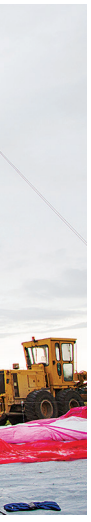
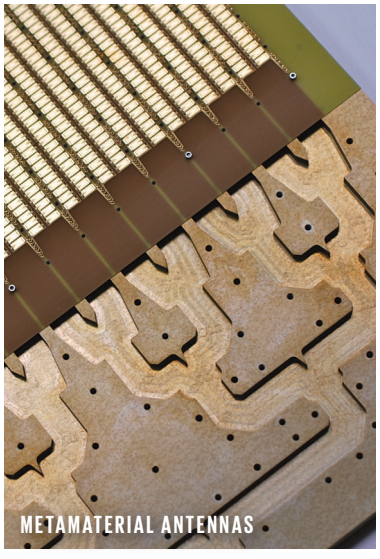
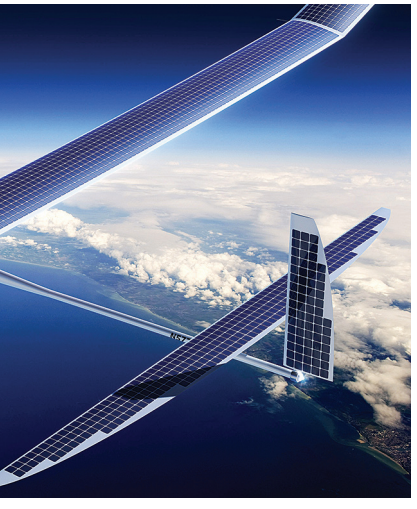






SECRET SOFTWARE

A new form of encryption could make practically unhackable code

Software can hold invaluable secrets within orderly lines of code. Algorithms now predict which Amazon.com products you're likely to buy next, whether an early movie script will be a box-office hit, and even whether or not a legal case will go to trial. Naturally, coders of such software don't want outsiders to be able to reverse engineer the programs and learn their secret formulas. Now, computer scientists from the University of California, Los Angeles, IBM Research, and the University of Texas at Austin have begun to pave the way toward eliminating that threat.

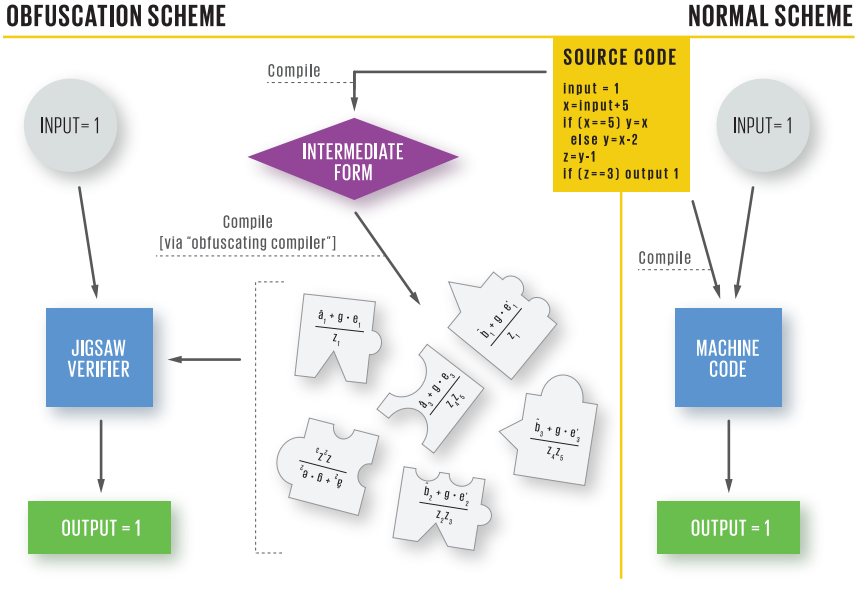
The researchers say they've developed a "mathematical obfuscation" scheme to encrypt such valuable software. They hope that one day this scheme will allow users to run programs normally while transforming the underlying code into math puzzles that would take hundreds of years to solve.

To see how software obfuscation works, you need to understand how programs ordinarily operate. People first write programs in languages that humans can understand, and then a compiler translates that script into machine code–the instructions the CPU can execute. The program can then receive inputs and produce the proper outputs.

In the researchers' obfuscation scheme, the compiler first translates code into an intermediate form; then an obfuscating compiler translates this form into pieces of what UCLA computer science professor Amit Sahai calls a "mathematical jigsaw puzzle." A special jigsaw verifier program, written in machine code, takes these puzzle pieces–essentially sequences of random numbers–and all the inputs and tries to assemble them. If the pieces "fit," then the completed "puzzle" will tell the CPU how to produce the correct output.

If the pieces don't fit–for instance, because a hacker has altered the code in an attempt to figure out how the software works–the resulting output is useless. "The modified software would not give

PUZZLE PROCESSING: Indistinguishability obfuscation hides the inner workings of software by turning it into a "mathematical jigsaw puzzle" that's nearly impossible for an attacker to solve.



you any insight into how the original software works," Sahai says.

In the scheme's current version, obfuscation requires too much computation to make it feasible, but its developers think a practical version is possible.

For decades, true software encryption eluded computer scientists. In 2001, seminal research by Boaz Barak at the Weizmann Institute in Rehovot, Israel, and his colleagues proved that there are some programs for which an ideal version of obfuscation is impossible. This notion of obfuscation, called virtual black box, demands that an encrypted program's inner workings be completely hidden. While disappointing, this research still held out the possibility that almost all programs could be encrypted using another type of scheme, which is called indistinguishability obfuscation, or IO. This type would also obscure the program; however, the source code could still, technically, be deciphered if you were willing to put in an impractical amount of time and resources.

"We didn't know if it was possible or not to achieve indistinguishability obfuscation" in 2001, says Barak, now a senior researcher at Microsoft Research New England.

What Sahai and his team—Sanjam Garg, Craig Gentry, Shai Halevi, and Mariana Raykova of IBM Research, and Brent Waters, an assistant professor of computer science at the University of Texas at Austin—say they've proved with their mathematical jigsaw puzzle trick is that indistinguishability obfuscation is possible. Their research will be presented this month at the 54th annual IEEE Symposium on Foundations of Computer Science, in Berkeley, Calif.

Such obfuscation would be particularly useful for software whose inner workings could reveal security flaws to hackers. When app makers issue security patches, hackers often grab them and quickly dig into the code to figure out a system's Achilles' heel, uncovering the vulnerability that the patch was designed to fix. "The crucial technical goal for us was to prevent this adversary from being able to do this," Sahai says.

According to Sahai, his team's research throws the doors open to new possibilities in the field of cryptography. In one possibility, called functional encryption, entire computable functions may be designed to run differently for individual users depending on their identity or purpose. For instance, a medical researcher might have a unique key that allows access to sensitive hospital data while keeping patients' personal information private.

Sahai's team says the next step in their research is to reduce the amount of computation needed to perform obfuscation. Based on previous trajectories in cryptographic research, the system could be practical in about a decade, Sahai says.

Microsoft Research's Barak, for one, extols the group's accomplishment. "This new work…gives a proof of concept that it could be achievable," he says. "Even if it's not yet practical…they showed how to use IO to achieve not all but many of the applications of full-fledged obfuscation," he says.

Shafi Goldwasser, a professor of electrical engineering and computer science at MIT, says that while the team used a looser definition of obfuscation than earlier research did, it's laudable that their system works on all programs under this new definition. "It's clear that this is a hammer that works on a lot of nails; it can be used to address many other open problems of cryptography," she says. "And from that point of view, it's a big deal." —DAVEY ALBA

Last year, the Hoover Dam hydroelectric plant installed the first of five wide-head turbines. These are designed to work efficiently even as the Colorado River shrinks under a record-long drought. The dry spell affecting the dam's power source has outlasted any other in the 77 years that the structure has generated electricity. By the time the fifth turbine is installed in 2016, Hoover Dam will likely need them all.

Lake Mead, which sits on the border between Nevada and Colorado behind Hoover Dam, is expected to drop 2.4 meters in 2014, as less and less water flows downstream from Lake Powell, which straddles Utah and Arizona. The sharp decline comes about because the U.S. Bureau of Reclamation needs to cut Lake Powell's water release by nearly 1 billion cubic meters to 9.2 billion m^3 for the 2014 water season, the smallest release since the lake was filled in the 1960s. The flow of water to Lake Powell from key tributaries has been decreasing for more than a decade, and the bureau's forecasters expect that the reservoir could hit an all-time low this season.

"This is the most extreme drought since measurements began in the early 1900s," says Jack Schmidt, a professor of watershed

PREVIOUS PAGE: CLOCKWISE FROM TOP LEFT: GOOGLE; UNIVERSITY OF STRATHCLYDE; TITAN AEROSPACE CORP.; KYMETA CORP.; O3B NETWORKS