

Jamie Starke

Victoria, British Columbia • resume@jamiestarke.com • github.com/jrstarke

I enjoy thinking big, planning for scalability, and looking at ways to harden systems against attacks. Infrastructure as Code and Deployment Automation help me achieve these quickly through consistent and predictable environments. Making infrastructure spending more efficient and increasing performance are challenges I enjoy.

TECHNOLOGIES

- Preference for and proficiency with *Amazon Web Services*, (such as *CloudFormation*, *CodeBuild*, *CodePipeline*, and more), *Terraform*, *TypeScript*, *JavaScript*, *Python*, *IntelliJ*.
- Competency with *Terraform*, *Jenkins*, *Chef*, *JavaScript*, *Groovy*, *Visual Studio*, *Eclipse*.
- Familiarity with *C#*, *Java*.
- Interest in Cloud Architecture in Public Cloud Services, Security, Deployment, and Automation.

RELEVANT EMPLOYMENT HISTORY

Software Dev. Engineer, DevOps **Workday** *January 2017 - current*

- Designed and implemented a solution to facilitate and easily configure migration of data between a variable set of databases, using terraform and AWS Database Migration Service.
- Created a reusable Terraform module for managing an EC2 Container Service Cluster and ensuring zero downtime during instance modifications.

DevOps Engineer **Giftbit** *June 2016 - January 2018*

- Achieved a consistent Infrastructure as Code model for the infrastructure of our newer Lightrail product using CloudFormation. New resources and permissions can be deployed through our development, staging and production environments by performing a pull request which triggers a CodePipeline to consistently deploy and configure each environment.
- Created a fast and consistent way for developers to deploy service changes using deployment scripts, reducing deploying new functionality from hours to minutes.
- Devised and implemented new security protocols to ensure the security of application secrets.

Lead Full-Stack Developer **SilkStart** *May 2014 - May 2016*

- Built a consistent model of our servers using Chef recipes and configurations to automate setup, configuration and deployment of our infrastructure, enabling us to easily replace servers in a matter of minutes. This also support easily switching infrastructure providers.
- Transitioned our infrastructure from Linode to Amazon Web Services to save about \$500/month over a two year period and gain access to advanced functionality.
- Developed a continuous integration pipeline using Jenkins, Docker, Packer and Chef to automate testing consistently and facilitate automated deployment to a staging environment.

SELECTED PROJECTS

- **Activity Aware** github.com/giftbit/activity-aware-ids-aws
Information Aggregation Framework. Initial use case alerts on Authorization Failure and Access Denied events within an AWS environment. Initial release supported incoming data from CloudTrail Logs and outgoing to Slack.

SELECTED RECENT WRITING

- **Activity Aware IDS for AWS: The simple way to stay aware of activity in your AWS Account**
February 2017
Description of the Principle of Least Privilege, The Shared Responsibility Model, and how Activity Aware can support both. <https://www.jamiestarke.com/2017/02/27/activity-aware-ids-aws-simple-way-to-stay-aware-activity-in-our-aws-account/>

EDUCATION

Master of Science in Computer Science from the University of Calgary