



Andy Reitz

(408) 480-4327

andrew.reitz@gmail.com / <http://redefine.dyndns.org/~andy>

San Francisco, CA

Profile

I am a generalist with nearly 20 years of experience in a variety of roles, who enjoys working with UNIX systems, networking, and programming. I am seeking a full-time position in the Developer Productivity, Release, or DevOps areas.

Skills

UNIX systems including Linux, macOS, FreeBSD, and Solaris

Programming in Ruby, Python, shell, Perl, Java, and C

Data networking including TCP/IP, Ethernet, VLAN, Routing, Firewall, NAT, and DHCP

Tools including Nagios, Jenkins, Review Board, puppet, git, Confluence, Jira, GitHub

Enterprise, AWS, and Pants

Experience

STRIPE INC.

SAN FRANCISCO, CA

1/2016 - PRESENT

In my 6+ years at Stripe, I served as a senior engineer focused on ensuring Stripe's engineers could write code with velocity and safety. I started off helping to maintain Stripe's Continuous Integration (CI) system, developer environment, and source control (SCM) systems. Over time, as the company grew, I focused more on Stripe's core SCM and Code Review tooling and the integrations between these systems and CI.

I joined Stripe as part of an experimental batch of SREs, but my role and body of work have been indistinguishable from other software engineers at the company. During my time at Stripe, I helped to maintain and push forward various tools used by engineers internally. I helped the company grow by completing over 450 interviews, writing design documents, managing projects, managing customer expectations, working across teams, mentoring junior engineers (including several interns), and partnering with recruiting on improving hiring practices. In the last year, I've focused on building a team around code storage, search, and review. This initiative involved mentoring new members of the team, setting the overall direction, and leading the scoping, design, and implementation of projects. In addition, I owned Stripe's relationship with GitHub, participating on their customer advisory board, negotiating contracts, and directing Stripe on the best ways to leverage GitHub's software.

For over 5 years, I was primarily responsible for the maintenance, performance, and reliability of the GitHub Enterprise Server at Stripe. I lead the development of Stripe's first Merge Queue, which enabled engineering velocity to be maintained as Stripe grew. I was heavily involved in developing Stripe's processes around code storage and code review. As a key stakeholder in Stripe's code review system, I was able to lead the development of several projects focused on improving code review processes, including the development of a system that could automate reviews when we could statically assert that a proposed change was safe. I also directed the migration of over 600 git repositories from github.com to an internal GitHub Enterprise Server instance.

TWITTER INC.

SAN FRANCISCO, CA

7/2011 – 10/2015

SOFTWARE ENGINEER, BUILD TEAM

4/2012 – 10/2015

In April of 2012, I joined a newly formed team at Twitter focused on developer productivity. During my time on this team, I performed a variety of duties, ranging from maintaining internal git and Review Board services, maintaining our use of maven in a large scala repository, and helping to maintain *Pants* (Twitter's homegrown build tool). Along the way, I worked on projects to convert said scala repository from Maven to Pants, performed critical integration work to upgrade Pants, and worked on getting Twitter into a monorepo, with a true monobuild. I also wrote documentation, developed training courses (that I frequently delivered inside of Twitter), and provided excellent support to our customers. During this time I worked on several side projects, including revamping Twitter's internal people directory which included adding a feature to support employee recognition.

SOFTWARE ENGINEER, RELEASE TEAM

7/2011 – 4/2012

As a release engineer at Twitter, my team was responsible for daily deploys of Twitter's frontend Ruby on Rails application. This role entailed not only performing deploys but also performing light QA, site reliability functions, and on-call support. I also helped maintain several in-house tools (written in everything from Ruby, to Rails, to Java). In addition, I was part of a small team that built out a new test environment and built a complete copy of Twitter from scratch. This effort involved learning how to install and configure every layer of Twitter's stack, including discovering service dependencies, rewriting config files, and figuring out how to bootstrap the system while including enough data for the test site to be useful.

POPSUGAR INC.

SAN FRANCISCO, CA

4/2008 – 6/2011

For two years, I was the sole dedicated member of the operations staff at PopSugar Inc. I was in charge of all aspects of the operation of PopSugar's web properties. This included maintaining two different software stacks (one Java/Tomcat, the other PHP/Lighttpd) in two separate managed facilities. I built and/or maintained the following: alert monitoring, performance monitoring, on-call support, build/release, MySQL administration, backups, and tools for development. Notable projects included: building an environment that passed PCI DSS audit, a full data center migration, and building alert monitoring from scratch.

AGGREGATE KNOWLEDGE

SAN MATEO, CA

9/2006 – 4/2008

As a Systems Administrator and Developer I helped to build out Aggregate Knowledge's web infrastructure, including managing Linux and Solaris machines, performing code releases, building networks, managing in-house IT resources, and automation programming. Notable projects included: building a new co-located facility from scratch, crafting a flexible on-call rotation system, writing custom Nagios plugins, and implementing custom statistics gathering/graphing scripts.

EDS

SUNNYVALE, CA

4/2003 – 8/2006

As a Field Engineer in the Automated Operations business unit at EDS (formerly Loudcloud, Inc.), I provided advanced support and maintenance of a state-of-the-art operations system. In this role I supported various legacy systems, including an automated

