SugarCRM leverages OPA to Save Valuable Time and Resources

SugarCRM has millions of users leveraging more than a half dozen customer experience solutions across several different geographies. SugarCRM needed infrastructure security and compliance controls that worked with modern software-defined systems, and could be tracked over time to prove compliance. The solution: Open Policy Agent (OPA) and Styra Declarative Autho rization Service (DAS).

With Styra DAS, SugarCRM saved time and resources by:

- Eliminating load balancer costs and risks with policy guardrails (managing cloud platform costs and preventing unforeseen spikes in infrastructure costs).
- Reducing time spent identifying issues and enforcing standards.
- Establishing compliance as code, simplifying audits and reporting.
- Cutting back on human error, security gaps and down time.

By automating admission control policy with Styra DAS and OPA, SugarCRM eliminated the operational, security and compliance risks that stemmed from human error. Here's a closer look at how SugarCRM leveraged these resources.

The Challenge

SugarCRM offers customer experience solutions that provide marketing automation, sales force automation, customer service, collaboration, Mobile CRM, data enrichment, and time-aware reporting for more than 4,500 companies in 120+ countries. The software pipeline at SugarCRM is highly automated, and that automation is critical for fast release cycles, ensuring high-quality software, and maintaining standards for security and compliance.

While they specialize in automation, SugarCRM still relied on some remaining manual operations, which were resource intensive and prone to human error. Prior to deploying OPA and Styra DAS, new code and app services came from multiple sources and the DevOps/platform team had to manually review all

workloads and yaml configuration to ensure proper operation and compliance while attempting to mitigate risks.

> "Styra DAS provides an automated way to build and enforce guardrails around Kubernetes deployments to prevent errors and limit risk. Moving from manual review to automated guardrails also means my team spends their cycles on crucial, more differentiated problems to accelerate our time-to-market, improve reliability and ease compliance."
>
> **Jorge Arroyo, SVP of Engineering and Cloud Operations, SugarCRM**

Senior Vice President of Engineering and Cloud Operations, Jorge Arroyo, says the manual process of reviewing Kubernetes workloads was siphoning valuable time and money. The platform team was constantly reviewing implementations and configurations, and enforcing everything manually. These manual operations opened the door to misconfiguration, due to human error.

Not only were errors hard to prevent, but without a record of all the hands-on reviews to meet security regulations, it was extremely difficult to prove compliance without further manual effort walking through code with auditors.

The Solution

Arroyo's priorities:

1. **Compliance, including readiness for periodic audits**
2. **Security**
3. **High availability, avoiding service gaps and down time**
4. **Removing friction and delays from on-boarding developers**
5. **Full GitOps visibility**

SugarCRM was already using OPA, but they knew they needed to establish governance around it as they added more and more policies and clusters—for example, teams could create external load balancers on test clusters, which can get very costly. This catalyzed action — SugarCRM knew they needed automated guardrails in place before moving into production.

Indeed, without automated controls in place, there were dozens of ways this type of oversight could manifest in risk to app reliability, security and compliance. The platform team knew that in order to scale, they had to codify policy and automatically deploy best practices. As they moved to production, they knew that container deployment was going to accelerate, yaml files would get more complex and manual policy checks simply would not scale to ensure app reliability and data safety.

After briefly considering writing their own management software for OPA, SugarCRM recognized that the solution was already available and wouldn't require more bandwidth from their team for custom development, implementation and ongoing maintenance. They relied on OPA and deployed Styra DAS, which integrated seamlessly with their existing process.

With OPA, SugarCRM had a common toolset and framework for expressing authorization policy at Kubernetes admission control. They no longer had to take the time to re-educate their team on policy standards since rules were agreed to once, and then automatically implemented as code across every cluster. Styra DAS also simplified policy enforcement with a built in library of best practices, allowing the platform team to spend less time researching which policies are important and how to write effective rules. Instead they can spend more time on differentiated work, improving platform availability and reliability and speeding time to market.

Styra DAS provided the team with a unified control plane for operationalizing OPA in production, at scale. Styra DAS removed the need for manual interaction and checks by automating compliance-as-code inside the CICD process. Arroyo and his team now build, test, distribute and monitor automated

> Styra DAS was critical to automating visibility and reporting around OPA."
>
> **Jorge Arroyo, SVP of Engineering and Cloud Operations, SugarCRM**

authorization policy for their Kubernetes clusters, to eliminate errors before they make it into production.

Adding to the operational efficiency, all policy decisions can be monitored in real time and tracked historically. That means SugarCRM can look back at every "allow and deny" decision to prove to the team and their peers in security and compliance that their policy-based controls are effective over time.

The Outcome

For SugarCRM, the days of manual rule enforcement are over. Styra DAS automates everything. That means the team can get back to doing what they do best: making better platforms so developers can focus on delivering more differentiated, innovative solutions. In short, Styra DAS dramatically cuts down on expenditures of time, money and resources while shoring up security and reducing down time.

Here's what that looks like day-to-day:

- No need for policy education, because the rules are implicit.
- No more overpaying for onboarding, training, or cleaning up user errors.
- Resource limits can be changed at any time, and enforcement happens automatically.
- Rule checks are coded inside the CI process, which means there's no need for manual interaction.
- Code checks automatically report feedback to developers so they can quickly identify and remediate any issues.
- The reduced risk of security gaps and down time means a better and more secure customer experience.
- To ease audits, every decision event is fully logged, with detailed reporting and visualization.
- It's easy to set up purpose-built policies that matter most to SugarCRM.

Arroyo reports that Styra DAS not only makes things easier, but functions as a safety net. It also allows them to stop being reactive, because there are simply fewer mistakes to react to. Styra policy-as-code guardrails, deployed early in the app development deployment cycle, mean that only the right workloads make it into production. The impact is felt on operations, security and the bottom line.

About Styra

We are reinventing policy and authorization for cloud-native. Today's cloud app infrastructure has evolved. Access, security, and compliance must also evolve. It's time for a new paradigm. It's time for authorization-as-code.

Learn more at www.styra.com

styra