Tim Wilde

Systems, software and network engineering triple-threat

Boston, MA
tim@krellis.org
github.com/twilde

PROFILE

A self-driven, passionate problem solver and engineer with a career spanning systems, software and network engineering. I don't shy away from tough technical issues; I run towards them, ready to get my hands dirty across the stack to find and solve a problem. I've started my own business and worked for companies small and large, and I've spent my career communicating technical information across organizations and to customers, technical and non-technical.

20+
years of BSD and Linux system engineering

20+
years of software engineering: Python, Perl, C++

15+
years operating and debugging TCP, UDP and IP, including IPv6

SKILLS

Languages	Python (primary) · Perl · C++ · Go (working familiarity) · HTML / JavaScript / CSS bash / sh
Reliability	SLI / SLO design and ownership · Incident command · Blameless postmortems Large-scale resource scheduling · Capacity management · Prometheus · Grafana
Systems and cloud	BSD and Linux · Borg · Kubernetes concepts · AWS and GCE · VMware ESX/ESXi Citrix XenServer · Redis
Networking and DNS	TCP / UDP / IP · IPv6 · BIND · Cisco routing and switching DoS prevention and mitigation
Services and data	Apache and nginx · MySQL and PostgreSQL · Microservices architecture

EXPERIENCE

Google

Sep 2012 – Present

Cambridge, MA

Staff Site Reliability Engineer — Oct 2022 – Present
Senior Site Reliability Engineer — Dec 2015 – Oct 2022
Site Reliability Engineer — Sep 2012 – Dec 2015

Systems engineer on the scalability and reliability of services including Search, Ads, News, AMP, Flights and Shopping. I work closely with Borg, Google's job scheduling and management system, and specialize in deep-dive debugging of production systems.

Search ML SRE — Late 2025 – Present

Reliability and SLO compliance for Google Search serving, including the systems behind AI Overviews and AI Mode, which must stay highly available around the globe. Includes large-scale resource scheduling, capacity planning and production support on one of Google's largest fleets.

Tech IRT — 2020 – Present

A member of Google's top-tier technical incident response team: a key role in large-scale incident response across Google's infrastructure, jumping into unfamiliar systems under pressure and coordinating across teams to minimize impact on partners and users.

Ads product-area resource management — 2020 – 2025

Resource tech lead for the whole Ads product area's fleet, managing technical policy and capacity. Identified and drove Ads-wide adoption of a new caching technology that delivered durable operating-expense savings; the approach became a template later adopted by other product areas.

Shopping SRE 2018 – 2020
Drove a project to migrate Shopping off dedicated machines and onto more efficient shared Borg resources, reducing its dedicated-machine footprint and realizing real savings, without disrupting a live, high-traffic product.

Travel Hotels SRE 2014 – 2018
Founding member and tech lead of the team, helping stand it up from scratch and instill a strong reliability culture from day one. The Hotels product crossed a substantial revenue milestone during this period.

Blunderbuss 2013 – Present · 20% project
An unfunded bug-assignment and notification system relied on by thousands of Google teams. I've been involved since 2013 and tech lead since about 2017, keeping it running reliably for years with no formal funding or dedicated support: a case study in high-leverage, low-overhead reliability engineering.

Team Cymru, Inc.

Apr 2007 – Sep 2012

Systems and Software Engineer · Chicago, IL and Orlando, FL

Software Engineer	Jul 2009 – Sep 2012
Senior Developer	Feb 2008 – Jul 2009
Researcher	Jul 2007 – Feb 2008
Consultant	Apr 2007 – Jun 2007

Valuable experience and insight into the world of information security. I developed a large-scale query tool that was an early introduction to Big Data, and had primary responsibility for building, organizing and managing a variety of bare-metal and virtualized systems, deployed both in our own datacenters and on partner premises.

Dynamic Network Services, Inc.

Nov 1998 – Aug 2006

Founder, System Administrator, & more · Worcester, MA and Manchester, NH

I founded DynDNS and built it into a multi-million-dollar-a-year business with well over a million customers, 100% self-funded at the time of my departure. I wore all of the hats at various times, with primary focuses on business development and overall system and network design and implementation, and was directly involved in the full systems, network and software design processes from concept to implementation.

I even accidentally created a de facto standard: DynDNS's "/nic/update" HTTP(S)-driven dynamic DNS record update protocol is still in use by dynamic DNS providers today, and it makes me cringe!

HOW I WORK

Run towards the hard problem
I specialize in deep-dive debugging of production systems and the code that runs on them, and I'm happiest jumping into something unfamiliar when it matters.

Automate the toil, reuse what exists
Reuse existing and off-the-shelf solutions where possible, and automate repetitive work, so teams can spend their time on engineering rather than operations. I'm interested in where AI can help with that, with care about what it is allowed to change in production.

Let SLOs decide where effort goes
Carefully crafted SLOs balance business needs, operational realities and competing goals like reliability and velocity, and they say what to focus on versus what can safely wait.

Leave the practices behind
I set best practices on the teams I join, through example as much as through documentation and policy-setting, so good operational habits outlast any one incident.

References available upon request.