



IEEE CS 28th Software Technology Conference

Tim Weil, *Alcohol Monitoring Systems*

The 2017 IEEE Computer Society's 28th Software Technology Conference (STC) brought together software practitioners from industry, government, and academia, providing a forum for them to share their opportunities, challenges, and successes. At the STC program, the exchange of ideas, technologies, data, and best practices gave attendees insight, methodologies, and tools for leading software development and business innovation in the areas of national defense, critical infrastructure, IT, and global markets. This year's conference featured presentation tracks, tutorials, and keynote presentations in the areas of software cybersecurity, systems engineering, metrics, agile, test and verification, program management, the Internet of Things (IoT), and cloud computing.

This was the first STC program I attended, and I'm here to say that the quality of this conference will have me in attendance for years to come. Previous STC programs

(2002 to 2015) have highlighted software technology initiatives of stakeholders from the US Department of Defense (DoD), Carnegie Mellon University's Software Engineering Institute (SEI), the Capability Maturity Model Integration (CMMI) Institute, supporting contractors, businesses, and academia (see conference.usu.edu/stc/proceedings.cfm).

This year's program, hosted by NIST in Gaithersburg, Maryland, added a new dimension to the STC agenda. With the support of NIST's Software and Systems Division (SSD; www.nist.gov/itl/ssd) and Computer Security Division (CSD; csrc.nist.gov/about), the conference also presented national testbeds for software vulnerability, quality, testing, validation, and defect management. Engineers, scientists, and executives from NIST's SSD and CSD programs provided an in-depth look at testing, cryptography, risk management, and IoT that made the quality of this conference an exceptional experience.

This year's distinguished presenters included the following:

- James Thompson, director, Deputy Assistant Secretary of Defense for Systems Engineering, DoD
- Barry Boehm, director, Center for Systems and Software Engineering, University of Southern California (USC)
- Gregory Touhill, president, Cyxtera Federal Group
- Kirk Botula, CEO, CMMI Institute
- Jefferey Voas, NIST
- Barry Horowitz, University of Virginia
- Anita Carleton, software solutions deputy director, SEI
- Rear Admiral James Downey, Deputy Commander, Surface Warfare Commander, Navy Regional Maintenance Center
- Rob Gold, director, Engineering Enterprise at DoD
- Bobbie Stempfley, director, CERT Division at SEI

Let's look at some highlights from these and other presentations.

Table 1. Software and the complexity of security issues.

Aircraft	Era of deployment	Lines of code (LOC)	Complexity (software security)
Wright Flyer	1903	None	Low
Apollo 11 Lunar Lander	1968	2,000	Low
SR-71 Blackbird	1964–1999	500,000	Low
Eurofighter Typhoon	2003–2017	1.5 million	Medium
F-22	1996–2011	1.7 million	Medium
A340	1991–2011	2 million	Medium
Rafale F3	1986–present	2 million	Medium
B-787 Dreamliner	2011–present	14 million	High
F-35 Joint Strike Fighter	2006–present	24 million	High

A Few Highlights

Three software projects captured my attention at the conference. In terms of complexity, functionality, and security issues, Carol Woody’s talk—“Security Measurement: Building Confidence that the System Is Secure”—introduced this topic. How can we establish reasonable confidence that the security for a system will meet its operational needs? The first challenge is to establish that the requirements define the appropriate security behavior and the design addresses these security concerns. The second challenge is to establish that the completed system, as built, fully satisfies the specifications. Measures to provide assurance must, therefore, address requirements, design, construction, and test. Software, as a major part of every system, typically handles more than 80 percent of the functionality, and software is never defect-free. Table 1 raises the issue of lines of code as a metric for the complexity of systems.

Rick Kuhn served as conference co-chair. He is a computer scientist with NIST’s CSD. Kuhn presented a paper from Eric Wong’s Software Assurance group from the University of Texas at Dallas. This paper provided another view of aviation software challenges—“Automated

Test Generation for High MC/DC Using Guided Concolic Testing.” Software, in many different fields and tasks, has replaced humans to improve efficiency and safety. However, devastating accidents with human casualties can be caused by implementation bugs and design defects in safety-critical software such as airplane navigation and autopilot systems. On 20 December 1995, American Airlines Flight 965 departed from Miami, Florida, for Cali, Colombia, and crashed into a 9,800-foot mountain due to a navigation software error, causing 159 deaths. Such tragedies are fortunately extraordinarily rare. To prevent them, the US Federal Aviation Administration requires that Level A (life-critical) systems meet the modified condition/decision coverage (MC/DC) criterion defined in DO-178B/C. However, this form of testing can be seven times the cost of conventional testing. Wong’s group has focused on reducing this cost.

In the 1990s, NIST was given the task of finding a replacement for the legacy Digital Encryption Standard with the next generation of cryptographic algorithms. That assessment eventually produced the Advanced Encryption Standard (AES), also known as Rijndael. A talk on

“Combinatorial Coverage Measurement of Test Vectors in Cryptographic Algorithm Validation” also introduced the Cryptographic Algorithm Validation Program at NIST, a joint American-Canadian security accreditation program, providing guidelines for validation testing of NIST-recommended cryptographic algorithms (see bit.ly/2x62bHJ). NIST’s Cryptographic Algorithm Validation System was used as a reference system against which AES implementations were tested. Applying combinatorial coverage measurement identified subsets of tests that are less thorough than others, but, fortunately for users, no errors were found in the official test suites.

STC Panels and Keynotes

A strong set of keynote and panel presentations covered contemporary issues facing the software industry:

- “Software Acquisition: Facing the Challenge, Valuing Velocity,” by James Thompson (bit.ly/2x4OGYR)
- “Software Cost Estimation Meets Software Diversity,” by Barry Boehm (bit.ly/2ywbUIP)
- “The Future of Defense Software Systems (Mythbusters Style),”



Figure 1. At the IEEE CS Software Technology Conference at NIST. Barry Boehm (SMHS Class of '53), 2nd from the left, and the author, Tim Weil (SMHS '70), 2nd from the right, share their Santa Monica High School roots. The photo also shows Rick Kuhn of NIST (far right) and Dr. Bhuvan Helkar (far left) of the University of South Florida.

by Anita Carleton and panelists (bit.ly/2x3hErP)

- “Cybersecurity for Computer-Controlled Physical Systems,” by Barry Horowitz (bit.ly/2ywE0Fz)
- “Demystifying the Internet of Things,” by Jeffery Voas (bit.ly/2x4Ytyd)

For a complete listing with the keynote presentations and briefings presented, see conference.usu.edu/stc/Schedule/grid.cfm.

See You at STC 2018

As a practitioner, writer, and sometime educator in the field of *vehicular networks*, I was given the opportunity to present a tutorial at STC on *vehicular public-key infrastructure* (“VPKI Hits the Highway,” bit.ly/2yuu6Ta). My presentation highlighted the complexity of the US Department of Transportation’s Connected Vehicle security infrastructure, known in the industry as the Security Credential Management System (bit.ly/2ilIZmA). One of the highlights of the tutorial was a

chance to meet with Barry Boehm, director emeritus of the USC Center for Systems and Software Engineering. As an STC presenter, Boehm is renowned in the software industry for his contributions to the field, including the Constructive Cost Model, the Spiral Model of the software process, the Theory W (win-win) approach to software management and requirements determination, the foundations for the areas of software risk management and software quality factor analysis, and two advanced software engineering environments: the TRW Software Productivity System and the Quantum Leap Environment. As it turns out, we are both graduates of Santa Monica High School, and we had an opportunity to chat as former SMHS alumni (Figure 1). It was a real pleasure to meet Barry, a thought leader in our industry.

I would like to thank the STC organizers, presenters, and the attendees at this year’s tutorial

on vehicular networks. My hope is that the interest and momentum from these presentations will continue in the conference program over the next few years. In particular, I would encourage readers to visit our research portal, VPKI Hits the Highway (securityfeeds.com/vpki.html). Over the past 10 years, I have maintained this aggregation website as a gateway to national and international vehicular networking pilots, standards, research, publications, and commercial products. □

Tim Weil is a network project manager with Alcohol Monitoring Systems in Littleton, Colorado. His interests include service management for vehicular networks using WAVE. In the areas of vehicular networks, Weil’s work includes the IEEE 1609 (WAVE) standards, US DOT VII/Intellidrive and Connected Vehicle programs, and being an author and speaker on topics in security for vehicular networks. He is an industry-certified security professional (Certified Information Systems Security Professional/Certified Cloud Security Professional, Certified Information Systems Auditor, Project Management Professional), past chair of the IEEE Denver Communication Society Chapter, a senior member of IEEE, and security editor for IT Professional. Contact him at trweil@ieee.org.

Read your subscriptions through the myCS publications portal at <http://mycs.computer.org>