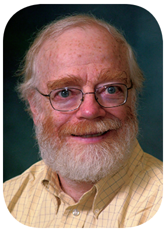
Perceptions and Reality



Steven M. Bellovin
Columbia University

Recently, a variety of news websites were all agog about a 2008 plane crash in Spain. Was a computer virus involved? Reading the headlines— “Malware Implicated in Fatal Spanair Plane Crash,” “Did Malware Cause Spanair’s McDonnell Douglas DC-9-82 (MD-82) EC-HFP to Crash?,” “Jetliner Crash Shows Dangers of Using Tainted USB Sticks,” and most ominously, “Murder by Malware: Can a Computer Virus Kill?” —would lead you to believe so, but the reality was rather more prosaic. That said, there are lessons we can learn from the incident and its aftermath. (While the conclusions in this essay are entirely mine, I thank *IEEE Security & Privacy*’s editorial board members; they supplied many useful links and helped shape my thinking about this.)

I read the interim report and the progress report issued by the official Spanish investigating agency (www.fomento.es/mfom/lang_en/direcciones_generales/organos_colegiados/ciaiac/investigacion/2008/spanair/interino.htm has both Spanish and English versions). One reporter—and as best I can tell, only one—read them, too, and provided the links I used (and came to the same conclusions I did; see www.zdnet.com/blog/bott/fact-check-malware-did-not-bring-down-a-passenger-jet/2354.) Together, the reports lay out a moderately complex sequence of events that apparently led to the crash. I say “apparently” because these are interim documents that present the facts rather than draw conclusions, but the broad outlines seem clear enough. A heater on the plane operated when the plane was on the ground (it should only operate in the air). The pilots noticed this while taxiing, and returned to the gate. A mechanic couldn’t reproduce the failure; however, since the heater in question was only needed in certain weather conditions and those weren’t in the forecast, he disabled the heater and the pilots decided to take off. Perhaps because they’d just been through the checklist, and perhaps because of the distractions of the interrupted take-off, the delay in the schedule, and a third person in the cockpit, the pilots neglected to lower the flaps and slats, which in turn was the proximate cause of the crash. A take-off warning system should have alerted them—but it was powered by a relay whose failure is the most likely cause of the original heater malfunction.

And the malware? A headquarters computer that was used to log mechanical failures was infected; allegedly, this prevented entering two previous failures. Possibly, the aircraft would have been grounded had the earlier failures been noted, but that would depend on airline policy and applicable regulations. It seems quite plausible, though, that nothing would have been done; as noted, under the weather conditions at the time there was no risk in disabling the heater. If someone had realized that the heater failure was most likely caused by the failure of a crucial relay, I’m sure action would have been taken, but the reports don’t indicate that this was understood except in retrospect. Besides, the mechanics didn’t even try entering the earlier reports until 24 hours after the failures—which was after the plane had crashed.

Why, then, is this a security story? The most important lesson here is that root-cause analysis is *hard*. At a time of increasing concern over cyberwarfare, it’s important to approach analysis of an incident with a great deal of humility. Acting in haste can be disastrous. A National Academies study made this point, quoting a senior official as saying “I have seen too many situations where government officials claimed a high degree of confidence as to the source, intent, and scope of a [cyber]attack, and it turned out they were wrong on every aspect of it. That is, they were often wrong, but never in doubt.”[1]

A second lesson is that complex systems fail in complex ways. In this incident, it took a (probable) relay failure, a mysterious symptom, apparent human errors, a hidden single point of failure that

continued on p. 87

continued from p. 88

both caused the heater misbehavior and disabled a crucial warning system, and—possibly but not in my judgment likely—malware infecting a support system that, if functional, might have caused the plane to be grounded. Of course, our security systems have similarly complex interconnections—think of everything that goes into accepting code because it's digitally signed, ranging from the honesty and competence of those running the top-level certificate authority to the correctness of the code that parses the signed object to the availability of the network connections used to check the revocation status. Are there hidden failure modes? Can an enemy trigger them? Can you tell an attack from a mysterious failure? (It's also worth noting the scope and depth of airplane crash investigations, compared to what happens after a typical security incident. Of course, airplane crashes are much less common.)

The final lesson concerns how we deal with the public and the press. Most reporters aren't technical experts; instead, they rely on us. We must be careful about the message we send. We not only have to explain the facts, we have to explain what they mean, and we have to do our best to make sure that reporters *understand* what we're saying. Often, they're focused on the hot topic *du jour*; what's sexy, sells. They won't always get it right (if for no other reason than that we won't always get it right), but the effort is worth making. And when there's a failure—when the press misreports something—we need to pick up our virtual quills and write something ourselves. □

Reference

1. W. Owens, K. Dam, and H. Lin, Technology, Policy, Law, and Ethics Regarding U.S. Acquisition and Use of Cyberattack Capabilities, National Academies Press, 2009, p. 142.

Steven M. Bellovin *is a professor of computer science at Columbia University. Contact him via www.cs.columbia.edu/~smb.*