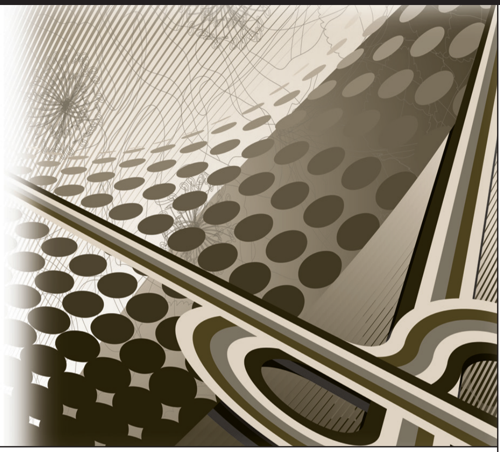
Sabotage!



David Alan Grier, *George Washington University*

The idea that an organized team of computer scientists might have created a major worm comes at an uneasy time for engineers.

At least Sam wasn't a conspiracy theorist. Few experiences are more tedious then that overwhelming monologue by an individual who has stumbled across an evil truth that is invisible to all save him. No, Sam wasn't a conspiracy theorist. He believed that there was an open conflict in the technical community between the moneyed classes and the proletariat of working engineers. According to him, financiers, boards of directors, bankers, insurance executives, and accountants had no interest in the production of good technology or any benefits that technology might bring to ordinary citizens. They were concerned only with the value of their investments.

Sabotage was Sam's favorite word to describe the state of the technology industry. He used the word in the reverse of its common meaning. When he talked about sabotage, Sam wasn't referring to the acts of common workers undermining the operation of a production process. Instead, he was referring to the efforts of financial leaders to maximize their profits at the expense of optimal production, new development, or progressive innovation. The keepers of the vested interests were increasing the price of goods, clinging to inefficient technologies, and thwarting ideas that were good for the general public, such as open source. This last subject was often the topic that would get Sam talking about industrial sabotage.

When faced with speeches such as Sam's diatribes on sabotage, listeners have three strategies: they can become quiet and ride out the storm; they can argue in an effort to change the speaker's opinions; or they can attempt to break contact and find a new collaborator.

In dealing with Sam, the first strategy quickly revealed itself to be a failure. Without any resisting force, he could talk about the sabotage of open source software as long as there were hours in the day.

The second strategy was equally ineffectual in my hands. At one point, I noted that Sam's ideas about sabotage were similar to those of the economist Oswald Veblen, who wrote about engineers and business in the years that followed World War I.

For a moment, Sam seemed interested in this connection. "What did he recommend?" he asked.

"Well," I said, "he argued that engineers should form an organization to take the control of production away from financial interests, an organization that he called "The Soviet of Engineers."

I should have known better than to use the word "soviet." I should have thought more carefully before I raised a historical example of a notoriously liberal economist, even though that economist was the uncle of a mathematician who greatly aided the development of the electronic computer.

My words were not a proper rebuttal. They were an effort to end an argument by employing the appearance of scholarship, and those words were repaid in kind. Sam showed himself distrustful of the word "soviet" and equally uncertain about the idea of engineers banding together. For the rest of the day, he railed about the ineffectiveness of all professional computer organizations including IFIPS, the ACM, CRA, and, of course, the IEEE Computer Society.

As have so many others in similar situations, I concluded that I could be more productive doing other work and moved to new tasks and new partners. However, I was in a position where I could do that easily. I had no boss, no assigned projects, no established goals. I didn't have to worry about the divided loyalties of the engineering profession, as so many engineers must. I didn't need

Published by the IEEE Computer Society

0018-9162/10/$26.00 © 2010 IEEE

to choose from among my own ideas, the ideas of my profession, and the ideas of my employer.

SOFTWARE AS SABOTAGE

In spite of his railing at my ill-conceived remarks, Sam probably did believe in a unity of interests and goals among engineers and technical personnel. He was a strong advocate for ethics education and often claimed that the scientific method should lead engineers to common social ideas and common goals. At the same time, I think he knew that such idealism was challenged by the waves of malware that have infected our systems and remind us that nothing is good or bad but thinking makes it so, that no system of thought lies beyond the motives of its originators.

In recent months, the malware that has most challenged our understanding of the role of engineers has been the Stuxnet virus. Stuxnet is a program that infects a supervisory control and data acquisition (SCADA) system from the Siemens Corporation that is used to control industrial processes such as chemical plants, oil refineries, and nuclear power plants. While much about Stuxnet is unknown as of this writing, it appears to be a program that is targeted at Iran's Bushehr nuclear reactor, a project that has been the concern of countries in the Middle East and Europe, as well as the United States.

News reports have focused on Stuxnet's origins. They speculate that it was created by some secret service that wanted to impede the progress of the Iranian atomic program. Some suggest that it was a military effort launched by Iran's enemies. Others have concluded that a group of criminal hackers developed the malware. There "were probably a number of participants in the Stuxnet development project who may have very different backgrounds," explained a well-circulated report. Some "of the code looks as if it originated with a 'regular' software developer with extensive knowledge of SCADA systems and/or Siemens control systems, rather than with the criminal gangs responsible for most malcode," the report noted.

Analysts have noted that some elements of Stuxnet appear to have been stolen while others might have been borrowed from other malware programs. Even if this is the case, it seems likely that the code was developed by a team of engineers who followed standard software engineering procedures. They developed malware specifications, designed the system, coded the malware, and debugged the system on a realistic testbed. If they were as organized as they appear to be, members of the team were collecting malware assessment data from the news reports.

If Stuxnet proves to be the agent of an established military, it will fit neatly into the past 60 years of warfare.

It is "feasible that what we're seeing here is the work of a more formally-constituted, multi-disciplinary 'tiger team'," the Stuxnet report adds. "Such official but unpublicized collaborations," the authors conclude, "might be more common than we are actually aware."

Even if such tiger teams aren't that common or even if one wasn't used for Stuxnet, the idea of such teams forces us to confront the ultimate goals of engineering. Malware, like any technology that is used as a weapon, carries no absolute engineering values beyond the claims that it has been designed well and built according to specifications. What is a virus to one community is a savior to its neighbor. Nothing is good or bad, says the poet, but thinking makes it so.

Of course, engineers have had to contend with the military aspects of their creation as long as engineers have existed. After all, the term "civil engineer," which is now used to describe the builders of roads, bridges, and other civic projects, was originally coined in the 18th century to distinguish such professionals from military engineers, which was implied by the unadorned term "engineer."

We have even become used to the idea that weapons are created by large coordinated technical staffs. Such staffs appeared in the first world war and matured in the second. New weapons must be created "by teams of men with different skills and angles of approach," explained an historian of World War II engineering.

The Stuxnet code carries several elements that suggest to some that it is indeed a weapon from some country's military. Of course, we have enough experience with malware to know that misleading clues can easily be inserted in any code. If Stuxnet proves to be the agent of an established military, it will fit neatly into the past 60 years of warfare. Since the end of World War II, countries have regularly used technology to disrupt the productive processes of their enemies without creating a public act of war. It has been a way for one government to sabotage another government.

At the same time, we might ultimately discover that Stuxnet isn't the product of a government but the organized creation of a group with minimal resources and no territory to call its own. It might have come from a gang of organized criminals, a rival corporation, or a terrorist group. Over the past year, a pair of social scientists, Diego Gambetta and Stefen Hertog, have been arguing that engineering education, far from instilling a common sense of social value, actually encourages individuals to join terrorist organizations. The "number of militant engineers relative to the total population of engineers is miniscule," Gambetta and Hertog acknowledge, "yet engineers, relative to other graduates, are overrepresented among violent Islamic radicals by three to four times the size we would expect."

Such ideas have produced the obvious responses from the engineering community, including a rather defensive statement from a former president of the National Academy of Engineering. We don't want to think ill about the education that has given us a career and an identity. Nor do we like to believe that the education that we received leads some people to do a bad thing. Yet we have lost some of the framework that allows us to see clearly what universal good might be.

Regularly, engineers are described as if they are actors working independently for the good of some universal society. While it might have been possible to make such a claim a century ago, today we find that most technical people are employed by institutions that are in competition. They are competing for market share, investment, the right to control a certain piece of territory, or the ability to govern a certain group of people. At that point, the good of society too quickly becomes the good of my employer. While nothing is inherently wrong with that state of affairs, we are doing little to erase the illusion of independent action and universal good and doing even less to help our students make wise judgments about questions that aren't easy to answer. "Is our organization doing good?" "Are our leaders making good decisions?" "Is our work being sabotaged?"

DIFFICULT JUDGMENT

Judgment, of course, comes not from education but from difficult experience. During that brief period of my life when I was employed by a start-up firm, I found myself struggling to keep communication flowing across that barrier that separates technical personnel from business leaders. Somehow, I believed, a good choice of words and a clever illustration would allow both sides to better understand what the company was doing and what needed to be accomplished for the greater good.

At one point, after I addressed a group of venture and angel investors, one of the venture partners approached me and said, "You seem to have your head screwed on. Let me tell you something. There isn't a business here. In two years, you'll be wiped off the map with the next technology. Sell the business now. You've got enough assets to get $1 million or so. Don't worry about the technical staff. They aren't keeping up with new developments and they'll be happy wherever they go."

The decision to sell wasn't mine to make. If it were, I'm not sure I would have done so. I had spent much of the prior year working with the chief engineer and helping him explain to the company president the demands that he was facing. I liked the people and felt that I had helped each side trust the other. The engineers realized that they had a short time to get the service to market. The president recognized that the technical problems demanded more resources to complete the work. To recommend a sale of the company at that moment would have seemed like betrayal.

Yet the decision to do nothing seemed to sabotage the company. As time marched forward, the tensions increased. The chief engineer started to feel that the president was undermining the technical staff by establishing unrealistic deadlines. The president concluded that the chief engineer didn't understand the pressures on the company, especially after he found the engineer shoveling the company parking lot after a light snow.

"If we don't do it, we'll get a ticket from the city," the engineer claimed as the president ranted about credit lines and closing market windows.

The end came a week or two later. The bank called and terminated a loan. A short scramble suggested that nothing could be done. The few assets that had value were sold on the open market. Employees left without pay. Accusations circulated about blame and responsibility.

The chief engineer left the company convinced that this work had been sabotaged, certain that the president had been more interested in preserving his investment than in producing a good product. As is true in so many situations, I again had three choices: I could be quiet; I could argue another point of view; or I could leave. As I had little stake in the venture, only a debenture for a few worthless shares of founders stock, I made only a cursory effort to engage the chief engineer before departing.

The recent reports about Stuxnet include claims from the Iranian government that it has arrested a number of people in connection with the case. We haven't received details about the arrests, at least none that can be verified by an independent agency. We can imagine that government may have arrested a foreign agent who planted the malware, or some loyal system programmers who thought that they had secured the site, or even a few from a local opposition party who may have had nothing to do with the reactor. In all cases, they were people who thought that they were doing their assignments properly and are now wondering how their work was sabotaged. C

David Alan Grier *is an associate professor of International Science and Technology Policy at the George Washington University. You can read more of his columns at www.computer.org/theknownworld. Contact him at grier@gwu.edu.*