Ryan Gerstenkorn

ryan_gerstenkorn@fastmail.fm
(360) 296-8611

SR DEVOPS ENGINEER, BRAVE; MAR 2018 - OCT 2018

- Lead implementation of update client and infrastructure for the new Chromium based Brave browser.

- Worked on the build infrastructure for the Chromium based browser while maintaining the older Muon version.

- Automated and improved existing AWS infrastructure for several Brave/BAT projects including ad infrastructure, chromium component updates, brave.com, basicattentiontoken.com, publishers.basicattentiontoken.org, and the BAT ledger micropayments system.

DEVOPS ENGINEER; CURSE SEP 2014 - MAR 2018

Twitch and Amazon sections are work for or relating to that company while working under the Curse subsidiary of Twitch/Amazon.

CURSE SEP 2014 - MAR 2018

- Lead operations and networking with the goal of reducing IT overhead by building services to automate common tasks and codify best practices. Managed an in-house BGP/OSPF network of 2K/200/50 nodes/services/sites running across most of well known web stacks, fronted by a distribution network serving 20B requests monthly. In this setup we where able to achieve 99.999% uptime.

- Lead the migration from Curse's private cloud to entirely AWS based infrastructure and designed the cloud infrastructure to replace the Curse private cloud. Scalability and security was met by implementing minimal loosely coupled design then stamping out discrete infrastructure as needed during migrations. Obsessing over alternatives for any multi-tenant resources along with a strong preference for optimizing expertise over always using the correct tool allowed us as a small team of 2 other people to pull the plug on our custom private-cloud in just one year.

    - Heavily relied on terraform modules along with custom tooling to reduce code duplication and allow us to safely migrate, manage and scale all of our public facing services. Terraform allowed us to largely replace tooling for 200+ clustered services on our private cloud with around 50 modules and about 50 app level repo's, one for each of our AWS accounts.

    - Isolating one public service to one account allowed us to restrict the effect of accidental or malicious changes while orchestrating RBAC based permissions and users with cross-account IAM roles.

- Worked an extensive on call and operations during the early days of the Twitch desktop app's AWS infrastructure, at the time (2014-2015) this was known as the Curse client. Much of this was related to scaling and maintaining our Aerospike DB, replicated world-wide over a OSPF routed VPN mesh.

- Implemented and managed a redundant and scalable monitoring and graphing system recording over 1 million metrics a minute across thousands of machines with Sensu, RabbitMQ, Redis, Carbon/Graphite.

- Migrated linux infrastructure from Puppet to Chef, and later refined this to simplify dependency management by adopting the Berkshelf workflow, unit testing and automated deployment.

- Designed and implemented a hardened LDAP password reset web app exposed through 2FA enforcing proxy and WAF.

- Maintained a site-wide job scheduler and coordinator written in python + celery which ran across approximately 1k nodes.

- Wrote and maintained a distributed API scraping cluster in C# utilizing RabbitMQ.

- Built a CI/CD system for code and infrastructure deployments which allowed for automatic spin-up and discovery of new services from source control with Jenkins pipelines. Designed to use cross account roles and implied repo naming schemes between various services allowed for a simple stateless way of managing user and deploy access between separate accounts and systems.

- Wrote tests a large amount of the internal Chef infrastructure tooling we used at Curse. We had approximately 100 private cookbooks and I wrote or rewrote about 70 of those adding tests.

- Handled security incident response and post mortem apart of the ops call rotation at Curse and Brave, responding to multiple issues off hours at both.

- Found and fixed a privileged management access bypass with the original (now deprecated) Curse authentication API despite it taking several month’s due to needing to work with multiple teams depending on the specific broken behavior.

- Maintained Curse’s OSPF and BGP networks, occasionally deep diving into network troubleshooting and making configuration changes to fix identified issues.

- Maintenance, configuration, and troubleshooting of Brocade networking and Layer 7 load balancing equipment. At one point tracked down and identified a work around for a long running issue to a bug in the L7 Brocade barrel processors with POST uploads occasionally failing when routing between CloudFlare and our load balancer. Tracking down the source of the issue required working with CloudFlare engineers to gather packet captures where ever possible since we where unable to reproduce the issue outside of our prod environment (or even on identical staging configurations).

- Managed 20 Gluster clusters of 3 nodes across ~5TB of storage using a personal cookbooks. Ran approximately 3 site wide system redeployments and several more updates with little service impact.
  - One update that did lead to availability compromise was related to a change in how NFS file locks are handled by the Linux kernel. This was quickly identified and resolved thanks to comprehensive statistics monitoring through Sensu and Carbon and displayed in Grafana.

  - Experience optimizing clustered storage for specific conditions including high contention and globally replicated latency sensitive workloads.

TWITCH AUG 2014 - MAR 2018

- Managed a tight deadline to exceed the Curse IT acquisition requirements set by Twitch. My effort here was recognized with the first employee of the month award under the then new Curse subsidiary.

- Worked to increase security across Curse's infrastructure by integrating our wide range of app's and services with Twitch's internal secret management system. As well as other common services such as monitoring where possible.

- One notable event was during the acquisition the Curse client which I had worked on in 2014-15 and which is mentioned above was rebranded and now available as the Twitch desktop client.

AWS/AMAZON AUG 2014 - MAR 2018

- Lead infrastructure design and deployment of amazonforums.com replacing the previous Amazon digital and device forums hosted under the amazon.com domain.

ENTERPRISE NETWORK TECHNICIAN, API DIGITAL; JUN 2014 - SEP 2014

- Provided emergency network support for Cisco and Adtran networking equipment. In some cases providing support directly on behalf of the manufacturer themselves, for example I provided support for various enterprise networks running Adtran equipment while acting as a representative of Adtran.
- On the side I also occasionally did automation and operations, for example I worked on a RT ticketing system build using Ansible.

SYSTEMS ADMINISTRATOR, DIGITAL FORTRESS; APR 2012 - JUN 2014

- Provided Sr support for Unix, OS X, and Windows systems where I was often times the highest level of support.

- Lead several large re-architecting projects including simplifying and unifying the scattered collection of Nagios monitoring systems, making sure that they remained secure and maintainable going forward.

- Replaced the aging backup system with a Bacula based infrastructure secured through an internal maintained x509 CA and separate physical backup network.

- Identified and resolved several critical issues, one of which risked taking 50k DNS records offline for an extended period of time. Addressed the mentioned issue and took steps to ensure long term availability and maintainability of the DNS infrastructure such as rebuilding and then documenting maintenance procedure for the authoritative DNS servers.

- Rebuilt the Unix web infrastructure and worked on moving to an automated infrastructure using Puppet.

REPAIR TECHNICIAN, VARIOUS; OCT 2008 - APR 2012

- Repaired Windows, Apple and Linux devices
- Circuit board soldering/repair and SMD reflows

VOLUNTEER

- Contribute and Maintain several open source projects
  - https://github.com/SousChef/varnish – Previously the main maintainer for this Chef cookbook.
  - https://github.com/RyanJarv/coderun – CLI wrapper for transparently running shell scripts in a container and severless environments.
  - https://github.com/RyanJarv/dockersnitch – Netfilter NFQUEUE based container networking interceptor. This was a quick proof of concept, it works but that's about it.
  - Wrote Javascript tests for the original Brave BAT client.
- RElectronics repair and recycling (2009-'10)
- Counselor, 4H (2006-'08)

Skills / Other / Notes

- Currently fluent in Chef, Python, Ruby, Bash, Go, C#, have previously used Puppet extensively and occasionally use and familiar with CloudFormation, PowerShell and Javascript. I've worked on several application and Infrastructure CI/CD pipelines and am familiar writing tests in most languages.

- Have worked most often with the following services: EC2, VPC, PrivateLink, ELB, Lambda, ElasticBeanstalk, ECS, RDS (Aurora/MSSQL), ElastiCache (Redis and Memcache), CodeDeploy, CloudFormation, CloudWatch, Route53, CloudFront, WAF, SQS, SNS, SES, S3, EFS, ElasticSearch, Fargate, ECR, Cognito, SSO, Organizations.

- Proficient with various container technologies having used Docker, docker-compose and the Docker Go lang API fairly extensively. More recently have been digging into Kubernetes and helm recently using it at https://github.com/RyanJarv/randrust (which is mainly just an example rust service/k8's helm deployment repo).

- Deep knowledge of networking fundamentals with plenty experience debugging low level protocol and hardware issues in complex production environments.

- Discovered and reported an issue with the AWS Route 53 API allowing non-detectable persistent control over specific domain names. (https://blog.ryanjarv.sh/2019/05/24/backdooring-route53-with-cross-account-dns.html)

- Proficient at gathering storage related metrics through system and tracing tools. Both for day to day health monitoring as well as hunting down hard to find storage bugs in Gluster, NFS, SMB, and Aerospike. These ran in environments ranging from both stable low latency inter-rack environments to very unreliable, high latency, multi-continent but performance critical.