HashiCorp

ROBLOX | NOMAD CASE STUDY

How Roblox built a platform for 100 Million players with Nomad

// Infrastructure Enables Innovation

Roblox Summary

Roblox is one of the most popular gaming companies for kids and teens. Roblox not only provides a global online entertainment platform but also has forged a community with four million developers who have produced forty million games beloved by young audience. As the company grows and evolves rapidly, using Nomad enables them to scale their global gaming platform easily and reliably.

ROBLOX FAST FACTS

100,000,000

Monthly Active Players across 200+ countries

< 8 min

to deploy an application

400 developers

deploying applications on Nomad

11,000 nodes and counting

Across 20 Nomad Clusters on bare metal and AWS

200% resource utilization

Double the game servers on the same hardware

4 SRE's

to manage Nomad

The Challenge

In 2017, Roblox was growing rapidly, surpassing 30M monthly active users (MAU). Internal engineering teams were scaling as well - driving significantly higher levels of resource consumption, capacity demands, and frequency of changes.

Rob Cameron, Technical Director of Cloud Services, adeptly realized that their infrastructure could not keep up. Dedicated servers was leading to increased resource waste. The company's reliance on manual workflows and homegrown tooling was resulting in significant productivity bottlenecks. Prior to deploying Nomad, Roblox was challenged with:

Deploying a new application would take up to 8 weeks

Manual scheduling via legacy tooling

Adding additional resource capacity would take up to 12 weeks

Management of around 10,000 on-prem servers by hand

$ Annual bills of $5m+ for Window's Licensing

Running non-containerized Windows applications

Goal to migrate and containerize to Linux over time

Cameron knew that Roblox needed an orchestrator to modernize its infrastructure — a tool that could enable resource management, efficient scheduling, container adoption, and developer velocity at scale.

"Almost every weekend is the biggest weekend we've ever had in Roblox, our infrastructure had just become unmanageable to deal with application deployment in the old way," said Cameron.

Why Nomad

Roblox evaluated Kubernetes, DC/OS, Docker Swarm, and HashiCorp Nomad. Nomad won on the following criteria:

> " Nomad was able to remain in place as the single orchestrator, seamlessly deploying both Windows and Linux workloads in-place before, during, and after the migration.

Operational Simplicity

Nomad's simplicity enabled Roblox to setup a working cluster and deploy applications on bare metal in just four days. As a former consultant, Cameron knew firsthand the hidden costs of adopting trending technologies without proper evaluation of their maintenance costs over time. Since the Infrastructure team was juggling multiple initiatives (including migration to containers), Cameron valued Nomad's operational ease-of-use and lean maintenance over more complex orchestrators such as Kubernetes.

While managed services were attractive, cloud costs incurred by many software companies were extraordinarily high – 50 percent or higher of their total revenue. Infrastructure costs, if left unchecked, were serious obstacles for achieving profitability. Roblox wanted an orchestrator that they could operate themselves with a lean budget and focus towards cost savings. Nomad won with an operator-friendly UX, ease-of-use, and ability to deploy to bare metal and cloud as a single, lightweight binary.

Flexible Workload Support

Roblox's annual Windows licensing costs were rising to tens of millions of dollars. To lower costs, Cameron foresaw that the company would eventually need to migrate segments of its Windows applications. Nomad's first class workload support was a big win for Roblox's migration strategy from 32-bit Windows to 64-bit Linux.

The migration to Linux would allow Roblox to achieve greater developer productivity and finer-grained operational control. Nomad was able to remain in place as the single orchestrator, seamlessly deploying both Windows and Linux workloads in-place before, during, and after the migration.

The Result

Today, Roblox has deployed Nomad on 11,000+ nodes in 20 clusters across bare metal and cloud — serving 100 million MAUs in 200+ countries with 99.995 percent uptime.

Improved Productivity

Roblox wants to maintain performance of a large scale platform without an overreliance on additional headcount. The operational simplicity of Nomad is the key to fast adoption and high productivity. Nomad allows Roblox to reduce time on learning, debugging, and fixing infrastructure so that engineers can spend more time on adding value to its core gaming business. Other teams within the infrastructure group are able to assist in managing the Nomad deployments because the technology is easy to learn.

- <8 minutes to deploy an application globally
- ~30 minutes to onboard a new developer into deploying applications onto Nomad
- 4 SREs managing Nomad, Consul, and Vault for 11,000+ nodes across 22 clusters, serving 420+ internal developers

"We have people who are first-time system administrators deploying applications, building containers, maintaining Nomad. There is a guy on our team who worked in IT help desk for eight years - just today he upgraded an entire cluster himself."

"That's the value proposition that I hope people understand. People seem to get stuck on "I need to run Kubernetes" because my mom runs it and my friend runs it - but do you really use it? Operate it at the level that's needed?"

Cost Savings

With the right technologies and focus, Roblox successfully implemented its containerization strategy, which helps the company to scale efficiently in dollars and personnel. By containerizing their legacy game engine, upgrading to 64-bit Linux CPU, and adopting Nomad as the single orchestration platform, Roblox achieved:

- Between 150-200% resource utilization - run double the workload on the same hardware
- During the course of migration, Roblox has saved more than $10 million in Windows licensing
- Zero downtime to migrate application deployments from on-prem to AWS

"It takes no effort. We take a data center in Nomad and split it across AWS and on-prem and just increase compute in AWS and decrease compute in our local data center. We drain nodes in Nomad and the applications/jobs migrate silently from on-prem to AWS with no understanding needed by the end developers."

Roblox Partner

Rob has been solving hard technical challenges for nearly 20 years in the industry, consulting with over one thousand different organizations around security and scalable infrastructure. Before focusing on the technical challenges facing the gaming industry, he spent most of his career working for Juniper Networks in the security space.

Rob Cameron, Technical Director of Cloud Services @ Roblox

Conclusion

As Roblox goes through their containerization journey, the company recognizes the key to choosing technologies is maximizing business value and empowering engineers to solve the right problem. Nomad has enabled Roblox to scale to 100 million MAUs rapidly without additional operational overhead. The platform with Nomad ensures Roblox can continuously scale to reach more players globally.

"We didn't want to choose any technology that requires the company to drive deep expertise, almost to the point where you have to be a code contributor back into the project to get what you want. Nomad is just very easy to adopt. For our developers who understand containers and microservices, Nomad is an immediate [tool that enables us] to move forward."

Technology Stack

- Infrastructure: Majority on-premises (bare metal) , AWS, GCP, Azure
- Workload type: Linux, Windows
- Container Runtime: Docker
- Orchestrator: Nomad
- CI/CD - Jenkins, TeamCity, CircleCI, Drone
- Data Service: CockroachDB, MongoDB, InfluxDB, ElasticSearch, Vitess, MSSQL
- Storage: Portworx, EBS
- Version Control: Gitlab, Github, Github Enterprise
- Networking: Consul, HAProxy, Traefik
- Provisioning: Terraform
- Security management: Vault

HashiCorp