



pandora® | CUSTOMER CASE STUDY

At the speed of sound

Global leader in streaming music discovery strikes the right chord using HashiCorp to accelerate application delivery from days to minutes

// Infrastructure Enables Innovation

Pandora Summary

Pandora is a leading music and podcast discovery platform, providing a highly-personalized listening experience, whether at home or on the go, through its mobile app, the web, and integrations with more than 2,000 connected products. Pandora is a subsidiary of Sirius XM Holdings Inc. (NASDAQ: SIRI). Together, Pandora and SiriusXM have created the world's largest audio entertainment company.

PANDORA FAST FACTS



Together SiriusXM and Pandora reach 100 million listeners



Replaced disconnected processes with a unified automation workflow



Rolled out a new application in 15 minutes instead of 3 days



3.3 billion ad supported listener hours*



Automated service discovery for 50,000+ services



Integrations with 2,000+ connected services

“ Every feature in each HashiCorp product leverages the others to create a complete ecosystem for our team to build from,” Cook says. “Consul and Nomad replace and automate so many of the disconnected processes that used to slow us down so we’re able to focus more on completing projects instead of worrying about how we’re going to do the work.”

CHRIS COOK,
DIRECTOR, SYSTEMS ENGINEERING, PANDORA

Tuning in and scaling up

Every month, 60 million users flock to Pandora for the audio content they love. Pandora’s proprietary Music and Podcast Genome projects provide an on-demand streaming music and podcast discovery platform that offers personalized listening experiences to its users.

Yet, for as much as Pandora’s revolutionary tools helped shape the burgeoning streaming content industry, they also presented a growing list of technical and operational challenges. The company’s longstanding do-it-yourself corporate culture resulted in a mix of ad-hoc development workflows, disparate infrastructure deployment processes, and a slew of homegrown tooling that made coordinating activities difficult and limited the organization’s agility.

“In our industry, it’s imperative to roll out new services quickly while making sure the existing ones perform at peak efficiency,” says Daniel Greene, principal systems engineer at Pandora. “Manually allocating compute resources and configuring networking elements wasn’t sustainable for a growing, service-based business. We needed to standardize and automate some of our infrastructure operations to minimize performance issues and eliminate delays in rolling out new features our users will love.”

Visibility, scale, and obstacles to overcome

As the most popular music streaming app in the United States, Pandora’s services rely on an expansive infrastructure for building, testing, and deploying new services quickly and efficiently. Traditionally, the company’s various engineering teams had the latitude to choose the databases, programming languages, and deployment methodologies that worked best for their specific team or project.

However, while the freedom to work their own way was great for creativity and productivity, it was silently creating numerous and complex issues for the infrastructure and operations team.

For starters, the manual DNS management practices meant Pandora infrastructure teams had to identify available servers and manually assign them to developers, who would use the space to code new features or services according to particular project guidelines.

Manually configuring space consumed far too much time and left the team scrambling to update their F5 and A10 load balancer configurations in a dynamic environment.

Worse, it created visibility gaps that made mapping application and code dependencies nearly impossible — especially across legacy systems. Engineers had to manually scour various servers to connect services, but had no way of predicting how services in production would respond to those in development.

“We’ve always prided ourselves on running lean and creatively applying open source technologies,” Greene explains. “But at some point the overhead of running such a variety of deployment and manual configuration processes, slowly diverging over time, creates so much complexity that it impacts our productivity and ability to deliver new services.”

A secure and standardized operation

While discovering and connecting services posed a monumental challenge, the manner in which those services were deployed was equally problematic. The team’s deep-seated culture of developers owning deployments eventually led to fractured processes in which each team managed versioning and updating its own tooling and workflows. Each team eventually came to require its own preferred skills, deployment best practices, and team dynamic. Practically, this created additional challenges for the operations team, who often had to scramble to provision resources for each team request that specified different deployment workflows and sift through volumes of Jira issues everyday just to keep servers updated.

“Developers had to manually go through a deployment checklist and could take days to ship something new because of the sheer number of steps and how many different deployment methods were involved,” says Chris Cook, Pandora’s director, systems engineering. “It created a number of incremental production bottlenecks and visibility gaps that slowed our productivity, jeopardized our ability to release new features and services, and created unnecessary stress for our team members.”

The inconsistencies across teams also made onboarding new developers increasingly difficult, even as demand for talent increased with the company’s growth. Differing workflows created longer-than-necessary learning curves that prevented new developers from immediately contributing and created incremental slow

downs that impacted long-term productivity, but couldn't be immediately recognized or addressed. "Something had to change if we wanted to continue producing great work that could support millions of users at a time and accelerate delivery of new features they'd love," Cook says. "Standardizing our deployment and configuration strategies across teams by automating service discovery and orchestrating a containerized environment would accomplish this while improving the developer experience."

Challenges



Standardizing application deployment methodologies across the enterprise



Automating networking to accelerate application delivery



Eliminating duplicate work and manual processes to improve productivity



Improving visibility and collaboration among teams

“ Nomad, Consul, and Vault pull our whole operation together into a unified ecosystem with all the features and capabilities in one place so that package deployment that used to take us two or three days can now take 15 minutes.”

DANIEL GREENE,
PRINCIPAL SYSTEMS ENGINEER, PANDORA

Operational harmony through automation

After a brief evaluation of other options, the Pandora team turned to HashiCorp Consul, Nomad, and Vault to standardize and streamline its services networking and development operations. In particular, the company was drawn to HashiCorp's product suite that offers a complete, end-to-end solution to give Pandora teams the automation products they need to work more efficiently, cohesively, and intelligently

Faster service networking for faster deployment

After a brief evaluation of other options, the Pandora team turned to HashiCorp Consul, Nomad, and Vault to standardize and streamline its services networking and development operations. In particular, the company was drawn to HashiCorp's product suite that offers a complete, end-to-end solution to give Pandora teams the automation products they need to work more efficiently, cohesively, and intelligently

As new services are deployed and registered within their environments, Consul automatically updates their load balancers with the new service information. What previously could have taken days as a manual process has been reduced to seconds and become zero-touch for operators.

“Consul enables us to fully automate networking tasks across our data center environment to provide improved, self-service alternatives to the most time-consuming aspects of our work and significantly accelerate application delivery,” Greene explains. “It also helped us reduce code dependencies by leveraging DNS and the Consul API directly while standardizing our processes across the board for a level of efficiency and observability we hadn't had before.”

In addition to service discovery and automated service networking, Pandora has deployed Consul as a service mesh to create a powerful additional way for developers to operate with least privileges by default. “While this sounds like it would obscure visibility, the ability to map and trace request paths really shines a light on a formerly dim corner, on low and high levels, of exactly what is in production and how it interacts.” explains Green. “Consul service mesh helps simplify and bring observability to a developer's deployment experience.” As a service mesh,

traffic between applications running in the service mesh is encrypted using TLS and the Envoy-based sidecars export span data allowing Pandora to track requests as they move from service to service. This data can be used to optimize the performance of these applications and provide key insights for reducing unplanned downtime.

Unlocking developer velocity

Meanwhile, Nomad — HashiCorp's high-performance container orchestrator — enables Pandora to effortlessly deploy and scale both containerized and non-containerized workloads from a single tool. Nomad streamlines large resource pool management and makes it easier to set up new hardware.

The flexible orchestration tool supports multiple types of workloads, enabling Pandora's developers to define an application's or service's deployment requirements and automatically deploy jobs onto clients — no matter if they're running on bare metal or in containers — using the built-in scheduler. At the same time, it ties with Consul and Vault, HashiCorp's secrets management solution, to automate securing service communication and streamlining data protection operations, which together drastically reduce the overall deployment process.

"Every feature in HashiCorp products leverages the others to create a complete ecosystem for our team to build from," Cook says. "Consul and Nomad replace and automate so many of the disconnected processes that used to slow us down so we're able to focus more on completing projects instead of worrying about how we're going to do the work."

Faster feature launches and 50,000 containers

HashiCorp products have transformed Pandora's development and deployment practices, standardizing previously disjointed workflows for faster, more consistent, and efficient delivery of new features and services. With Consul and Nomad, the Pandora team can deploy applications with just a few command lines, register the service using whichever name and proxy the developer prefers, and connect them automatically.

"Developers used to have to do quite a bit of work, like duplicating infrastructure, to implement products and then wait on a systems admin once all the steps were done," Greene says. "Nomad, Consul, and Vault pull our whole operation together into a unified ecosystem with all the features and capabilities in one place so that service builds and deployments that used to take us two or three days can now take 15 minutes."

Cook says that implementing HashiCorp's products has also improved the performance and resilience of its streaming service as well. Specifically, fault-tolerant Nomad helps the team orchestrate thousands of services on a global scale, automatically migrating applications to alternate hosts in case of an outage or other service degradation.

The team's early success and comfort with the HashiCorp suite has Cook and Greene eager to leverage their newfound capabilities as the company further adopts a multi-cloud, hybrid infrastructure.

"Developing a more substantive service mesh to realize the flexibility and agility of a hybrid cloud infrastructure is one of our primary objectives for the near future," Greene says. "HashiCorp has proven its ability to deliver the tools we need right now while providing a roadmap for our future plans. The long-standing support and track record of delivering on ambitious roadmaps gives us confidence that Consul and Nomad will continue providing the flexibility and efficiency we need to keep up with and stay ahead of our customers' expectations."

Outcomes



Created a standard development workflow across all development teams for greater efficiency and consistent work product



Automated service discovery for more than 50,000 service instances



Reduced lead time to application rollout from several days to 15 minutes



Enabled greater self-service capabilities for developers to deploy their services without depending on a systems admin

Solution

Pandora adopted HashiCorp Consul to automate service discovery across a containerized environment, orchestrated by HashiCorp Nomad, to create a seamless, end-to-end deployment workflow.

Pandora Partners



Daniel Greene is a principal systems engineer at Pandora Media and is responsible for helping manage the on-premises operations infrastructure that supports Pandora's primary service delivery and internal systems. Prior to joining Pandora, Daniel spent more than a decade in web hosting operations and automation.

Daniel Greene,
Principal Systems Engineer, Pandora



Chris Cook started with Pandora as a system administrator when the company could still fit into a single room, and has been responsible for architecture and foundational decisions establishing processes and platforms to enable Pandora's extreme growth.

Chris Cook,
Director Systems Engineering, Pandora

Technology Stack

- Infrastructure: Google Cloud Platform, on-premises bare metal
- Platform: Bare metal Debian, HashiCorp Nomad managed Docker Containers
- Proxies: HAproxy, fabio
- Load balancers: F5
- Firewalls: Arista
- API gateway: Internal
- CA: HashiCorp Vault
- IAM: HashiCorp Vault, LDAP
- Provisioning: Jenkins, Red Hat Ansible

