Keeping it Safe and Quiet

Age 14 to 18
Article by Simon Singh
Published February 2012, December.

Most mathematical research is published in academic journals or discussed at conferences, but some of the mathematicians who work in the area of cryptography realise that the rules are different for them. If they discover a new code or crack an existing code, then national security concerns might mean that their work is kept secret for years or decades.

For example, the Bletchley Park codebreakers had to wait until the 1970s before their contribution to the war effort was declassified, by which time many of the leading figures had already died. Alan Turing, perhaps the most famous Bletchley codebreaker, tragically committed suicide in 1954, having received no public recognition for his contribution to breaking the German Enigma code. As 2012 is the centenary of his birth, many mathematicians and historians hope that this will be an opportunity to remember his genius and the extraordinary efforts of his Bletchley colleagues.

Although modern encryption is a more public affair, because of its relevance to the general public and businesses, there is still a large amount of clandestine cryptography, and there are cryptographers whose brilliance continues to be shrouded in government secrecy. Indeed, while writing 'The Code Book', I discovered that a vital cryptographic invention, something that many of us use every day, has its own secret history.

The invention is something called public key cryptography (PKC), which is vital to how we exchange data in the information age. Without PKC, everything from e-commerce to secure phone calls would not be possible. First, I will explain why PKC is necessary, then I will explain how it works, then who discovered it, and then who really discovered it.

Why is PKC vital in the information age?

Let's meet three characters who often crop up in discussions about cryptography, Alice, Bob and Eve. Typically, Alice wants to send a secret message to Bob, but Eve (the eavesdropper) is trying to intercept the message. Naturally Alice wants to protect the message, so she encrypts or scrambles it. In order for this to work, however, Bob has to be able to unscramble the message, which means that he needs to know the recipe that Alice used to scramble the message in the first place. Alice has somehow

to get the scrambling recipe, known as the key, to Bob without it falling into the hands of Eve.

Essentially, the only solution for Alice is to send the key to Bob via a trusted courier beforehand. This so-called key distribution has been at the heart of cryptography for millennia. The officials who ran the German Enigma network would distribute keys, and in the 1970s banks employed specially vetted dispatch riders, who would race across the country with padlocked briefcases, personally distributing keys to everyone that the bank would communicate with over the next week. Similarly, government security agencies would transport tons of keys around the world every day. When ships carrying key material came into dock, crypto custodians would march on board, collect stacks of cards, paper tapes, floppy disks, or whatever other medium the keys might be stored on, and then deliver them to the intended recipient.

Key distribution might seem like a mundane issue, but it was the weakest link in the chain of security, because there was always the risk of a courier selling keys to the enemy. Also, as communication networks grew in size, the problem also grew, and it became clear that key distribution was turning into a logistical nightmare. However, finding a solution seemed to be impossible. If Alice wants to share a secret message with Bob, then, surely she must first agree another secret with him, namely the key?

Common sense seems to suggest that key distribution is an annoyingly necessary part of secure communication, but PKC demonstrates that that there is a way around the problem.

How does PKC work?

Let's return to Alice, Bob and Eve and let's imagine encryption in terms of locking a message inside a box. Alice puts her message in a box, puts a padlock on the box, and then sends it to Bob. The good news is that the padlock stops Eve reading the message, as she does not have a key. Unfortunately, it also stops Bob accessing the message - unless he has a copy of Alice's key. In other words, we have run into the key-distribution problem again. Alice cannot securely send Bob a message unless she has already sent him the key.

However, there is a sneaky solution to this problem, which avoids distributing any keys. Let's turn the problem on its head, and let's allow the receiver, not the sender, to take more responsibility for encryption. Imagine that Bob designs a padlock and a key. Although Bob would make only one copy of the key, which he would keep with him at all times, he would manufacture hundreds of padlocks, and distribute them to post offices all over the world. Then, if Alice wants to send a message to Bob, she would simply go to her local post office, ask for one of Bob's padlocks, then put the message in a box and finally lock it using Bob's padlock. The nature of padlocks is such that Alice (or anybody else) can easily lock Bob's padlock shut, but only Bob

has the key required to open the padlock. The key never leaves Bob, and so the key-distribution problem no longer exists.

Who discovered PKC?

The solution to the key-distribution problem has a certain 'make-you-kick-yourself' quality. What seemed impossible for thousands of years suddenly seems possible. The first people to announce that they had developed theoretical and then a practical approach to PKC were Whitfield Diffie and Martin Hellman at Stanford University, and Ronald Rivest, Adi Shamir and Leonard Adleman at MIT. When they published their research over the course of a few years in the mid-1970s, they soon became became cryptographic superstars. They had made one of the greatest contributions in the history of cryptography.

I have only described PKC in terms of an analogy with padlocks, and the real challenge for Diffie, Hellman, Rivest, Shamir and Adelman were to transform the padlock analogy into something mathematical that could be implemented on a computer and sent down a wire. They had to invent a mathematical padlock, something that involved a public formula that Alice could apply to scramble a message, but something that only Bob could unscramble because he had the mathematical key.

There are various websites that describe the mathematics of PKC.

Who really discovered PKC?

Without wanting to undermine the brilliant work of the US-based researchers usually associated with the discovery of PKC, I want to go back to the real origins of this form of encryption at the Government Communications Headquarters (GCHQ) in Britain. The true father of PKC was James Ellis, a brilliant, unpredictable and introverted cryptographer. His colleague Richard Walton recalls: "He was a rather quirky worker, and he didn't really fit into the day to day business of CESG. But in terms of coming up with new ideas he was quite exceptional. You had to sort through some rubbish sometimes, but he was very innovative and always willing to challenge the orthodoxy. We would be in real trouble if everyone in GCHQ was like him, but equally we need some people with his flair and originality."

Fully aware of the problems of key distribution, Ellis was the first to develop a theoretical form of PKC, but neither he, nor anyone else at GCHQ, could provide the necessary mathematics. Three years later, however, a pair of Cambridge graduates, Clifford Cocks and Malcolm Williamson, joined GCHQ and invented two separate techniques for implementing Ellis's idea. Together, Ellis, Cocks and Wiliamson had made the greatest breakthrough in twentieth century cryptography, but they could tell nobody about what they had done. Public-key cryptography was classified top secret.

Even though PKC was invented independently in America, then commercialized and made public, GCHQ remained silent about its own work throughout the 1980s and much of the 1990s. It was not until the summer of 1997 that GCHQ eventually decided that the true history of the invention of PKC would be explained at a major conference in December. Sadly, James Ellis, aged 73, died just a month before the world learned of his great contribution to the information age.

In 1987, Ellis wrote an internal GCHQ memorandum. It includes his thoughts on the secrecy that so often surrounds cryptographic work: "Cryptography is a most unusual science. Most professional scientists aim to be the first to publish their work, because it is through dissemination that the work realizes its value. In contrast, the fullest value of cryptography is realized by minimizing the information available to potential adversaries. Revelation of secrets is normally only sanctioned in the interests of historical accuracy after it has been demonstrated that no further benefit can be obtained from continued secrecy."

Simon Singh is a science writer and the author of "The Code Book" and "Fermat's Last Theorem".