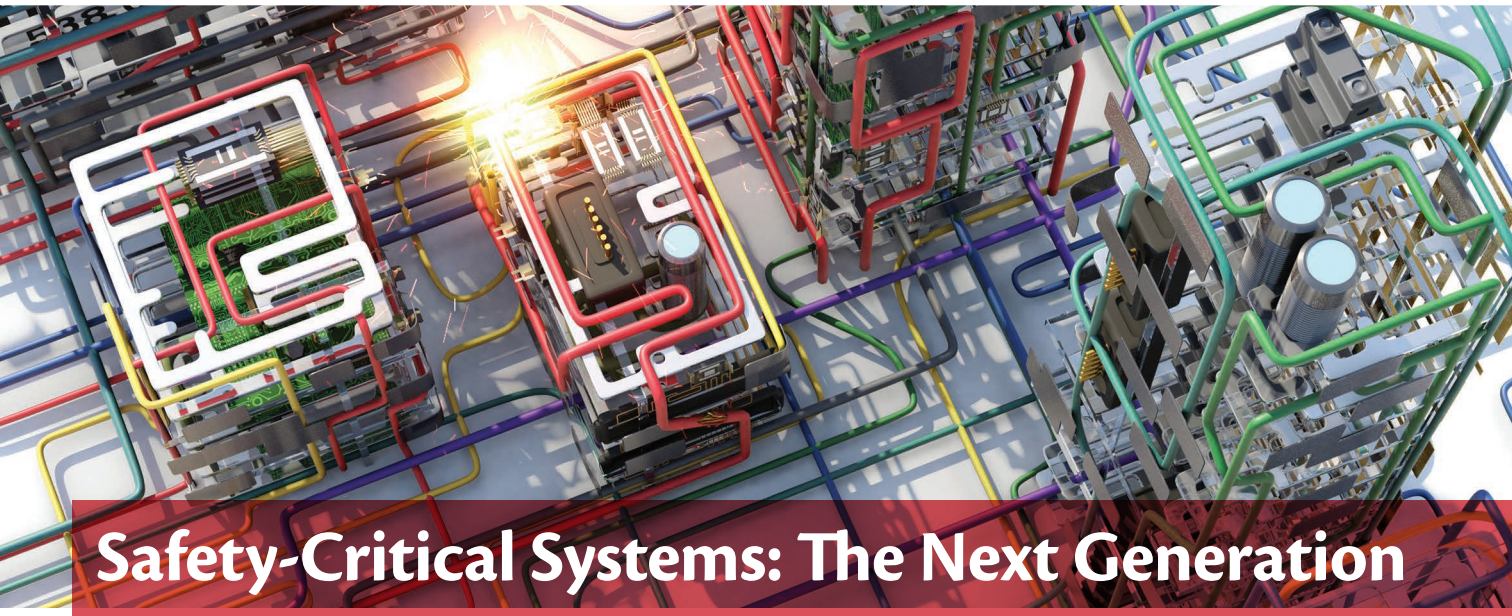

Safety-Critical Systems: The Next Generation

Robin Bloomfield | Adelard
Jay Lala | Raytheon Company

Safety-critical computer-based systems are woven into the fabric of our lives. These systems must work adequately given user behavior, system interactions, changing environments and expectations, organizational turbulence, regulatory caution, routine component and operator failure, the complexity of international projects, and adaptation and refurbishment as well as security-related issues such as intentional malicious attacks and supply-chain risks.

Safety and Security

Apart from the fact that safety-critical systems is an important topic in its own right, *IEEE Security & Privacy* is addressing these issues for two other reasons. First, the magazine's remit is much broader than "security and privacy." Its tagline of "Building Dependability, Reliability, and Trust" reflects that we are partially owned by the IEEE Reliability Society and have a broad interest in trust and dependability. The other reason is that safety will be an increasingly relevant application area for security and privacy specialists. These days, air gaps and isolation are seldom credible arguments for security—the US Department of Homeland Security found, on average, 11 connections between SCADA and enterprise systems.[1] Thus, we can't consider any computer-based safety system to be truly safe unless we also address its security.

Both safety and security aim to protect something. Broadly speaking, safety is concerned with protecting the environment from the system, whereas security is concerned with protecting the system from the environment. The issue is how to ensure that the protection is adequate. A classic view of safety is that it's concerned with preventing accidents by identifying potential weaknesses, initiating events, internal hazards, and potentially hazardous states, and then identifying and applying appropriate mitigations to reduce the risks to a tolerable level. Security

1540-7993/13/$31.00 © 2013 IEEE Copublished by the IEEE Computer and Reliability Societies

is concerned with protecting assets against internal and external threats and vulnerabilities that compromise them using controls that reduce the risk of compromise to an acceptable level.

The next generation of safety-critical systems includes not only the rather obvious application areas such as air traffic management, nuclear power plant control, and military systems but also networked patient care, driverless cars, autonomous air vehicles, and personal apps. Undoubtedly, there will be new technologies for building and assuring these systems as well as the adaptation and evolution of tried and tested approaches.

In This Issue

A key part of any safety culture is learning from experience. Although medical devices bring many benefits, they do so at a human price in terms of avoidable deaths. Yet learning from incidents and accidents is hard: we must view the accounts and analyses through the lenses of those who did the original collection as well as the technical and often legal factors that shaped their narrative and analysis. Single catastrophic accidents require inquiries with senior academics and judges to ascertain the substantive lessons learned (Diane Vaughan and *Challenger*[2] and Sir Charles Haddon-Cave and *Nimrod*[3] disasters come to mind). Yet, how and what are we to learn from the myriad everyday tragic and near misses?

The article "Analysis of Safety-Critical Computer Failures in Medical Devices," by Homa Alemzadeh and her colleagues, investigates the safety implications of the emerging trend of embedding computers in medical devices. The article provides a fascinating description of a thorough methodology for assessing experience with medical devices. The authors also show how hard it is to ascertain what one might initially think are straightforward facts.

One of the shocking revelations from their article is that malfunctioning medical devices are a major cause of serious injury and death in the US. Between 2006 and 2011, over 5,000 recalls and more than a million adverse events were reported to the US Food and Drug Administration (FDA). Nearly 23 percent of these recalls were due to computer-related failures, and the vast majority presented serious health consequences of injury or death to patients.

Software failures remain the major cause for recalls of computer-based medical devices, but hardware, battery, and I/O failures are also significant contributors, suggesting that designs with greater fault tolerance would benefit patients by reducing the need to remove failed devices.

Other evidence from the FDA points to several hundred deaths per year in the US alone from infusion pumps. These are shocking figures, and it's not surprising that the FDA and the industry have major initiatives to address them.

Time and again, accident investigations call for clear responsibilities and for those developing and using the devices and system to understand what they're doing. It's not that more paperwork, bureaucracy, and commoditization are required; what's needed are better engineering and better communication.

Some possible approaches for better engineering are described in "Verifying Cyber-Physical Interactions in Safety-Critical Systems," by Sayan Mitra, Tichakorn Wongpiromsan, and Richard Murray. The article examines the safety aspects of cyber-physical systems. As computer hardware and software are embedded in everything from motor vehicles to power plants to military sensors and weapons, the interactions between cyber and physical components can create novel safety challenges.

The authors use the mission planning, navigation, and control system of an autonomous vehicle designed for the DARPA Urban Challenge as a case study. The article also demonstrates how to learn from experience. In the third round of the 2007 DARPA Urban Challenge, the autonomous Ford Econoline van deviated from the computer-generated path and began to stutter in the middle of a busy intersection. Earlier in the competition, the van had successfully driven on- and off-road, parked, merged with traffic, and undertook U-turns all with no human driver and in accordance with traffic rules.

This article examines why the error occurred and describes some of the modeling and analysis techniques that are needed to gain assurance of this type of autonomous system.

There are several highlights. The first is the development of precise claims about the system's behavior, drawing on the security properties world to define properties in terms of "safety" and liveness, or their combination. Another is the discussion of the advances in safety verification and analysis tools. Overapproximating the vehicle's trajectory and finding properties that are always true (types of invariants) are key approaches that are sometimes tractable with present tools. The authors relate these advances to help leverage current technologies, such as simulation. Although the article has some mathematical notations, please don't put it off but persevere in reading it—the underlying ideas are important and can be grasped at different levels.

Although we're perhaps more cautious than the authors about how automatic safety analyses will become, we agree that analysis that exploits advances in modeling and formal verification will have an important role to play. But, as with all such efforts, we're left with assessing the trustworthiness of the tools and epistemic doubt in the validity of the models.

Whereas the first two articles address the need to learn from experience and for rigorous safety analysis, the third article examines a specific method for challenging systems. Independent challenge is an important part of building confidence and is formalized in, for example, the UK requirements for statistical testing and static analysis of safety-critical software in nuclear protection applications. Challenge also forms part of most development life cycles for safety-critical systems.

"Fault Injection for Software Certification," by Domenico Cotroneo and Roberto Natella, describes the use of software fault injection as a tool to facilitate certification of software embedded in safety-critical systems. Deliberately inserting software faults to assess the robustness of error-handling systems has been a widespread practice for critical applications.

The authors discuss their tool, SAFE, and how it's used to insert a realistic, representative set of faults. They illustrate the tool's application and methodology on the evaluation and improvement of a real-time operating system adopted for avionics applications. Through their fault injection experiments, they show how the routine action of an exception-handling routine actually results in termination of the real-time kernel's timer functions rather than providing for a graceful degradation and continuation of critical applications. The authors then suggest a more graceful recovery procedure that attempts to restart a critical kernel process rather than kill it. There will always be residual software bugs. The way we treat errors that manifest during operation has a great impact on system safety.

This special issue touches on just a few of the many aspects of safety-critical systems' design and certification—the continuing need to learn from experience, the need to fix mundane as well as sophisticated problems, the importance of rigorous analysis of appropriate engineering models, the need for independent challenges, and so on.

Traditionally, safety-critical systems' focus has been on mitigating the adverse effects of accidental faults and software errors. Most safety-critical systems have been stand-alone, with their electronics embedded into isolated subsystems. However, with the many benefits of networking systems together, even embedded computers are now accessible from outside the systems they serve. As such, new threat vectors have emerged in the form of malicious code, supply-chain attacks, and even radio frequency attacks on vital communications links. Some of the techniques and tools described in this issue can be adapted from accidental faults to deal with malicious ones. However, more needs to be done to address these emerging new threats to safety; to achieve security-informed safety, security and safety engineers need to appreciate their respective approaches. ■

Acknowledgments

We thank all those who submitted articles to this special issue and the external reviewers for their multiple rounds of thorough reviews, comments, and suggestions.

References

1. "Evidence from Mr. S. McGurk," *Cybersecurity: Assessing the Immediate Threat to the United States Hearing before the Subcommittee on National Security, Homeland Defense and Foreign Operations of the Committee on Oversight and Government Reform, House of Representatives One Hundred Twelfth Congress First Session May 25, 2011*, US Government Printing Office, 2011, p. 56.
2. D. Vaughan, *The Challenger Launch Decision: Risky Technology, Culture and Deviance at NASA*, Univ. Chicago Press, 1997.
3. C. Haddon-Cave, *The Nimrod Review: An Independent Review into the Broader Issues Surrounding the Loss of the RAF Nimrod MR2 Aircraft XV230 in Afghanistan in 2006*, Stationery Office, 28 Oct. 2009.

Robin Bloomfield is a professor at City University London and a founder of Adelard. Contact him at reb@csr.city.ac.uk.

Jay Lala is a Principal Engineering Fellow at Raytheon Company. Contact him at Jay_Lala@raytheon.com.