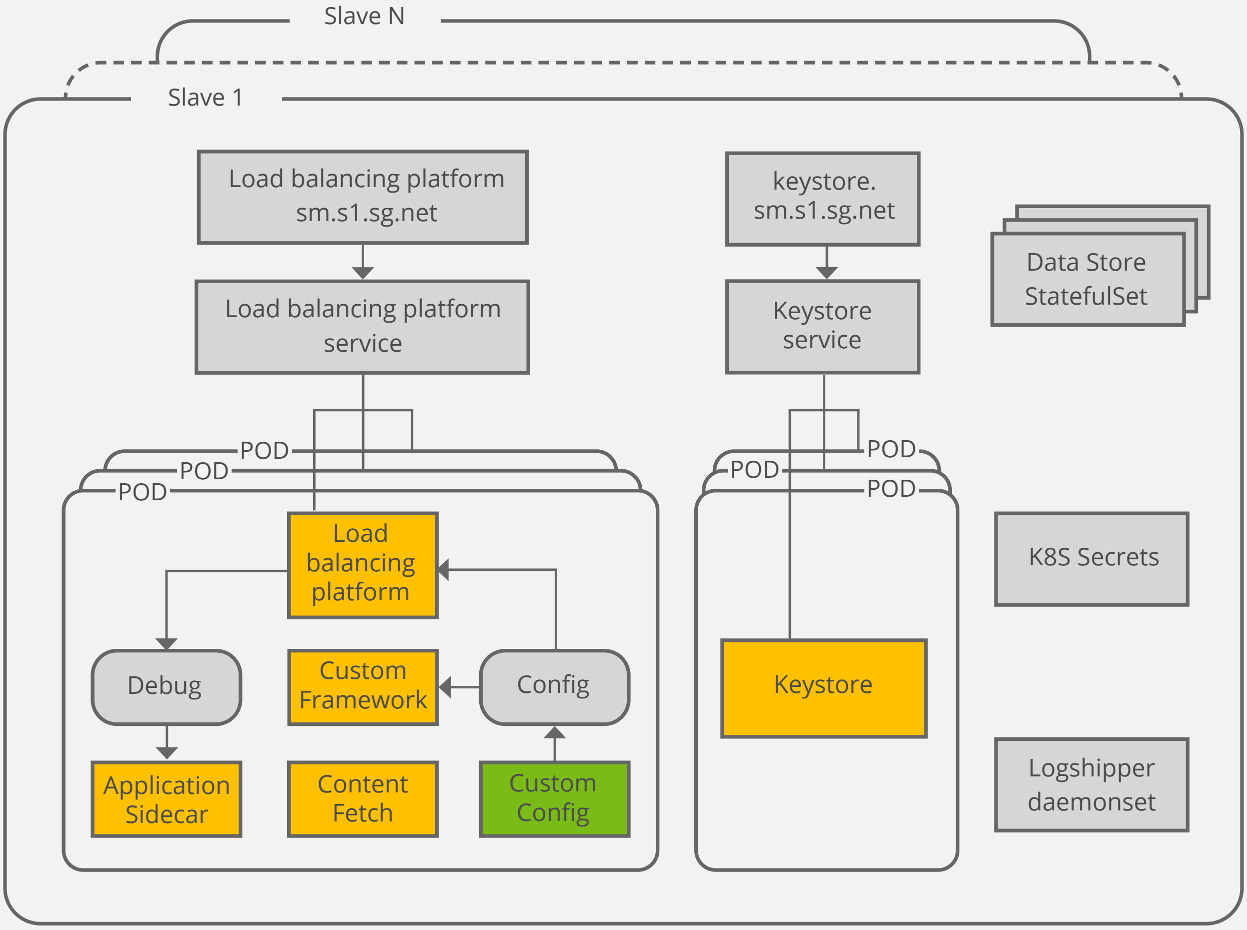
Challenge

The original monolithic product had multiple components. Our client wanted to improve the speed of deployment while remaining cloud agnostic. We had to create an architecture that could work on both bare metal as well as cloud infrastructures. Monitoring capabilities for each component and the ability to get service stats were also required.



Services running as processes on the same physical machine

Solution



Solution architecture

1] After studying the existing architecture and stack, GS Lab decided to containerize each component and orchestrate all the containers using K8S.
2] We also moved the stack from our custom stats collector, Graphite & Grafana to Telegraf & Prometheus for monitoring.
3] The solution required the creation of inter-component communication based on a local file system that was routed through mounted persistent volumes in K8S deployment.
4] We also introduced a DaemonSet to send logs to monitoring systems at other remote locations.
5] We set up probe checks with RPC port monitoring using custom scripts and started reporting existing stats through telegraf.
6] A monitoring mechanism was engineered using prometheus.

7] We created an end-to end components deployment on K8S.
8] All sensitive keys and certificates were moved to a K8S Vault to provide access to the required containers at startup.
9] All client keys were moved to the keystore POD.

Impact

Reduced Costs

Improved Scalability

Improved Monitoring

Deployments became simple and fast. They could be achieved completely remotely which enabled the business to scale up across multiple geographies. Now, clients could be on-boarded across the globe and served quickly. It also became possible for them to scale up individual services easily and fast.

- Technical enhancements allowed quick deployment of all PODs & containers from the repository.
- Easy individual component scalability was achieved by adjusting resources in runtime with K8S.
- K8S's built-in system monitoring tools were utilized to keep a check on each individual container and POD.
- We also reduced the risk of system threats by integrating remote systems through individual PODs and storing sensitive data in secured K8S Vaults.
- K8S persistent volume was leveraged to store data like application logs and system statistics.
- Systematic management of log rotations was achieved by using daemonsets and Cron Jobs.
- GS Lab created multiple monitoring RPC checks using custom readiness and liveness checks in cluster. This removed the dependency on ins-monitoring binary.
- Load balancing and fault tolerance across the platform was also improved using K8S platform capabilities.

All these enhancements led to greater customer delight and cemented their confidence in our client's platform.

Great Software Laboratory (GS Lab) has been the technology partner of choice to 100+ organizations across North America, Europe and Asia-Pacific for over 16 years. Leveraging our expertise in 130+ tools & technologies, we have created 300+ 'first-of-its-kind' solutions to real-world problems. Our 'Beyond code' philosophy ensures that we not only push boundaries of existing technologies but also try out newer problem solving approaches to keep our customers one step ahead of their competitors. Our global team of 1200+ employees is adept at creating 'real value' at each stage of the customer growth journey, right from proof-of-concepts to completely scaled up products. For more information about our solutions & offerings, please visit www.gslab.com

gslab™

www.gslab.com