

Alexander Wauck

DevOps Software Engineer and Systems Administrator with a great love of automation and an intense focus on security and correctness.

Technical Proficiencies

- Languages: Python, Go, bash, C, C++, Rust
- APIs/Frameworks: gin-gonic, Flask
- Databases: PostgreSQL, MySQL, Cassandra, MongoDB
- Operating Systems: Debian/Ubuntu Linux, Red Hat Enterprise Linux/CentOS
- AWS Services (primarily EC2, RDS, S3, CloudFront, and Route53)
- Ops/Deployment Technologies: Salt, Ansible, OpenShift/Kubernetes, Docker, Jenkins
- Miscellaneous Software and Related Skills
 - Source control: Git, Mercurial, Subversion, CVS
 - Multimedia production and file formats
 - Software reverse-engineering

Work Experience

Exosite

Minneapolis, MN

November 2015-June 2018

Promoted to Senior DevOps Engineer in May 2017.

At Exosite, I worked as a Senior DevOps Engineer. This means that I kept the company's more than 100 EC2 VMs (including an OpenShift Origin cluster) running, along with many other services both in AWS and outside of it. I was highly focused on increasing automation to improve Ops productivity and make management of a large number of VMs and other resources by a small team tractable. I also helped Exosite acquire ISO 27001 certification.

CodeWeavers

St. Paul, MN

March 2015-October 2015

CodeWeavers develops a Wine-based solution for running Windows software on non-Windows platforms (e.g. Linux, Mac OS X) and offers software porting/development services. I wrote security-oriented software for a major defense contractor that subcontracted some work to CodeWeavers.

FPX

Bloomington, MN

May 2014-March 2015

I worked on an acquired product formerly called Glider (now Contract Management by FPX), using Python and Django, among other technologies. I worked on both the frontend (using Flight, Bootstrap, and jQuery) and the backend (using Python and Django), coordinating between the two via REST API. This included some non-trivial frontend work implementing a highly interactive feature on top of a third-party component not designed for the task.

Thomson Reuters

St. Paul, MN

March 2013-May 2014

The part of the company I worked for came from the Reuters side; it provides a batch-oriented financial data distribution service.

- Wrote a deployment test suite for a component that previously lacked such
- Tweaked database queries to retrieve additional data sets
- Implemented new functionality (e.g. more detailed usage tracking) in a server-side C++ program

Intradyn, Inc.

Eagan, MN

August 2012-March 2013

Intradyn, Inc. produces Linux-based email archiving appliances. I developed and maintained the software for these appliances.

- Maintained and improved a complex software system written in Python, Bash, Clojure, Erlang, and C++, using Django, PostgreSQL, and RabbitMQ
- Wrote scripts to automatically fix or enhance PostgreSQL databases and RabbitMQ queues in a fail-safe manner on systems I cannot access directly
- Diagnosed and resolved the occasional customer problem

Open Systems International, Inc.

Medina, MN

September 2010-August 2012

Open Systems International, Inc. produces control and monitoring (SCADA) software, primarily for electrical utilities. I was a software developer maintaining and developing three central pieces of user-facing software.

- Maintained and improved complex software written in C#
- Worked around the occasional bug or quirk in Win32
- Reverse-engineered legacy software
- Monitored compile process with occasional manual intervention
- Occasional maintenance of simple software written in C
- Proposed improvements to software development tools and process
- Developed and maintained a small web application using Python and Django
- Wrote a specification and unit tests for functionality that sorely needed such

Google Summer of Code

St. Peter, MN

Summer 2008, Summer 2009

I spent two summers working on the Common Print Dialog as part of the Google Summer of Code. I was jointly mentored by Jonathan Riddell of KDE and Till Kamppeter of the Linux Foundation, and the specifications for the layout and function of the dialog were developed by Peter Sicking of m+mi works. The dialog was implemented in C++ using the Qt and KDE libraries and the CUPS API. Tasks:

- Maintain contact with foreign collaborators
- Implement user interface specification
- Implement Dbus API

Education

Princeton University, September 2009 – June 2010 (Graduate studies)

Gustavus Adolphus College, St. Peter, MN, May 2009, Summa Cum Laude
BA, Dual Majors: Computer Science (GPA 3.807), Mathematics (GPA: 3.71)