The New York TimesThe New York Times | http://nyti.ms/1L5wnHN

The Opinion Pages | CONTRIBUTING OP-ED WRITER

Volkswagen and the Era of Cheating Software

SEPT. 23, 2015

Zeynep Tufekci

FOR the past six years, Volkswagen has been advertising a lie: “top-notch clean diesel” cars — fuel efficient, powerful and compliant with emissions standards for pollutants. It turns out the cars weren’t so clean. They were cheating.

The vehicles used software that cleverly put a lid on emissions during testing, but only then. The rest of the time, the cars spewed up to 40 times the legal limit of nitrogen oxide emissions. The federal government even paid up to $51 million in tax subsidies to some car owners on the false assumption of environmental friendliness.

In a world where more and more objects are run by software, we need to have better ways to catch such cheaters. As the Volkswagen case demonstrates, a smart object can lie and cheat. It can tell when it’s being tested, and it can beat the test.

The good news is that there are well-understood methods to safeguard the integrity of software systems. The bad news is that there is as yet little funding for creating the appropriate regulatory framework for smart objects, or even an understanding of the urgent need for it. We are rightly incensed with Volkswagen, but we should also consider how we have ceded a lot of power to software that runs

everything from our devices to our cars, and have not persisted in keeping tabs on it. We correctly worry about hackers and data leaks, but we are largely ignoring the ramifications of introducing software, a form of intelligence, to so many realms — sometimes called the Internet of Things.

Corporate cheating is not novel: that's why we have regulations to oversee the quality of many objects, ranging from lead in paint to pesticide residue in food. If similar precautions are not extended to the emergent realm of computer-enhanced objects, especially when the software is proprietary and thus completely controlled by the corporation that has huge incentives to exaggerate performance or hide faults during tests for regulatory benchmarks, Volkswagen will be neither the first nor the last scandal of the Internet of Cheating Things.

And cheating on crucial standards is more than slight misconduct. In 1999, in the aftermath of a major earthquake in Turkey, I walked on mangled streets lined by a zigzagged skyline: Some buildings had collapsed into twisted heaps while others next to them stood tall. A seasoned earthquake rescuer explained to me how survival could be so random. Some of the builders cheated on the codes for concrete — too much sand, no interconnecting metal rods to keep the columns in place. Just this month, a powerful earthquake in Chile — where strict building regulations are properly enforced — killed about 20 people, while 17,000 perished in Turkey's 1999 earthquake.

Cheating software does not generate a trail of dust the way cheating concrete does. Volkswagen's duplicity had been going on for at least six years. Last year, confronted by higher-than-allowed test results obtained by researchers at a West Virginia University lab, Volkswagen managers claimed that the differences were the result of flaws in the testing and the way the vehicles were being driven, and kept up the apparent deceit for another year. Had it not been for the diligence of researchers in two small labs, one in Germany and one in the United States, they might have gone on cheating without notice.

This isn't the first instance of a car company caught cheating by using a "defeat device" on emissions tests. In 1998, Ford was fined $7.8 million for using defeat devices that allowed its Econoline vans to reduce emissions to pass testing, and then to exceed pollution limits when driving at highway speeds. The same year,

Honda paid $17.1 million in fines for deliberately disabling a “misfire” device that warned about excess emissions. In 1995, General Motors paid $11 million in fines for the “defeat devices” on some of its Cadillac cars, which secretly overrode the emissions control system at times. The largest penalty for defeat devices to date was an $83.4 million fine in 1998 on Caterpillar, Volvo, Renault and other manufacturers.

Computational devices that are vulnerable to cheating are not limited to cars. Consider, for example, voting machines. Just a few months ago, the Virginia State Board of Elections finally decertified the use of a touch-screen voting machine called “AVS WinVote.” It turned out that the password was hard-wired to “admin” — a default password so common that it would be among the first three terms any hacker would try. There were no controls on changes that could be made to the database tallying the votes. If the software fraudulently altered election results, there would be virtually no way of detecting the fraud since everything, including the evidence of the tampering, could be erased.

If software is so smart and its traces of tampering are possible to erase, does this mean that we have no hope of catching cheaters? Not at all. We simply need to adopt and apply well-known methods for testing computing devices.

First, smart objects must be tested “in the wild” and not just in the lab, under the conditions where they will actually be used and with methods that don’t alert the device that it’s being tested. For cars, that means putting the emissions detector in the tail pipe of a running vehicle out on the highway. For voting machines that do not have an auditable paper trail, that means “parallel testing” — randomly selecting some machines on Election Day, and voting on them under observation to check their tallies. It is otherwise too easy for the voting machine software to behave perfectly well on all days of the year except, say, Nov. 8, 2016.

Second, manufacturers must not be allowed to use copyright claims on their software to block research into their systems, as car companies and voting machine manufacturers have repeatedly tried to do. There are proprietary commercial interests at stake, but there are many ways to deal with this obstacle, including creating special commissions with full access to the code under regulatory supervision.

Third, we need to regulate what software is doing through its outputs. It's simply too easy to slip in a few lines of malicious code to a modern device. So the public can't always know if the device is working properly — but we can check its operation by creating auditable and hard-to-tamper-with logs of how the software is running that regulators can inspect.

None of this is impossible. There is one industry in particular that employs many of these safeguards in an admirable fashion: slot machines in casinos. These machines, which in some ways present the perfect cheating scenario, are run by software designed by the manufacturers without a centralized database of winnings and losses to check if frequencies of losses are excessive. Despite all these temptations, in many jurisdictions, these machines run some of the best regulated software in the country. The machines are legally allowed to win slightly more often than lose, of course, ensuring a tidy profit for the casinos (and tax revenues for the local governments) without cheating on the disclosed standards.

It's a pity that casinos have better scrutiny of their software than the code running our voting machines, cars and many other vital objects, including medical devices and even our infrastructure. As computation spreads in society, our regulatory systems need to be funded appropriately and updated in their methods so that keeping our air clean and our elections honest is not a worse gamble than a slot machine.

Zeynep Tufekci is an assistant professor at the School of Information and Library Science at the University of North Carolina and a contributing opinion writer.

A version of this op-ed appears in print on September 24, 2015, on page A35 of the New York edition with the headline: VW's Cheating Software.