Online Ethics Center
FOR ENGINEERING AND SCIENCE

Software Engineer Challenges Authenticity of "Killer Robot" Software Tests

Author(s)

Richard G. Epstein

Description

An article about the authenticity of the Robot's software tests and the ethical issues raised from the falsification of those tests.

Body

Mabel Muckraker

Special to the *Silicon Valley Sentinel-Observer*

Silicon Valley, USA

The "killer robot" case took a significant turn yesterday when a Silicon Valley University professor issued a report questioning the authenticity of software tests that were reportedly performed on the "killer robot" software by Silicon Techtronics. Dr. Wesley Silber, professor of software engineering, told a packed news conference held at the university that the test results reflected in Silicon Techtronics internal documents were not consistent with test results obtained when he and his associates tested the actual robot software.

Silicon Valley is still reacting to Professor Silber's announcement, which could play an important role in the trial of Randy Samuels, the Silicon Techtronics programmer who has been charged with manslaughter in the now infamous killer robot incident. Pressed for her reaction to Professor Silber's report, Prosecuting Attorney Jane McMurdock reiterated her confidence that a jury will find Randy Samuels guilty. McMurdock shocked reporters, however, when she added, "But this does raise the possibility of new indictments."

Ruth Witherspoon, spokesperson for the Justice for Randy Samuels Committee, was almost exultant when she spoke to this reporter. She said, "McMurdock cannot have it both ways. Either the programmer is responsible for this tragedy, or management must be held responsible. We believe that the Silber report exonerates our friend and colleague, Randy Samuels."

Silicon Techtronics CEO Michael Waterson issued a terse statement concerning the Silber report: "Soon after the indictment of Randy Samuels was announced, I personally asked the esteemed software engineer Dr. Wesley Silber to conduct an impartial inquiry into quality assurance procedures at Silicon Techtronics. As the chief executive of this corporation, I have always insisted on quality first, despite what you might have read in the press. I promised Professor Silber that he would have access to all information relevant to this unfortunate situation. I told him in a face-to-face meeting in my office that he should pursue his investigation wherever it might lead, regardless of the implications. It never occurred to me, based upon the information that I was getting from my managers, that there might be a problem in which software quality assurance procedures were either lax or deliberately circumvented. I want the public to be reassured that the person or persons who were responsible for the failure of software quality assurance within the robotics division of Silicon Techtronics will be asked to find employment elsewhere."

Roberta Matthews, widow of Bart Matthews, the robot operator who was killed in the incident, spoke to the Sentinel-Observer by telephone from her home. She said, "I still want to see Mr. Samuels punished for what he did to my husband. I don't understand what all the commotion is about. The man who murdered my husband should have tested his own software!"

The Sentinel-Observer interviewed Professor Silber in his office shortly after his news conference. On his office wall were numerous awards he has received because of his work in the field of software engineering and software quality assurance. We began

the interview by asking Professor Silber to explain why it is that software is sometimes unreliable. He answered our question by citing the enormous complexity of software.

"Large computer programs are arguably the most complex artifacts ever fashioned by the human mind," Professor Silber explained, seated in front of a large computer monitor. "At any point in time, a computer program is in one of an extremely large number of possible states, and it is a practical impossibility to assure that the program will behave properly in each of those states. We do not have enough time to do that kind of exhaustive testing. Thus, we use testing strategies or heuristics that are very likely to find bugs, if they exist."

Professor Silber has published numerous papers on software engineering. He made headlines last year when he published his list of "Airlines to Avoid As If Your Life Depended Upon It." That list named domestic airlines he deemed irresponsible because of their purchase of airplanes that are almost completely controlled by computer software.

Professor Silber told the Sentinel-Observer about his work at Silicon Techtronics: "Mike [Waterson] told me to go in there [into the company] and conduct an impartial review of his software testing procedures and to make my findings public. Mike seemed confident, perhaps because of what his managers had told him, that I would find nothing wrong with quality assurance at Silicon Techtronics."

Professor Silber explained that "quality assurance" refers to those methods a software developer uses to assure that the software is reliable. These methods are applied throughout the development life-cycle of the software product. For example, when a programmer writes code, one quality assurance measure is to test the code by actually running it against test data. Another would be to run special programs, called static analyzers, against the new code. A static analyzer is a program that looks for suspicious patterns in programs, which might indicate an error or bug. These two forms of quality assurance are called dynamic testing and static testing, respectively. Software consists of discrete components or units that are eventually combined to create larger systems. The units themselves must be tested, and this process of testing individual units is called unit testing. When the units are combined, the integrated subsystems must be tested and this process of testing the integrated subsystems is called integration testing.

Soon after arriving at Silicon Techtronics, Professor Silber focused his attention on procedures for dynamically testing software at the high tech company. Assisted by a cadre of graduate students, Professor Silber discovered a discrepancy between the actual behavior of the section of program code (written by Randy Samuels) that caused the Robbie CX30 robot to kill its operator and the behavior of that code as recorded in test documentation at Silicon Techtronics. This discovery was actually made by Sandra Henderson, a graduate student in software engineering who is completing her doctorate under Professor Silber.

We interviewed Henderson in one of the graduate computer laboratories at Silicon Valley University. "We found a problem with the unit testing," Henderson explained. "Here are the test results, given to us by Mr. Waterson at Silicon Techtronics, which are purported to be for the C [programming language] code which Randy Samuels wrote and which caused the killer robot incident. As you can see, everything is clearly documented and organized. There are two test suites: one based upon white box testing and another based upon black box testing. Based upon our own standards for testing software, these test suites are well designed, complete, and rigorous."

She explained that black box testing involves viewing the software unit (or component) as a black box that has expected input and output behaviors. Test suites are designed to cover all "interesting" behaviors that the unit might exhibit but without any knowledge of the structure or nature of the actual code. If the component demonstrates the expected behaviors for all inputs in the test suite, then it passes the test. On the other hand, white box testing involves covering all possible paths through the unit. Thus, white box testing is done with thorough knowledge of the unit's structure. In white box testing, the test suite must cause each program statement to execute at least once so that no program statement escapes execution.

Henderson went on to explain the significance of software testing: "Neither black box nor white box testing proves that a program is correct. However, software testers, such as those employed at Silicon Techtronics, can become quite skillful at designing test cases so as to discover new bugs in the software. The proper attitude is that a test succeeds when a bug is found. Basically, the tester is given a set of specifications and does his or her best to show that the code being tested does not satisfy its specifications."

Henderson then showed this reporter the test results that she actually obtained when she ran the critical "killer robot" code using the company's test suites for white box and black box testing. In many cases, the outputs recorded in the company's test documents were not the same as those generated by the actual killer robot code that Henderson tested.

During his interview with the Sentinel-Observer yesterday, Professor Silber discussed the discrepancy: "You see, the software that was actually delivered with the Robbie CX30 robot was not the same as the software that was supposedly tested -- at least according to these documents! We have been able to determine that the actual killer code, as we call it, was written after the software tests were supposedly conducted. This suggests several possibilities: First, the software testing process, at least for this critical part of the software, was deliberately faked. We all know that there was enormous pressure to get this robot out the door by a date certain. Another possibility is that there was some kind of version management difficulty at Silicon Techtronics, so that correct code was written, successfully tested, but the wrong code was inserted into the delivered product."

We asked Professor Silber to explain what he meant by "version management." He said, "In a given project, a given software component might have several versions: version 1, version 2, and so forth. These reflect the evolution of that component as the project progresses. Some kind of mechanism needs to be in place to keep track of versions of software components in a project as complex as this one. Perhaps the software testers tested a correct version of the robot dynamics code, but an incorrect version was actually delivered. However, this raises the question as to what happened to the correct code."

Professor Silber sat back in his chair and sighed. "This really is a great tragedy. If the killer code had gone through the testing process in an honest manner, the robot would never have killed Bart Matthews. So, the question becomes, what was going on at Silicon Techtronics that prevented the honest testing of the critical code?"

The Sentinel-Observer asked Professor Silber whether he agreed with the notion that the user interface was the ultimate culprit in this case. He responded, "I don't buy the argument, being put forth by my colleague Professor Gritty, that all of the culpability in this case belongs to the user interface designer or designers. I agree with some of what he says, but not all of it. I have to ask myself whether Silicon Techtronics was placing too much emphasis on the user interface as a last line of

defense against disaster. That is, perhaps they knew there was a problem, but they felt that the user interface could allow the operator to handle that problem."

The Sentinel-Observer then asked Professor Silber about the charge made against him that he should never have accepted Waterson's appointment to conduct an impartial investigation into the accident. Critics have pointed out that Silicon Valley University, and Professor Silber in particular, has many business ties with Silicon Techtronics, and thus he could not be counted on to conduct an impartial investigation.

"I think my report speaks for itself," Professor Silber replied, visibly annoyed by our question. "I have told you reporters over and over again that this was not a government investigation but a corporate investigation, so I believe that Silicon Techtronics had the right to choose whomever they desired. I believe I was known to them as a person of integrity."

Late yesterday, Robbie CX30 project manager Sam Reynolds hired attorney Valerie Thomas, and Thomas issued this statement on his behalf: "My client is shocked that someone at Silicon Techtronics has misled Professor Silber concerning the software tests for the Robbie CX30 robot. Mr. Reynolds asserts that the software was tested and that he and others were well aware of the fact that there was something wrong with the robot dynamics software. However, Mr. Ray Johnson, my client's immediate superior at Silicon Techtronics, decided that the robot could be delivered to Cybernetics Inc., based upon Mr. Johnson's Ivory Snow Theory. According to that theory, the software was nearly bug free and thus could be released. According to Mr. Johnson, the risk of failure was very small and the cost of further delaying delivery of the robot was very great. According to my client, Mr. Johnson felt that the environmental conditions that could trigger erratic and violent robot behavior were extremely unlikely to occur. Furthermore, Mr. Johnson felt that the robot operator would not be in danger because the user interface was designed so as to permit the operator to stop the robot dead in its tracks in the case of any life-threatening robot motion."

Johnson, robotics division chief at Silicon Techtronics, could not be reached for comment.

Randy Samuels will be placed on trial next month at the Silicon Valley Court House. When contacted by phone, Samuels referred all questions to his attorney, Alex

Allendale.

Allendale had this to say concerning Professor Silber's findings: "My client submitted the software in question in the usual way and with the usual documentation and with the usual expectation that his code would be thoroughly tested. He was not aware until Professor Silber's report came out that the code involved in this terrible tragedy had not been tested properly or that the test results might have been faked.

"Mr. Samuels wants to again express the great sorrow he feels about this accident. He, more than anyone else, wants to see justice done in this case. Mr. Samuels once again extends his heartfelt condolences to Mrs. Matthews and her children."

Continue to Part 9: Silicon Techtronics Employee Admits Faking Software Tests

Rights

Use of Materials on the OEC

Resource Type

Case Study / Scenario

Topics

Product Liability
Lab and Workplace Safety
Workplace Ethics
Intellectual Property and Patents
Risk
Safety
Collaboration

Discipline(s)

Computer Sciences
Computer, Math, and Physical Sciences
Engineering
Mathematics