# Ryan Parman

**Cloud-native engineering leader with a focus on reliability, scalability, and security for the modern web.**

**Links:** [GitHub (personal)](https://github.com/skyzyx) • [GitHub (side project)](https://github.com/northwood-labs/) • [LinkedIn](https://www.linkedin.com/in/rparman/) • [Stack Overflow](https://stackoverflow.com/users/228514/ryan-parman) • [Role-targeted résumés](https://github.com/skyzyx/resume/blob/master/resumes/)  
**Format:** [Web](https://github.com/skyzyx/resume/blob/master/resumes/ryanparman-general-cv.md) • [PDF](https://github.com/skyzyx/resume/blob/master/resumes/ryanparman-general-cv.pdf) • [Word](https://github.com/skyzyx/resume/blob/master/resumes/ryanparman-general-cv.docx) • [OpenDocument](https://github.com/skyzyx/resume/blob/master/resumes/ryanparman-general-cv.odt) • [Raw Markdown](https://github.com/skyzyx/resume/raw/master/resumes/ryanparman-general-cv.md)

## Summary

Ryan Parman is a cloud-native engineering leader with over 25 years of experience, who specializes in technical leadership, software development, site reliability engineering, and cybersecurity for the modern web. A seasoned problem-solver who excels at listening, adapting, and driving continuous improvement. Committed to delivering exceptional work, building impactful solutions, and elevating team performance.

## Work Experience

Older roles are truncated for brevity. If interested, details can be found [on GitHub](https://github.com/skyzyx/resume/blob/master/ryanparman-previously.md).

### [McGraw Hill](https://www.crunchbase.com/organization/mcgraw-hill-education) — Remote (since COVID), previously Seattle, WA

#### Principal Engineer, Cloud Center of Excellence (January 2024—October 2024)

* Assumed a role influencing the technical direction of the entire organization. Ensured a focus on real-world, actionable feedback and provided strategic direction aligned with practical needs.
* Continued to be involved in the oversight and direction of our AWS stack, security, guardrails, and more.
* Identified opportunities to extend the security measures and guardrails developed for AWS to other cloud platforms.

#### Principal Cloud and Platform Engineer (June 2020—January 2024)

* Transitioned from Engineering Manager to a strategic technical leadership role.
* Prolific documentarian. Documentation is worth 50% of your grade.
* Either directly or collaboratively designed and maintained [AWS Control Tower](https://aws.amazon.com/controltower/), [Artifactory](https://jfrog.com/artifactory/), [GitHub Enterprise](https://github.com/enterprise), [GitHub Actions](https://github.com/features/actions), [CircleCI](https://circleci.com/enterprise/), [Jenkins](https://www.jenkins.io), and more.
* Partnered with McGraw Hill Enterprise Architecture and [AWS Professional Services](https://aws.amazon.com/professional-services/) to deploy [AWS Control Tower](https://aws.amazon.com/controltower/) and [AWS Identity Center](https://aws.amazon.com/iam/identity-center/). Lowered costs and increased control over account guardrails.
* Managed the program for building and maintaining base AMIs for all of McGraw Hill. Leveraged insights from the [Center of Internet Security](https://www.cisecurity.org), security patching, and the specific needs of internal customers to develop a unified build pipeline integrating best practices.
* Using [AWS SDKs](https://aws.amazon.com/developer/tools/), conducted comprehensive scans of Route 53 to obtain a mapping of thousands of active websites owned by McGraw Hill. Focused on identifying and remediating misconfigurations, rotating certificates, and more.
* Co-implemented self-hosted runners for GitHub Actions. Focused on the Linux runtime environment.
* Rebuilt our [Artifactory](https://jfrog.com/artifactory/) cluster with a “cattle, not pets” approach. Ran the project from inception to completion, including the majority of development. Worked across dozens of teams and hundreds of services to complete the project.
* Enabled continuous token and password rotation for our engineering teams by building a *Token Vending Machine*, providing a "push-button, receive-token" solution.
* Proactively added support for lower-cost ARM64 CPUs, opening the door for ~$450k/year in cost savings.

#### Engineering Manager, Site Reliability (October 2018—June 2020)

* Managed a team of four, while working to level-up the team's technical skills and leadership capabilities. Conducted regular 1:1s, performance reviews, and career development discussions.
* Led the [*Site Reliability Engineering*](https://sre.google/in-conversation/) (SRE) team in addressing macro-oriented problems affecting decentralized, heterogeneous engineering teams across the company. Empowered greater self-service for engineering teams.
* Revamped the Seattle SRE interview process to prioritize a 70/30 focus on software engineering (Dev) and systems operations (Ops). Emphasized leadership qualities, bias for action, and high curiosity.
* Owned and served as the key decision-maker in development of a core platform for company-wide, reliability-focused projects.
* Formed and led a leadership group to establish a process maintaining reusable Terraform modules which could be composed together according to a service’s needs.
* Customized the Amazon Linux AMIs to comply with Level-2 [CIS](https://www.cisecurity.org) Guidelines for both Amazon Linux and [Docker](https://www.docker.com). Collaborated closely with cybersecurity, operations, and various business units to ensure compliance.
* Developed custom security and operational tooling where off-the-shelf tools wouldn't give us what we needed, to understand the current posture of ±200 AWS accounts.
* Reduced the time to deploy a new service from dozens of weeks to a single meeting by implementing a *Monitoring-as-Code* methodology, and defining broad-use [Service Level Objectives](https://sre.google/sre-book/service-level-objectives/) (SLOs).

#### Staff Software Engineer (October 2016—October 2018)

* Led the development of multiple Tier-1 services within the educational content authoring pipeline, leveraging technologies such as [REST](https://martinfowler.com/articles/richardsonMaturityModel.html), [GraphQL](https://graphql.org), API design, [Amazon ECS](https://aws.amazon.com/ecs/), [Docker](https://www.docker.com), [Terraform](https://www.terraform.io), [ePubs](https://www.w3.org/publishing/epub3/), and security best practices.
* Provided the technical direction of these projects, promoted their adoption across the organization, provided comprehensive documentation, and offered ongoing guidance on adoption.
* Led the development of the authoring component of [McGraw Hill’s SmartBook 2.0 product](https://www.mheducation.com/news-media/press-releases/mcgraw-hill-connect-unveils-smartbook.html), and the internal system which indexes authored content, builds ePubs, and encodes images/video for McGraw Hill’s ePub CDN.
* Introduced the adoption of continuous integration (CI), continuous delivery (CD), rapid deployment practices, and Docker containers.
* Introduced a more hands-on monitoring approach, enabling development teams to actively engage in their own operations. Achieved significantly lower *Mean Time to Recovery* (MTTR).
* Served as a core resource in adopting *Infrastructure-as-Code* (IaC) tools such as [Terraform](https://www.terraform.io) and [Packer](https://packer.io).

### [WePay](https://www.crunchbase.com/organization/wepay) — Redwood City, CA

#### DevOps Engineer (April 2015—September 2016)

* Led a cross-company initiative to upgrade the monolithic application from PHP 5.4 to PHP 5.6 (the latest at the time). Facilitated cross-team collaboration among all major engineering teams and QA departments in order to achieve results.
* Initiated a program to automate the creation of base server images for our cloud servers. They allowed new servers to boot and begin serving traffic ~75% faster.
* Began investigating ways to implement *configuration-as-code* for our cloud infrastructure.

#### Senior API Engineer (April 2014—April 2015)

* Took the lead on the company’s [HackerOne](https://www.hackerone.com) program, coordinating across teams to address security issues.
* Built a development environment for engineering teams. Reduced new engineer onboarding time from 2 weeks → 1 day.
* Instrumental in designing WePay’s MFA-as-a-Service offering. (U.S. patent filing [US15042104](https://patents.google.com/patent/US20160241536A1/en?inventor=Ryan+Parman))
* Developed new API endpoints to help expand WePay’s business and support its partners.

### [Amazon Web Services](https://www.crunchbase.com/organization/amazon-web-services) — Seattle, WA

#### Web Development Engineer II (March 2010—April 2014)

* AWS hard-forked my open-source *CloudFusion* project into the [AWS SDK for PHP](https://aws.amazon.com/sdk-for-php/), then hired me to work on it.
* Collaborated with the [AWS Elastic Beanstalk](http://aws.amazon.com/elasticbeanstalk/) team to provide PHP support for the platform, which launched in March 2012.
* Played a key role in the creation and development of the [AWS SDK for PHP](https://aws.amazon.com/sdk-for-php/) v2, incorporating significant changes in the PHP language and community since CloudFusion was first written in 2005.
* Collaborated with the AWS Design team on the [AWS Management Console](https://console.aws.amazon.com), to build a robust and user-friendly console. Led one of the first teams to provide reusable UI building blocks at AWS.
* Focusing on Amazon’s *Customer Obsession* leadership principle, I successfully pushed for being better stewards of our community. Included increased transparency, better communication, and improved tooling for developers. [[Examples](#Xa39a3ee5e6b4b0d3255bfef95601890afd80709)]

## Projects

Proof that I can code, call APIs, interact with SDKs, and build user-facing software. I have live-coding anxiety, so live-coding interviews will always present me at my worst, not my best.

* **DevSec Tools:** Building a [website](https://github.com/northwood-labs/devsec-ui), [CLI tool, and Go library](https://github.com/northwood-labs/devsec-tools) for helping developers identify potential web security configuration issues (in-progress).
* **Custom Linux Packages:** Building a [repository of custom Linux packages](https://github.com/northwood-labs/package-building/wiki) (in-progress).
* **CSP Evaluator:** Building a [parser and evaluator for Content Security Policy (CSP) directives](https://github.com/northwood-labs/csp-parser) in Go (in-progress).
* **Terraform Provider:** Built a [custom provider](https://github.com/northwood-labs/terraform-provider-corefunc) which provides a set of utility functions for use in Terraform/OpenTofu.
* **Multi-Platform Docker:** Built a [downloader for GitHub release assets](https://github.com/northwood-labs/download-asset) which simplifies building multi-platform images.
* **AWS Organization Security:** Built a [library + CLI tool](https://github.com/northwood-labs/assume-spoke-role) which simplifies the AWS pattern for multi-account organizations which they call “hub and spoke.”
* **AWS Session Manager:** The terminal is the right tool for shell sessions. Built a [TUI](https://en.wikipedia.org/wiki/Text-based_user_interface) for [simplifying connections to SSM-enabled EC2 instances](https://github.com/northwood-labs/ssm-shell) using your Terminal.
* **Configuration for tflint:** Built a [tool for generating up-to-date configurations for AWS/GCP/Azure](https://github.com/northwood-labs/tflint-config-generator) for use with [tflint](https://github.com/terraform-linters/tflint).

## Recommendations

See a [selective list of recommendations](https://github.com/skyzyx/resume/blob/master/selected-recommendations.md) from co-workers and peers.

## Keywords and Skills

* **Languages:** [Bash](https://www.gnu.org/software/bash/), [Go](https://go.dev), [PHP](https://www.php.net) (modern), [Python](https://www.python.org).
* **Linuxes:** [Alpine Linux](https://alpinelinux.org), [Amazon Linux](https://aws.amazon.com/linux/), [CentOS](https://en.wikipedia.org/wiki/CentOS), [Ubuntu](https://ubuntu.com).
* **Cloud/DevOps/SRE:** TLS and cipher suites, [ARM64](https://aws.amazon.com/ec2/graviton/), [AWS Control Tower](https://aws.amazon.com/controltower/), [AWS Elastic Beanstalk](http://aws.amazon.com/elasticbeanstalk/), [AWS Identity Center](https://aws.amazon.com/iam/identity-center/), [AWS Lambda](https://aws.amazon.com/lambda/), [AWS RDS Aurora](https://aws.amazon.com/rds/aurora/), [AWS SDKs](https://aws.amazon.com/developer/tools/), [AWS Secrets Manager](https://aws.amazon.com/secrets-manager/), [AWS Well-Architected](https://aws.amazon.com/architecture/well-architected/), [Amazon ACM](https://aws.amazon.com/certificate-manager/), [Amazon CloudFront](https://aws.amazon.com/cloudfront/), [Amazon EC2](https://aws.amazon.com/ec2/), [Amazon ECS](https://aws.amazon.com/ecs/), [Amazon Route 53](https://aws.amazon.com/route53/), [Amazon S3](https://aws.amazon.com/s3/), [Ansible](https://www.redhat.com/en/technologies/management/ansible), [Artifactory](https://jfrog.com/artifactory/), [CIS](https://www.cisecurity.org), [CentOS](https://en.wikipedia.org/wiki/CentOS), [Docker](https://www.docker.com), [EC2 Image Builder](https://aws.amazon.com/image-builder/), [GCP](https://cloud.google.com), [GitHub Actions](https://github.com/features/actions), [GitHub Enterprise](https://github.com/enterprise), [Nginx](https://www.nginx.com), [Packer](https://packer.io), [Redis](https://redis.io), [Terraform](https://www.terraform.io), [kubectl](https://github.com/kubernetes/kubectl), automation, cloud configuration security, multi-platform development, operational reliability, performance, scalability.
* **SDE/SWE/DevTools:** API design, API versioning, CLI tools, [CircleCI](https://circleci.com/enterprise/), [Docker](https://www.docker.com), [GitHub Actions](https://github.com/features/actions), [Git](https://git-scm.com), [GraphQL](https://graphql.org), [JWT](https://jwt.io), [NFS](https://en.wikipedia.org/wiki/Network_File_System), [REST](https://martinfowler.com/articles/richardsonMaturityModel.html), [Redis](https://redis.io), [Subversion](https://subversion.apache.org), [Vagrant](https://www.vagrantup.com), [WordPress](https://wordpress.org), [XSLT](https://developer.mozilla.org/en-US/docs/Web/XSLT), [ffmpeg](https://ffmpeg.org), [twelve-factor applications](https://12factor.net), automation, building platforms, code generation, defensive cybersecurity, multi-platform development, performance, scalability, software library design, software testing, technical documentation.
* **PM/TPM:** [Confluence](https://www.atlassian.com/software/confluence), [Jira](https://www.atlassian.com/software/jira), building platforms, collaboration, coordination with downstream services, cross-collaboration (dozens of teams, hundreds of services), organization of complex projects, product development, product roadmap management, project documentation, project management, technical documentation.

## Groups and Accomplishments

* U.S. patent filing, [“System and Methods for User Authentication across Multiple Domains”](https://patents.google.com/patent/US20160241536A1/en?inventor=Ryan+Parman) (US15042104) (2016)
* U.S. patent filing, [“Hive-based Peer-to-Peer Network”](https://patents.google.com/patent/US8103870B2/en?inventor=Ryan+Parman) (US8103870B2) (2007)
* Voting representative for AWS, [PHP Framework Interoperability Group](http://www.php-fig.org) (2012–2013)

## Education

Obtained a **Bachelor of Arts** degree in *Design and Visualization* from *Silicon Valley College* (now [Carrington College](http://carrington.edu/schools/san-jose-california/)) in San Jose, CA.

Graduated in *November 2003* with a **3.84** GPA.