

# Keeping Kids Happy: How Roblox uses containers to deliver smiles

Lisa-Marie Namphy - Dev Advocate & Community Architect, Portworx  
Rob Cameron - Technical Director, Roblox

The Roblox logo, featuring the word "ROBLOX" in a bold, white, sans-serif font. The letter "O" is stylized as a square with a smaller square inside it, tilted slightly to the right.

# A Little More About Lisa-Marie Namphy

- Architecting open source communities for over 10 years
- Runs the world's largest CNCF community (Cloud Native Containers)
- 200+ meetups (Kubernetes, OpenStack, Cloud Native X, Diversity & Inclusion)
- Currently at Silicon Valley Startup: Portworx
- Loves wine, dogs, literature, sports



@SWDevAngel

# A Little About Rob Cameron

As seen on the speaker page  
of conference website



# A Little About Rob Cameron

- Rob + Lox = Roblox?
- Technical Director for Infrastructure @ Roblox
- Loves Linux, Containers, Golang, and playing cello
- Dislikes outages, gluten, bad configuration changes
- Twenty years working in tech
- Authored six books, two patents, and some code along the way
- Passionate about player experience



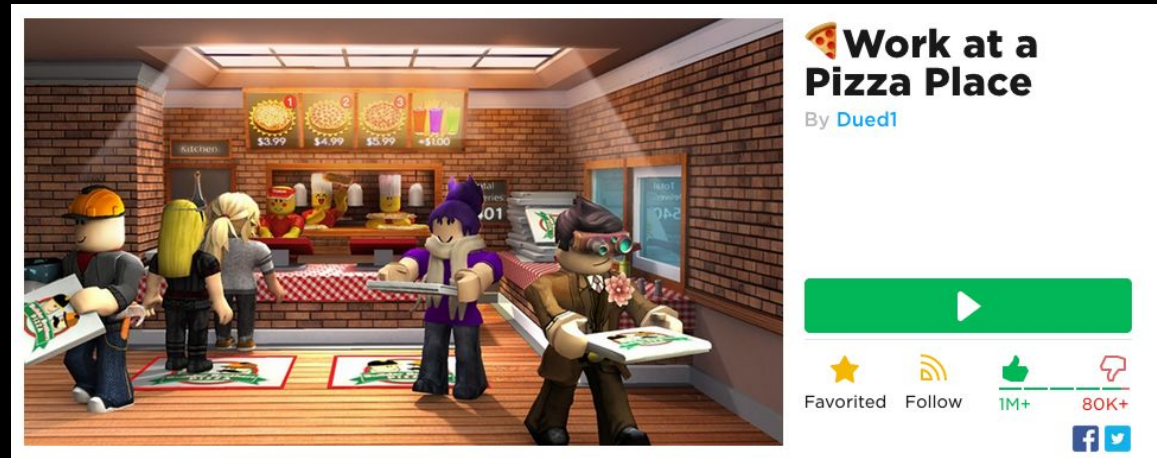
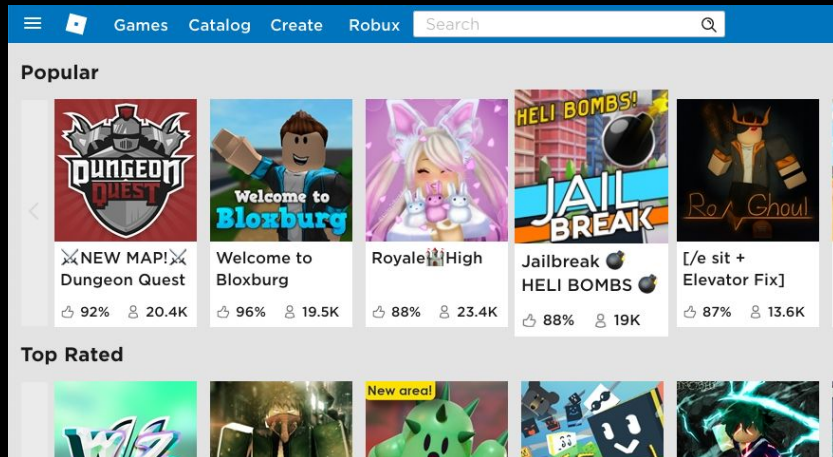


# Roblox Overview

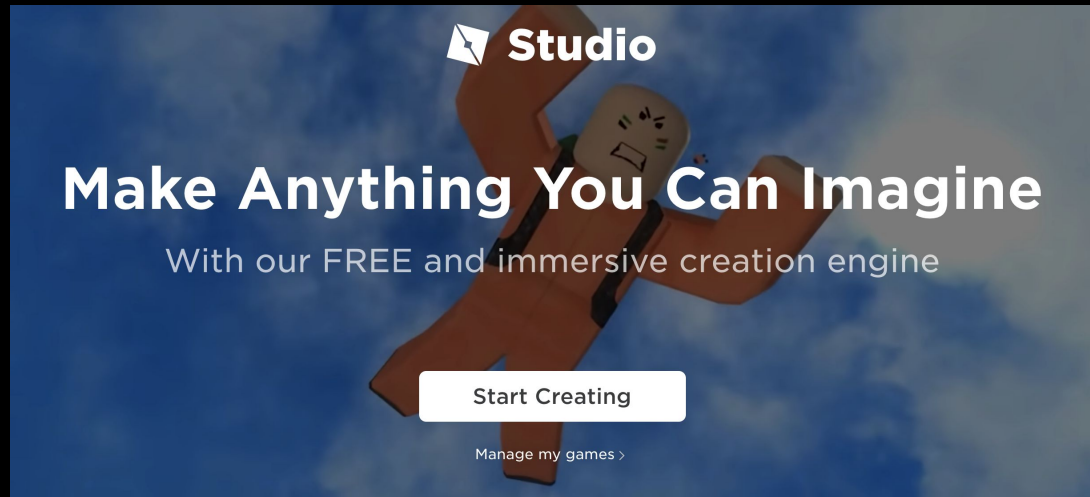
# A Little About Roblox



- Massively multiplayer and online game creation system
- Players from around the world can play together
- Anyone can create, publish, and monetize their own game
- Over 100 million monthly active users (MAU)



# Roblox Studio



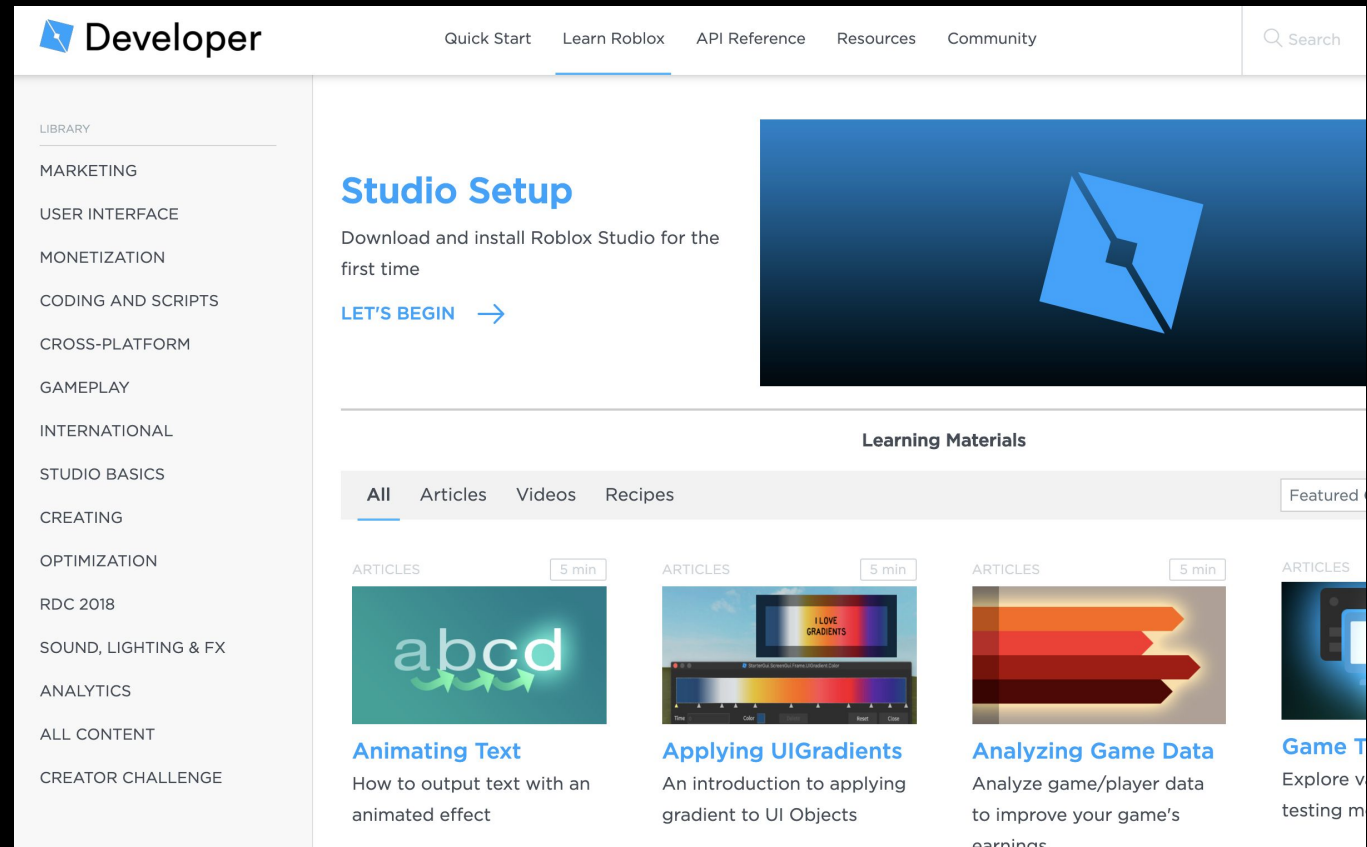
**Studio**

## Make Anything You Can Imagine

With our FREE and immersive creation engine

[Start Creating](#)

[Manage my games >](#)



**Developer** [Quick Start](#) [Learn Roblox](#) [API Reference](#) [Resources](#) [Community](#)

- LIBRARY
- MARKETING
- USER INTERFACE
- MONETIZATION
- CODING AND SCRIPTS
- CROSS-PLATFORM
- GAMEPLAY
- INTERNATIONAL
- STUDIO BASICS
- CREATING
- OPTIMIZATION
- RDC 2018
- SOUND, LIGHTING & FX
- ANALYTICS
- ALL CONTENT
- CREATOR CHALLENGE

## Studio Setup

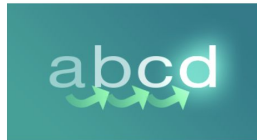
Download and install Roblox Studio for the first time

[LET'S BEGIN](#) →

### Learning Materials

[All](#) [Articles](#) [Videos](#) [Recipes](#) [Featured](#)

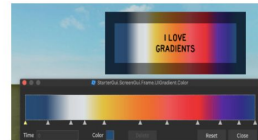
ARTICLES 5 min



### Animating Text

How to output text with an animated effect


ARTICLES 5 min



### Applying UIGradients

An introduction to applying gradient to UI Objects


ARTICLES 5 min



### Analyzing Game Data

Analyze game/player data to improve your game's earnings

ARTICLES



### Game Testing

Explore various testing methods

# Roblox Infrastructure Principals

- Build a globally available hybrid cloud to serve our players
- Reliability > Performance > Cost
  - Cost matters, but efficacy is important
- Enhance the player experience
  - fast game starts
- How do you explain to a 9 year old Roblox is broken?



# Moving Our Game Servers to Linux

## The First Big Step

- Reduce licensing costs for Windows
  - Instant savings of over \$5M/year
- Enhance capabilities for players
  - Larger game instances: 100, 200, 1000 players?
  - Migrate to 64bit for more memory/features
- Total project estimated to take around 24 months



# Moving Everything Else to Containers

## The Second Leap

- Burn down tech debt
  - Many legacy tools that are costly to maintain
- Increase server workload density
  - maybe up to a 3:1 (or more) compression
- Continue to migrate off of Windows
  - Windows is providing less value for us
- Companywide container re-education program
  - Going from pure Windows to Linux containers





# The Roblox Global Hybrid Cloud

# Where can we position our infrastructure?

- Build our own edge compute (PoPs) to be close to players
  - High density, low latency game servers
  - Edge network termination
- Build hybrid data centers
  - Mostly bare metal
  - Strategic use of cloud compute
- Global Network Backbone
  - Connect all sites/DCs/cloud providers
  - Minimize player latency



Photo by Shane Rounce on Unsplash



# Why Build When You Can Rent?

- Overall the cost of using cloud is too much for what we need
  - Networking would be a huge cost for us due to game server traffic
  - For some of our compute use cases cloud costs up to 10x more
- Strategically using cloud services
  - Some services are easier to use in the cloud due to lack of humans
  - Bursting compute as we wait for servers/racks/sites
  - Use any cloud provider for the lowest cost compute
- Long Term Investment
  - Still focused on metal in leased spaces for our cost model
  - Ultimately we will continue to reduce infrastructure costs as we can
  - Focus on strategic hires that can assist us in creating better solutions

# Bringing Compute to the Players

- Edge compute being close to the player offers the best experience
  - We utilize some amazing match making to provide this for players
  - Latency matters in gaming
- Server Density
  - Design servers with a reasonable amount of players/node
  - More servers per rack
  - Less racks per site to reduce physical space
- Networking
  - High bandwidth, low latency connections across the planet
  - Backbones, PoPs and DCs offer lots of connectivity
  - Managing network capacity often harder than server capacity



# Orchestrating Services

Photo by Manuel Nägeli on Unsplash



# Shipping With Containers

- All in one shippable environment
  - Patch the container, not just the OS
  - Let developers control their own environment
- Cgroup security controls
  - Memory Limits/CPU management
  - Limiting syscalls
- Transforming your organization to support
  - A perfect way to destroy your company
  - Education and tooling need to be a focus

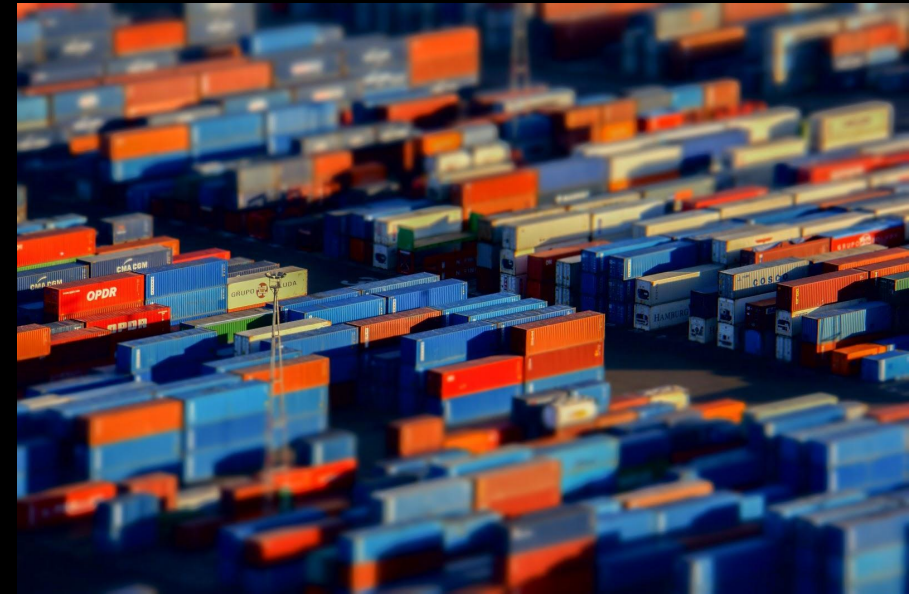
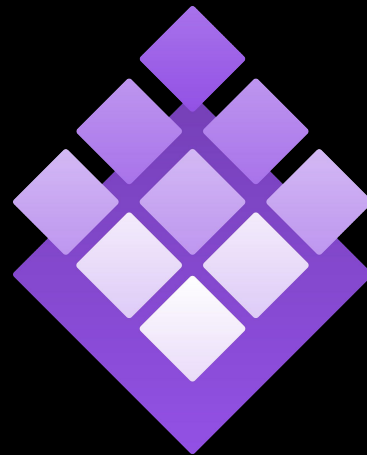
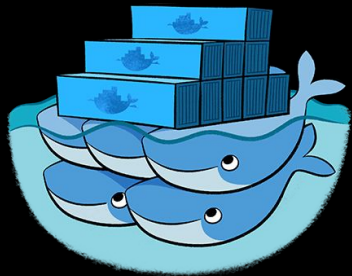


Photo by Tim Easley on Unsplash

# Choosing An Orchestrator

- Which orchestrator should we use?
- How many people will we need?
- Will we need Windows support?
- How can you not choose Kubernetes?



# Using The Hashistack + Portworx

Nomad, Consul, and Vault

- Operational simplicity
- Easily containerized
- Multi-platform/workload support
- Added Portworx for reliable storage
- Mostly managed by a team of 4 people



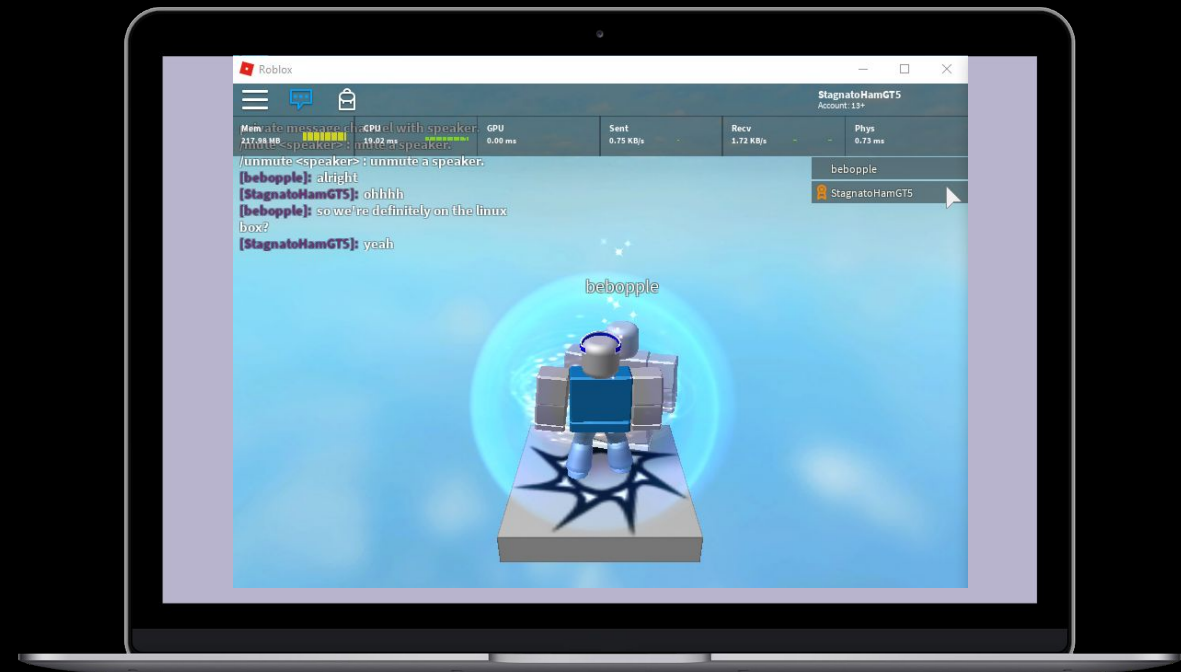
# Migrating Our Game Servers



- Convert ~15,000 servers over to Linux
  - A two year project condensed to 10 months
  - Deployed one PoP per day across 8 days for initial launch
- Added 11 more PoPs within one year of initial deployment
  - Started with a few hundred nodes per site
  - Some sites over 1,000 game servers alone
- Manage game service deployments with Nomad
  - Deploy, upgrade, and secure service deployments
  - Reduced deployment time from hours to minutes
  - Secure secret management and rotation
  - Global deploys to in ~8m

# The Penguin Has Landed (on Game servers)

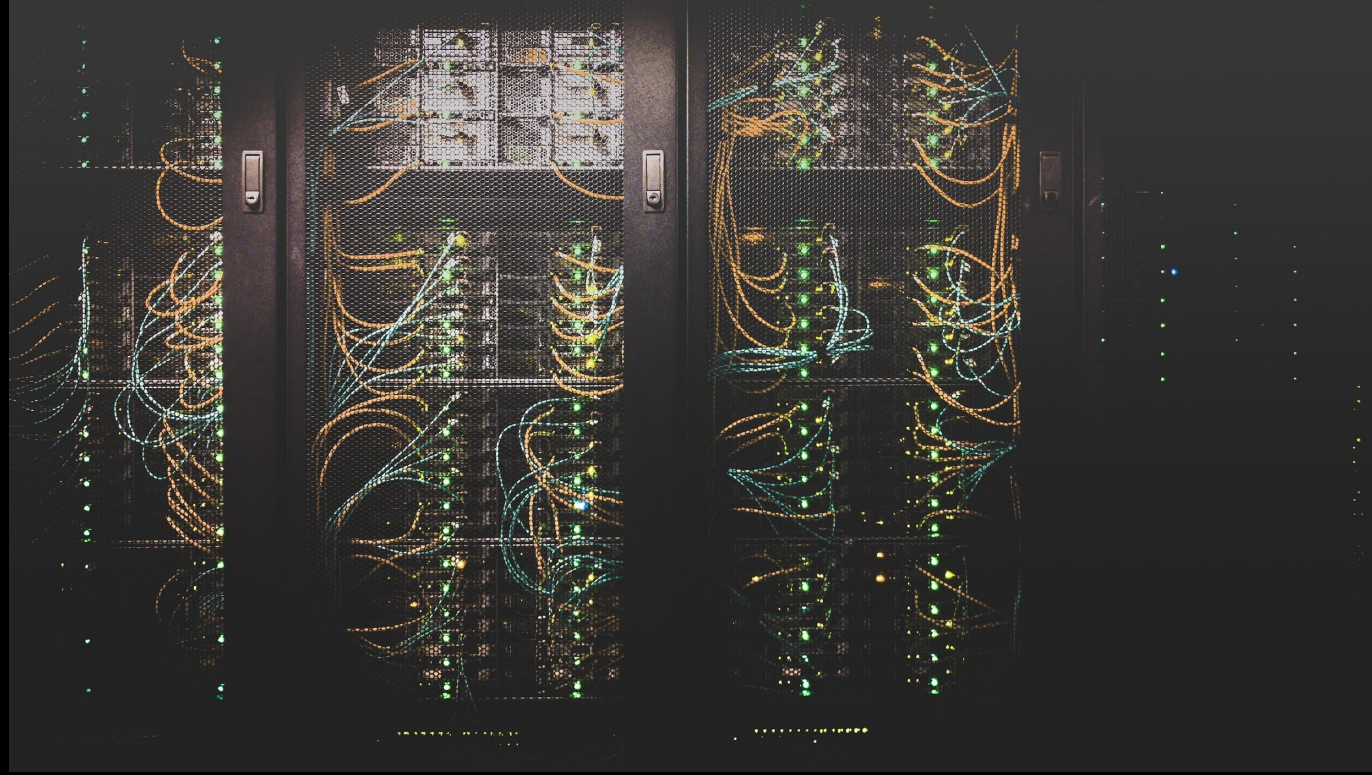
- ~200,000 active containers (~350,000 today)
- ~5000 orchestrated hosts (~12,000 today)
- Increased server capacity
  - 1.5 - 2x game instance per server
  - Move to 64bit
- Linux Kernel woes
  - Long time SLAB bug
  - Finally fixed in Kernel 5.3



# Migrating Our Platform Gradually



- Straight to Linux
  - Some services can easily be ported to run on Linux
  - Most of our code base is C# and mostly works
  - Other services need a rewrite (or want to rewrite)
- Running Windows Services With Nomad
  - We wrote our own driver to run our existing services
  - This will help us burn down a lot of old tech debt
- Scaling services sanely
  - Autoscaling can make bad code run at a larger scale
  - Ensuring that we don't provide more resource without correct usage



# Storage and Networking

Photo by Taylor Vick on Unsplash

# Reliable Container Storage

- Challenges
  - Data that is worth storing is valuable to your organization
  - Data that is stored should not be lost
  - Using the solution should be easy and require little maintenance
- Desires
  - Snapshots
  - Encryption at rest
  - Performant
  - Scalable



# Portworx Container Storage



- Total of ~22 clusters globally
  - Integrated with Nomad, simple to deploy new jobs with storage
  - ~10PB of global storage
- Use Cases
  - Consul, Nomad, Docker Registries
  - Telemetry systems (InfluxDB, Prometheus, Grafana)
  - Databases (PostgreSQL, CockroachDB, MySQL, MSSQL Linux)
  - Build volumes (Drone)
- Technical Support
  - Generally continues to run with little intervention
  - Awesome TAC/Support for when we make bad choices

# Container Networking

- Keeping it simple
  - Using Nomad's default networking solution (Docker Bridge, Host mode)
  - Minimize support effort for complex networking solutions
- Traefik
  - One of the larger Traefik deployments in the world
  - Some scalability challenges, working various solutions
- Gocast
  - BGP anycast network solution with Consul integration
  - <https://github.com/mayuresh82/gocast>
- Service Mesh
  - Consul connect (planned)
- CNI
  - Maybe?

# Global Network Backbone

- Internet and provider peering at all PoPs
  - Connect with IX, ISPs, and SPs
  - Backbone connectivity
  - Cloud provider Connectivity
- Global traffic often exceeds 1.2Tbp/s
  - 50x growth over the last two years
  - Gaming Traffic
  - Platform Services/Web Traffic
- Latency Matters
  - Player experience for gaming is key
  - Game starts, web page load times

# OSS Load Balancing Stack

- Building our own Ingress Edge (~100Gbp/s + web traffic)
  - Scalable solution that empowers long term growth
- GLB/L4LB
  - Github Load Balancer for L4
  - Strong solution with several pull requests provided
- HAProxy
  - Awesome scalability with infinite\* configuration options
  - Provided a lot of missing observability
- Edge/Core Termination
  - Latency reduction (200-500ms in remote regions) for Web
  - Game starts 500ms faster vs Vendor solution
  - Dynamic termination based on latency to PoPs



# Tooling and Education

Photo by Clem Onojeghuo on Unsplash

# Technology is Easy, People are Difficult

- Containers are a perfect way to destroy your company
  - Containers potentially require a lot of changes to internal systems
  - People often do not like change, even if the end goal is better
- Moving to containers is hard
  - Unsurprisingly a lot of applications may not be ready to drop in containers
  - Lots of tooling may not be compatible
- Moving from Windows services to Linux containers is harder
  - Lack of familiarity with how containers work
  - Lack of familiarity with Linux
  - MSFT is doing a lot to change this and it is appreciated

# Observability is Key

- Orchestration is complicated, are you sure it is working?
  - Smaller services can block an entire cluster/deployment
- Everyone will complain, can you show them everything is OK?
  - Giant dashboards may lead to confusion
- The perception of how a system works comes through lots of data
  - Working to simplify the data to show system status is helpful

# Tooling to Scale

- Chef
  - Configuration management for servers
  - Can be abused for operational work (probably don't do this)
- Ansible
  - Operational tasks for servers
  - Software Updates, controlled changes
- Terraform
  - Trying to use this as much as possible/practical
  - Configuration as code for infrastructure
- Custom tools
  - PowerShell
  - Golang Tools
  - Microservices



# Would You Like to Know More?

- Portworx Architect's Corner with Rob Cameron
  - <https://portworx.com/architects-corner-roblox-runs-platform-70-million-gamers-hashicorp-nomad/>
- HashiCorp Case Study
  - <https://www.hashicorp.com/case-studies/roblox/>
- Ubuntu Masters
  - <https://www.youtube.com/watch?v=SwIU4U-BWRo>
- Container World
  - <https://www.youtube.com/watch?v=tkyqzfi4h14>
- Roblox Linkedin (We are always hiring!)
  - <https://www.linkedin.com/company/roblox>

# What Matters Most?

- Keeping kids happy

# Questions?



# Rate today's session

**Cyberconflict: A new era of war, sabotage, and fear**

See passes & pricing

David Sanger (The New York Times)  
9:55am-10:10am Wednesday, March 27, 2019  
Location: Ballroom  
Secondary topics: Security and Privacy

**Rate This Session**

We're living in a new era of constant sabotage, misinformation, and fear, in which everyone is a target, and you're often the collateral damage in a growing conflict among states. From crippling infrastructure to sowing discord and doubt, cyber is now the weapon of choice for democracies, dictators, and terrorists.

David Sanger explains how the rise of cyberweapons has transformed geopolitics like nothing since the invention of the atomic bomb. Moving from the White House Situation Room to the dens of Chinese, Russian, North Korean, and Iranian hackers to the boardrooms of Silicon Valley, David reveals a world coming face-to-face with the perils of technological revolution—a conflict that the United States helped start when it began using cyberweapons against Iranian nuclear plants and North Korean missile launches. But now we find ourselves in a conflict we're uncertain how to control, as our adversaries exploit vulnerabilities in our hyperconnected nation and we struggle to figure out how to deter these complex, short-of-war attacks.

**David Sanger**  
The New York Times

David E. Sanger is the national security correspondent for the *New York Times* as well as a national security and political contributor for CNN and a frequent guest on *CBS This Morning*, *Face the Nation*, and many PBS shows.

Session page on conference website

Attending Notes Remove

**Cyberconflict: A new era of war, sabotage, and fear**

9:55 AM - 10:10 AM, Wed, Mar 27, 2019

**Speakers**

David Sanger  
National Security Correspondent  
The New York Times

Ballroom

*Keynotes*

David Sanger explains how the rise of cyberweapons has transformed geopolitics like nothing since the invention of the atomic bomb. From crippling infrastructure to sowing discord and doubt, cyber is now the weapon of choice for democracies, dictators, and terrorists.

**SESSION EVALUATION**

O'Reilly Events App

# Thank You!

