# BOB BYERS

189 LOWELL ST • ANDOVER, MA 01810

978 758-4301 • BOB.BYERS1111@GMAIL.COM


## PROFESSIONAL SUMMARY


Developer with strength in writing automation. Python, Perl, tcl/Excpect, Selenium, git, third-party testing APIs. Also adept with AWS with Python (boto3). Writes robust, readable and maintainable code and mentoring junior team members.

Accomplishments: Sole developer of a test automation framework, contributed many enhancements to others. Wrote network traffic generation tools filling gaps in APIs of vendor tools. Instrumented advanced test environments to catch memory leaks, race conditions and other defects early in the release cycle.


## CORE COMPETENCIES

- Highly robust, readable and maintainable code
- Broad understanding of data communication principles
- Excellent technical communication skills
- Competent in: Python, perl, tcl/Expect, Linux, Selenium, C, bash, AWS, boto3, packet generation and analysis, TCP/IP, SIP, OSPF, HTTPS, TLS, git, Perforce, SVN, Jira, Bugzilla
- Can-do attitude to find solutions using existing tools
- Deep investigator of complex problems
- Wide-ranging experience with computer science topics
- Go-to resource for in-house test framework
- Strong relationship with developers: built confidence that defect reports were accurate and non-trivial
- Continual Learner

## PROFESSIONAL EXPERIENCE

Akamai, Cambridge, Massachusetts 2016-2018

**Senior SQA Engineer**

- Developed new test automation suites per defined test cases
- Enhanced existing automation suites accommodating new product functionality
- Led code reviews for new and enhanced automated test suites
- Incorporated under-used aspects of automation library to achieve highly robust test scripts
- Coded run-time decision making into test suites to deal with changing environments
- Created secure key management tool to synchronize ssh keys across >100 servers
- Created Jira tools utilizing the Jira API

Oracle, Bedford, Massachusetts 2012-2016

(originally Acme Packet, acquired by Oracle 2013)

**Senior SQA Engineer**

- Wrote readable and maintainable test automation for SIP based VoIP products

- Wrote test suites for GUI testing
- Leveraged APIs of traffic generation tools to simulate real-world data traffic patterns and loads
- Improved existing automation tools
- Created tools for various needs of the SQA team
- Mentored junior team members
- Led code reviews of automation scripts
- Developed new test plans
- Investigated test failures
- Submitted CRs for product defects
- Revised network monitoring tool for generating lab outage alerts

Arbor Networks, Chelmsford, Massachusetts 2008-2012

**Principal SQA Engineer**

- Developed automated test suites
- Owned SQA life cycle for key product features
- Provided SQA input during product requirements definition stage
- Provided SQA project scoping for product release scheduling
- Wrote test plans utilizing traffic generation tools to simulate real-world, very complex data traffic.
- Wrote tool to modify packet captures (remove encapsulation, change IP Addresses, convert IPv4 headers to IPv6, add GTP headers, etc.
- Composed tool for creating traffic patterns to be replayed through the DUT

Ellacoya Networks, Merrimack, New Hampshire 2001-2008

**Principal SQA Engineer**

- Developed automation library
- Led testing DPI/subscriber management solution for tier-1 ISPs
- Tested, wrote test plans and automated strategic new product features
- Reviewed and provided feedback for product requirements and functional design
- Solved issues on-site at customer and beta sites
- Mentored junior team members

IronBridge Networks, Lexington, Massachusetts 2000-2001

**Senior SQA Engineer**

- Developed automation for testing of routing protocols (BGP, OSPF)
- Wrote test plans and held test plan reviews

Nortel Networks, Billerica, Massachusetts 1998-2000

**Principal Support Engineer**

- Provided expert Level 3 escalation support to premium customers of Nortel's router, switch and net management products
- Traveled to customer sites for troubleshooting network problems

- Acted as prime support contact for three large, strategic customers

Harvard School of Public Health, Cambridge, Massachusetts 1994-1998

**Network Engineer / SAS Programmer**

- Wrote SAS applications for statistical analysis and data management in public health research
- Designed LAN for new research building.
- Interfaced with faculty and administration on data network needs
- Found solutions to technical obstacles for providing connectivity to remote offices
- Recommended new capital expenditures for network infrastructure

## COURSES, WORKSHOPS & SEMINARS

edX, coursera 2016-2018

Partial list:

- **AWS Developer: Building on AWS** (https://courses.edx.org/courses/course-v1:AWS+OTP-AWSD1+1T2018/course/)
- **Version Control with Git (Atlassian)** (https://www.coursera.org/learn/version-control-with-git)
- **Python for Data Science** (https://courses.edx.org/courses/course-v1:UCSanDiegoX+DSE200x+1T2018/course/)

Harvard IACS (Institute for Applied Computational Science), Cambridge, Massachusetts 2016-2018

Frequent workshop attendee. Partial list of attended sessions:

- ComputeFest
- Advanced Python
- Data Visualization
- Python for Scientists and Engineers

## EDUCATION

**Certificate in Applied Science (CAS), Harvard Extension School**, Cambridge, MA

The CAS consists of a minimum of 8 graduate level courses and is as challenging as a master's degree from many graduate schools. Courses completed:

- Compiler Design (two semesters, wrote a compiler)
- Computer Hardware Architecture
- Advanced Topics in Data Networking
- Internet Architectures and Data Communications
- Operating Systems
- UNIX Systems Programming

- TCP/IP Principles
- Data Structures
- C++

**Bachelor of Arts,** Mathematics, INDIANA UNIVERSITY, Bloomington, IN

## OTHER FACTS

### Musician

- Spent time as a musician (trombone), arranger and composer
- Performed jazz and classical music with combos, swing bands, brass quartets, orchestras, bands, pit orchestras, etc. throughout New England and Europe

### Favorites from the Zen of Python

- Simple is better than complex
- Readability counts
- If the implementation is hard to explain, it's a bad idea
- Errors should never pass silently