CRYPTOGRAPHY

A Tricky Path to Quantum-Safe Encryption

In the drive to safeguard data from future quantum computers, cryptographers have stumbled upon a thin red line between security and efficiency.

Peter Diamond for Quanta Magazine

The most promising quantum-secure encryption schemes rely on the near impossibility of navigating a lattice with hundreds of spatial dimensions. Unless you know the secret route.

By: Natalie Wolchover

September 8, 2015

Comments (1)

PDF Print

On August 11, the National Security Agency updated an obscure page on its website with an announcement that it plans to shift the encryption of government and military data away from current cryptographic schemes to new ones, yet to be determined, that can resist an attack by quantum computers.

“It is now clear that current Internet security measures and the cryptography behind them will not withstand the new computational capabilities that quantum computers will bring,” NSA spokesperson Vanee’ Vines stated in an email, confirming the change. “NSA’s mission to protect critical national security systems requires the agency to anticipate such developments.”

Quantum computers, once seen as a remote theoretical possibility, are now widely expected to work within five to 30 years. By exploiting the probabilistic rules of quantum physics, the devices could decrypt most of the world's "secure" data, from NSA secrets to bank records to email passwords. Aware of this looming threat, cryptographers have been racing to develop "quantum-resistant" schemes efficient enough for widespread use.

The most promising schemes are believed to be those based on the mathematics of lattices — multidimensional, repeating grids of points. These schemes depend on how hard it is to find information that is hidden in a lattice with hundreds of spatial dimensions, unless you know the secret route.

But last October, cryptographers at the Government Communications Headquarters (GCHQ), Britain's electronic surveillance agency, posted an enigmatic paper online that called into question the security of some of the most efficient lattice-based schemes. The findings hinted that vulnerabilities had crept in during a decade-long push for ever-greater efficiency. As cryptographers simplified the underlying lattices on which their schemes were based, they rendered the schemes more susceptible to attack.

Building on the GCHQ claims, two teams of cryptanalysts have spent the past year determining which lattice-based schemes can be broken by quantum computers, and which are safe — for now.

"This is the modern incarnation of the classic cat-and-mouse game between the cryptographer and cryptanalyst," said Ronald Cramer of the National Research Institute for Mathematics and Computer Science (CWI) and Leiden University in the Netherlands. When cryptanalysts are quiet, cryptographers loosen the security foundations of the schemes to make them more efficient, he said. "But at some point a red line might be crossed. That's what happened here." Now, the cryptanalysts are speaking up.

Open Secrets

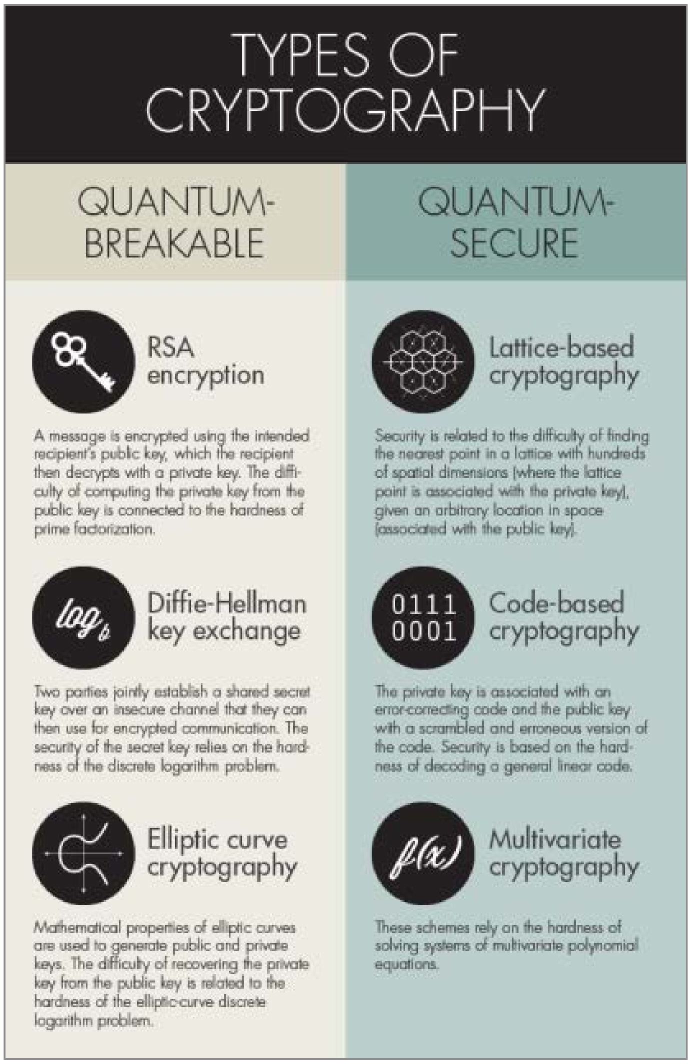
Every time you visit a website with a URL that begins "HTTPS," you send or receive encrypted data. Secure Internet transactions are made possible by public-key cryptography, a revolutionary invention of the 1970s. Up until then, cryptography had mostly been a game for governments and spies; two parties, such as a spy and a handler, had to agree in advance on a secret cipher or "key" in order to communicate in secret. (The simple "Caesar cipher," for example, shifts the letters of the alphabet by some agreed-upon number of positions.) Public-key cryptography makes it possible for anyone to send anyone else an encrypted message that only the recipient can decrypt, even if the parties involved never agreed on anything and no matter who is listening in.

"The reception from NSA was apoplectic," Martin Hellman, one of the three Stanford University researchers who invented public-key cryptography, recalled in 2004.

In public-key cryptography, data is secured by math problems that are easy to solve, but hard to reverse engineer. For example, while it is easy for a computer to multiply two prime numbers to produce a larger integer, as in the calculation 34,141 x 81,749 = 2,790,992,609, it is hard — that is, it takes an impractically long time on a computer — to factorize a large enough integer into its component primes. In a crypto scheme based on prime factorization, the primes serve as a person's "private key," which is not shared. The product of the primes serves as the "public key," which is distributed publicly. When someone else uses the public key to encrypt a message, only the person in possession of the private key can decrypt it.

Two efficient public-key encryption schemes that emerged in the late 1970s remain the most widely used today: RSA (invented by Ron Rivest, Adi Shamir and Leonard Adleman), based on the prime factoring problem, and the Diffie-Hellman key exchange (invented by Whit Diffie and Hellman), based on what's called the discrete logarithm problem. Although there was no actual proof that either prime factors or discrete logarithms were impossible to compute in a reasonable time frame, no one could find algorithms for efficiently computing them.

"Over time, people build up confidence in the hardness of some problem because so many people have tried to think about how to break it and cannot," said Jill Pipher, a mathematician and cryptographer at Brown University.


TYPES OF CRYPTOGRAPHY

QUANTUM-BREAKABLE

RSA encryption

A message is encrypted using the intended recipient's public key, which the recipient then decrypts with a private key. The difficulty of computing the private key from the public key is connected to the hardness of prime factorization.

Diffie-Hellman key exchange

Two parties jointly establish a shared secret key over an insecure channel that they can then use for encrypted communication. The security of the secret key relies on the hardness of the discrete logarithm problem.

Elliptic curve cryptography

Mathematical properties of elliptic curves are used to generate public and private keys. The difficulty of recovering the private key from the public key is related to the hardness of the elliptic-curve discrete logarithm problem.

QUANTUM-SECURE

Lattice-based cryptography

Security is related to the difficulty of finding the nearest point in a lattice with hundreds of spatial dimensions (where the lattice point is associated with the private key), given an arbitrary location in space (associated with the public key).

Code-based cryptography

The private key is associated with an error-correcting code and the public key with a scrambled and erroneous version of the code. Security is based on the hardness of decoding a general linear code.

Multivariate cryptography

These schemes rely on the hardness of solving systems of multivariate polynomial equations.

Olena Shmahalo/Quanta Magazine

All three of the most widely used cryptographic schemes can be broken by algorithms designed to run on future quantum computers (left column). Cryptographers have devised a variety of schemes, three of which appear on the right, that are thought to be quantum-secure.

With existing algorithms, it takes years to compute the prime factors associated with a public key of typical length. And so, RSA and the Diffie-Hellman key exchange became the armor of the Internet, and a sense of security reigned.

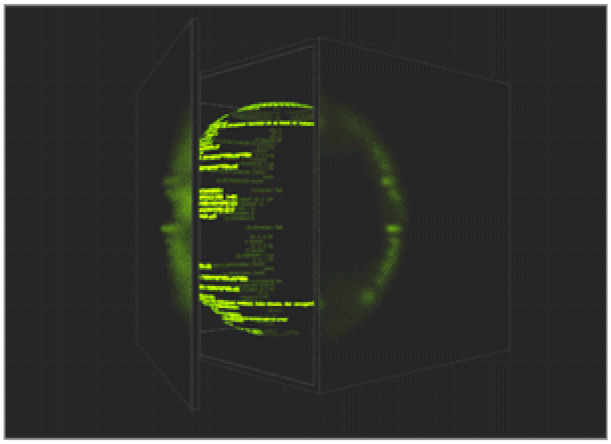
That security, it turns out, came with an expiration date.

Shor's Algorithm

Assumptions about which math problems are hard for computers to solve shattered in 1994, when an AT&T researcher named Peter Shor revealed the theoretical decrypting power of future quantum computers.

In an ordinary computer, information is stored in units called bits that can exist in either of two states, designated 0 or 1. The computer's computational capacity is proportional to the number of bits. In quantum computers, however, the information-storing units, called qubits, can exist in both the 0 and 1 states simultaneously. (Qubits might take the form of subatomic particles spinning both clockwise and counterclockwise at the same time, for example.) Because a system of many qubits can exist in all possible combinations of all their possible individual states, the computational capacity of a quantum computer would increase exponentially with the number of qubits.

Related Articles:



A New Design for Cryptography's Black Box

A two-year-old cryptographic breakthrough has proven difficult to put into practice. But new advances show how near-perfect computer security might be surprisingly close at hand.

The Proof in the Quantum Pudding

How do you know if a quantum computer is doing what it claims? A new protocol offers a possible solution and a boost to quantum cryptography.

This would seem to make quantum computers more powerful problem solvers than classical computers. However, actually tapping their potential requires finding an algorithm for juggling their simultaneous realities, so that in the end, the right one — that is, the state of the system corresponding to the correct answer — emerges. For more than a decade after quantum computing was conceived in the early 1980s, no promising algorithms emerged, and the field languished. "Frankly, nobody paid any attention," said Seth Lloyd, a quantum computing theorist at the Massachusetts Institute of Technology.

All that changed in 1994 when Shor, who is now at MIT, devised a quantum computer algorithm capable of efficiently computing both prime factors and discrete logarithms, and thus of breaking both RSA encryption and the Diffie-Hellman key exchange. “At that point there was a killer app for quantum computing — maybe you could call it a quapp — and the interest in quantum computing boomed,” Lloyd said.

With the superior computational capabilities of quantum computers revealed by Shor’s algorithm, researchers worldwide have been racing to build them ever since. In parallel, cryptographers have raced to come up with new schemes that quantum computers can’t crack. “We didn’t know where to look for a long time,” said Chris Peikert, a cryptographer at the Georgia Institute of Technology in Atlanta. “But lattices seem to be a very good foundation.”

Lost in Lattices

Just as the security of RSA encryption is based on the idea that it’s easy to multiply primes but hard to compute prime factors, the security of lattice-based crypto schemes rests on how easy it is to get lost in a 500-dimensional lattice: You simply start at a lattice point and jiggle the spatial coordinates, ending up at some location nearby. But it’s exceedingly hard to find the nearest lattice point, given an arbitrary location in 500-dimensional space. In lattice-based schemes, the private key is associated with the lattice point, and the public key is associated with the arbitrary location in space.

Despite its promise, lattice-based cryptography got off to a slow start. In the 1980s, public keys based on lattices were too long, requiring megabytes of data to transmit. Cryptographers were forced to simplify the underlying lattices for the sake of efficiency. In a generic lattice, lattice points are generated by taking all possible linear combinations of some set of vectors (arrows pointing in different directions). Assigning a pattern to these vectors makes the resulting lattice simpler, and the associated keys shorter. Invariably, however, simplifying the lattice also makes it easier to navigate, allowing private keys to be deduced from public keys, thereby breaking the scheme. “Lattices became synonymous with disaster — with failed attempts at crypto,” said Jeff Hoffstein, a mathematician at Brown.

While the rest of the world moved on, some cryptographers continued to tinker with lattices. In 1995, Hoffstein, with Pipher and another Brown colleague, Joe Silverman, devised a cryptographic scheme based on “cyclic” lattices, which are generated by vectors that can rotate in any direction and still land on another lattice point. NTRU, as they called the scheme, was extremely efficient — even more so than the RSA and Diffie-Hellman protocols. Although there was no proof that the cyclic lattices underlying NTRU were hard for computers to navigate, or that NTRU was secure, 20 years have passed and no one has found a way to break it, boosting confidence in its security.

The promise of lattices grew dramatically in 1997, when the IBM researchers Miklós Ajtai and Cynthia Dwork devised the first lattice-based crypto scheme that was provably as hard to break as the underlying lattice problem is hard to solve. Building on this work, Oded Regev, a theoretical computer scientist now at New York University’s Courant Institute of Mathematical Sciences, proved in 2005 that crypto schemes based on a problem called learning with errors (LWE) are secure against quantum computers, as long as the problem of finding the nearest point in a generic lattice is hard for quantum computers (as most researchers presume). LWE was inefficient, but Regev, Peikert and Vadim Lyubashevsky, who is now at

IBM Research in Switzerland, soon developed analogous schemes based on “ideal” lattices (which are closely related to cyclic lattices), and showed that these more efficient schemes, dubbed Ring-LWE, are secure as long as the underlying, ideal lattice problem is hard.

Learning With Errors

In 2005, Oded Regev devised a cryptographic scheme based on a problem called “learning with errors,” which he proved is as secure as the lattice problem is hard. LWE-based schemes work something like this:

Pick any odd number, and don’t tell anyone what it is. That’s your private key. Now multiply it by any other number, and add a small even number to it. (For example, if your original number was 121, you might multiply it by five and then add two, yielding 607. In practice, the numbers are much larger.) Do this many times, producing a list of enlarged, perturbed versions of your private key. This list of numbers is your public key. Tell the world.

Now, say someone wants to send you a message (e.g., 0 or 1, attack or retreat, yes or no). First this person randomly selects half of the numbers listed in your public key and adds them together. Then, to send the message “0,” your correspondent simply sends the sum back to you. To send the message “1,” the person adds one to the sum, then sends this back to you. Now, to decode the message, you simply divide the sum you’ve received by your private key. If the remainder is even, the message is “0.” If the remainder is odd, the message is “1.”

Once again, however, there seemed to be a tradeoff between security and efficiency, and an irksome impossibility of having both. Ring-LWE had better security assurances than NTRU and was far more versatile, but it was not as efficient. Some researchers believed they could do better. Since 2007, they have been considering cryptographic schemes based on “principal ideal lattices,” which are generated by a single vector, in much the same way that the set of integers { … , -6, -3, 0, 3, 6, 9, … } can be generated by multiples of the integer 3.

“They were greedy; they were not happy with existing efficiency,” Regev said.

Cat and Mouse

As academic cryptographers devised crypto schemes based on principal ideal lattices, so did people behind the scenes at GCHQ. Their secret scheme, called Soliloquy, employed techniques from number theory to reduce the public-key size from a matrix of large numbers down to a single prime number. In the underlying lattice problem, this is equivalent to generating a lattice with a single, very short vector. “Unfortunately, the constructions used to do this were its Achilles heel,” a GCHQ spokesperson said in an email.

In their paper published last October, titled “Soliloquy: A Cautionary Tale,” the GCHQ researchers revealed that they had invented Soliloquy and then abandoned work on it in 2013 upon discovering a quantum attack that could break it. The paper provided only a vague sketch of the attack, however, leaving open the question of how it worked and which other lattice-based schemes might be affected. It seemed that in the pursuit of efficiency, a red line had been crossed. But where was the line?

"There was this initial idea that this attack could possibly be broader, and perhaps implicated all of lattice-based cryptography," Pipher said. Others were skeptical that the attack worked at all.

Cryptographers have spent almost a year determining the scope of the Soliloquy attack. "People became obsessed," Hoffstein said. "There was a frenzy." It turned out that the GCHQ team had not worked out many details themselves, but merely had "sufficiently strong evidence that an attack could be developed and hence that Soliloquy could not be recommended for real-world use," as the spokesperson put it in an email. In a March paper, Regev, Peikert, Cramer and Léo Ducas of CWI worked out the part of the attack that required only an ordinary computer; last week, Jean-François Biasse and Fang Song of the University of Waterloo in Ontario laid out the quantum steps.

Besides Soliloquy, the findings indicated that other schemes based on principal ideal lattices generated by a single short vector are also broken, whereas schemes based on more generic ideal lattices, such as Ring-LWE and NTRU, are not affected. "There seem to be some initial technical obstacles in transferring these techniques to other important schemes," Cramer said, adding that it warrants further study.

On the security-efficiency continuum, cryptographers slid too far to the efficiency side. In their scramble to find the best quantum-resistant schemes for banks, governments and the rest of the secure Internet, the Soliloquy attack has forced them back, toward schemes that are somewhat less efficient, but more firmly based on hard lattice problems. That is, presumed hard lattice problems.

There's no proof that quantum computers cannot find their way around lattices. "It could be that all these problems are actually easy," Peikert said. "But it seems unlikely, given what we know."

As for why extreme efficiency and perfect security appear to be so diametrically opposed, Hoffstein said: "The universe is an irritating place, and this is just another example of it."

Correction: This article was revised on Sept. 8, 2015, to clarify that the schemes broken by the Soliloquy attack are those based on principal ideal lattices generated by a single, short vector, and not those based on principal ideal lattices generated by more than one short vector.

Share This Article

Add a Comment

View Comments (1)