Lessons for the Internet Age

Let's pause for a moment to consider some of the lessons of cryptographic history—morals that were well-understood by the early twentieth century. In the late twentieth century, cryptography changed drastically because of modern computer technology and new cryptographic algorithms, but these lessons are still true today. They are too often forgotten.

Breakthroughs Happen, but News Travels Slowly

Mary Stuart was beheaded when her letters plotting against Elizabeth were deciphered by frequency analysis, which Al-Kindi had described nine centuries earlier. Older methods have also remained in use to the present day, even for high-stakes communications. Suetonius explained the Caesar cipher in the first century A.D. Yet two millennia later, the Sicilian Mafia was still using the code. Bernardo Provenzano was a notorious Mafia boss who managed to stay on the run from Italian police for 43 years. But in 2002, some *pizzini*—ciphertexts typed on small pieces of paper—were found in the possession of one of his associates. The messages included correspondence between Bernardo and his son Angelo, written in a Caesar cipher—with a shift of three, exactly as Suetonius had described it. Bernardo switched to a more secure code, but the dominos started to topple. He was finally traced to a farmhouse and arrested in April 2006.

Even scientists are not immune from such follies. Although Babbage and Kasiski had broken the Vigenère cipher in the mid-nineteenth century, *Scientific American* 50 years later described the Vigenère method as "impossible of translation."

Encoded messages tend to look indecipherable. The incautious, whether naïve or sophisticated, are lulled into a false sense of security when they look at apparently unintelligible jumbles of numbers and letters. Cryptography is a science, and the experts know a lot about code-breaking.

Confidence Is Good, but Certainty Would Be Better

There are no guarantees that even the best contemporary ciphers won't be broken, or haven't been broken already. Some of the ciphers have the potential to be validated by mathematical proofs, but actually providing those proofs will require deep mathematical breakthroughs. If anyone knows how to break modern codes, it is probably someone in the National Security

Agency or a comparable agency of a foreign government, and those folks don't tend to say much publicly.

In the absence of a formal proof of security, all one can do is to rely on what has been dubbed the Fundamental Tenet of Cryptography: *If lots of smart people have failed to solve a problem, then it probably won't be solved (soon).*

Of course, that is not a very useful principle in practice—by definition, breakthroughs are unlikely to happen "soon." But they do happen, and when they do, indigestion among cryptographers is widespread. In August 2004, at an annual cryptography conference, researchers announced that they had been able to break a popular algorithm (MD5) for computing cryptographic operations called *message digests*, which are fundamental security elements in almost all web servers, password programs, and office products. Cryptographers recommended switching to a stronger algorithm (SHA-1) but within a year, weaknesses were uncovered in this method as well.

A provably secure encryption algorithm is one of the holy grails of computer science. Every weakness exposed in proposed algorithms yields new ideas about how to make them stronger. We aren't there yet, but progress is being made.

A provably secure encryption algorithm is one of the holy grails of computer science.

Having a Good System Doesn't Mean People Will Use It

Before we explain that unbreakable encryption may finally be possible, we need to caution that even mathematical certainty would not suffice to create perfect security, if people don't change their behavior.

Vigenère published his encryption method in 1586. But foreign-office cipher secretaries commonly avoided the Vigenère cipher because it was cumbersome to use. They stayed with simple substitution ciphers—even though it was well-known that these ciphers were readily broken—and they hoped for the best. By the eighteenth century, most European governments had skilled "Black Chambers" through which all mail to and from foreign embassies was routed for decryption. Finally, the embassies switched to Vigenère ciphers, which themselves continued to be used after information about how to crack them had become widely known.

And so it is today. Technological inventions, no matter how solid in theory, will not be used for everyday purposes if they are inconvenient or expensive. The risks of weak systems are often rationalized in attempts to avoid the trouble of switching to more secure alternatives.

In 1999, an encryption standard known as WEP (Wired Equivalent Privacy) was introduced for home and office wireless connections. In 2001, however, WEP was found to have serious flaws that made it easy to eavesdrop on wireless networks, a fact that became widely known in the security community. Despite this, wireless equipment companies continued to sell WEP products, while industry pundits comforted people that "WEP is better than nothing." A new standard (WPA—Wi-Fi Protected Access) was finally introduced in 2002, but it wasn't until September 2003 that products were required to use the new standard in order to be certified. Hackers were able to steal more than 45 million credit and debit card records from TJX, the parent company of several major retail store chains, because the company was still using WEP encryption as late as 2005. That was long after WEP's insecurities were known and WPA was available as a replacement. The cost of that security breach has reached the hundreds of millions of dollars.

Similarly, many of today's "smart card" systems that use RFID (Radio Frequency Identification) tags are insecure. In January 2005, computer scientists from Johns Hopkins University and RSA Data Security announced that they had cracked an RFID-based automobile anti-theft and electronic payment system built into millions of automobile key tags. They demonstrated this by making multiple gasoline purchases at an Exxon/Mobile station. A spokesman for Texas Instruments, which developed the system, countered that the methods the team used were "wildly beyond the reach of most researchers," saying "I don't see any reason to change this approach."

When encryption was a military monopoly, it was possible in principle for a commander to order everyone to start using a new code if he suspected that the enemy had cracked the old one. The risks of insecure encryption today arise from three forces acting in consort: the high speed at which news of insecurities travels among experts, the slow speed at which the inexpert recognize their vulnerabilities, and the massive scale at which cryptographic software is deployed. When a university researcher discovers a tiny hole in an algorithm, computers everywhere become vulnerable, and there is no central authority to give the command for software upgrades everywhere.

The Enemy Knows Your System

The last lesson from history may seem counterintuitive. It is that a cryptographic method, especially one designed for widespread use, should be regarded as more reliable if it is widely known and seems not to have been broken, rather than if the method itself has been kept secret.

The Flemish linguist Auguste Kerckhoffs articulated this principle in an 1883 essay on military cryptography. As he explained it,

> The system must not require secrecy, and it could fall into the hands of the enemy without causing trouble.... Here I mean by system, not the key itself, but the material part of the system: tables, dictionaries, or whatever mechanical apparatus is needed to apply it. Indeed, it's not necessary to create imaginary phantoms or to suspect the integrity of employees or subordinates, in order to understand that, if a system requiring secrecy were to find itself in the hands of too many individuals, it could be compromised upon each engagement in which any of them take part.

In other words, if a cryptographic method is put in widespread use, it is unrealistic to expect that the method can remain secret for long. Thus, it should be designed so that it will remain secure, even if everything but a small amount of information (the key) becomes exposed.

Claude Shannon restated Kerckhoffs's Principle in his paper on systems for secret communication: "... we shall assume that *the enemy knows the system being used*." He went on to write:

> The assumption is actually the one ordinarily used in cryptographic studies. It is pessimistic and hence safe, but in the long run realistic, since one must expect his system to be found out eventually.

Kerckhoffs's Principle is frequently violated in modern Internet security practice. Internet start-up companies routinely make bold announcements about new breakthrough proprietary encryption methods, which they refuse to subject to public scrutiny, explaining that the method must be kept secret in order to protect its security. Cryptographers generally regard such "security through obscurity" claims with extreme skepticism.

Even well-established organizations run afoul of Kerckhoffs's Principle. The Content Scrambling System (CSS) used on DVDs (Digital Versatile Disks) was developed by a consortium of motion picture studios and consumer electronics companies in 1996. It encrypts DVD contents in order to limit unauthorized copying. The method was kept secret to prevent the manufacture of unlicensed DVD players. The encryption algorithm, which consequently was never widely analyzed by experts, turned out to be weak and was cracked within three years after it was announced. Today, CSS decryption programs, together with numerous unauthorized "ripped" DVD contents, circulate widely on the Internet (see Chapter 6, "Balance Toppled" for a more detailed discussion of copy protection).

Kerckhoffs's Principle has been institutionalized in the form of encryption standards. The *Data Encryption Standard* (DES) was adopted as a national standard in the 1970s and is widely used in the worlds of business and finance. It has pretty much survived all attempts at cracking, although the inexorable progress of Moore's Law has made exhaustive searching through all possible keys more feasible in recent years. A newer standard, Advanced Encryption Standard (AES), was adopted in 2002 after a thorough and public review. It is precisely because these encryption methods are so widely known that confidence in them can be high. They have been subjected to both professional analysis and amateur experimentation, and no serious deficiencies have been discovered.

These lessons are as true today as they ever were. And yet, something else, something fundamental about cryptography, is different today. In the late twentieth century, cryptographic methods stopped being state secrets and became consumer goods.

Secrecy Changes Forever

For four thousand years, cryptography was about making sure Eve could not read Alice's message to Bob if Eve intercepted the message *en route*. Nothing could be done if the key itself was somehow discovered. Keeping the key secret was therefore of inestimable importance, and was a very uncertain business.

If Alice and Bob worked out the key when they met, how could Bob keep the key secret during the dangers of travel? Protecting keys was a military and diplomatic priority of supreme importance. Pilots and soldiers were instructed that, even in the face of certain death from enemy attack, their first responsibility was to destroy their codebooks. Discovery of the codes could cost thousands of lives. The secrecy of the codes was everything.

And if Alice and Bob never met, then how could they agree on a key without *already* having a secure method for transmitting the key? That seemed like a fundamental limitation: Secure communication was practical only for people who could arrange to meet beforehand, or who had access to a prior method of secure communication (such as military couriers) for carrying the key between them. If Internet communications had to proceed on this assumption, electronic commerce never could have gotten off the ground. Bit packets racing through the network are completely unprotected from eavesdropping.

And then, in the 1970s, everything changed. Whitfield Diffie was a 32-year-old mathematical free spirit who had been obsessed with cryptography since his years as an MIT undergraduate. 31-year-old Martin Hellman was a hard-nosed graduate of the Bronx High School of Science and an Assistant Professor at Stanford. Diffie had traveled the length of the country in search of collaborators on the mathematics of secret communication. This was not an easy field to enter, since most serious work in this area was being done behind the firmly locked doors of the National Security Agency. Ralph Merkle, a 24-year-old computer science graduate student, was exploring a new approach to secure communication. In the most important discovery in the entire history of cryptography, Diffie and Hellman found a practical realization of Merkle's ideas, which they presented in a paper entitled "New Directions in Cryptography." This is what the paper described:

> *A way for Alice and Bob, without any prior arrangement, to agree on a secret key, known only to the two of them, by using messages between them that are not secret at all.*

In other words, as long as Alice and Bob can communicate with each other, they can establish a secret key. It does not matter if Eve or anyone else can hear everything they say. Alice and Bob can come to a consensus on a secret key, and there is no way for Eve to use what she overhears to figure out what that secret key is. This is true even if Alice and Bob have never met before and have never made any prior agreements.

It was revealed in 1997 that the same public-key techniques had been developed within the British secret Government Communication Headquarters (GCHQ) two years before Diffie and Hellman's work, by James Ellis, Clifford Cocks, and Malcolm Williamson.

The impact of this discovery cannot be overstated. The art of secret communication was a government monopoly, and had been since the dawn of writing—governments had the largest interests in secrets, and the smartest scientists worked for governments. But there was another reason why governments had done all the serious cryptography. Only governments had the wherewithal to assure the production, protection, and distribution of the keys on which secret communication depended. If the secret keys could be produced by public communication, everyone could use cryptography. They just had to know how; they did not need armies or brave couriers to transmit and protect the keys.

Diffie, Hellman, and Merkle dubbed their discovery "public-key cryptography." Although its significance was not recognized at the time, it is the invention that made electronic commerce possible. If Alice is you and Bob is Amazon, there is no possibility of a meeting—how could you physically go to Amazon to procure a key? Does Amazon even *have* a physical location? If Alice is to send her credit card number to Amazon securely, the encryption has to be worked out on the spot, or rather, on the two separate spots separated by the Internet. Diffie-Hellman-Merkle, and a suite of related methods that followed, made secure Internet transactions possible. If you have ever ordered anything from an online store, you have been a cryptographer without realizing it. Your computer and the store's computer played the roles of Alice and Bob.

It seems wildly counterintuitive that Alice and Bob could agree on a secret key over a public communication channel. It was not so much that the scientific community had tried and failed to do what Diffie, Hellman, and Merkle did. It never occurred to them to try, because it seemed so obvious that Alice had to give Bob the keys somehow.

Even the great Shannon missed this possibility. In his 1949 paper that brought all known cryptographic methods under a unified framework, he did not realize that there might be an alternative. "The key must be transmitted by non-interceptable means from transmitting to receiving points," he wrote. Not true. Alice and Bob can get the same secret key, even though all their messages are intercepted.

Alice and Bob can get the same secret key, even though all their messages are intercepted.

The basic picture of how Alice communicates her secret to Bob remains as shown in Figure 5.6. Alice sends Bob a coded message, and Bob uses a secret key to decrypt it. Eve may intercept the ciphertext *en route*.

The goal is for Alice to do the encryption in such a way that it is *impossible* for Eve to decrypt the message in any way other than a brute-force search through all possible keys. If the decryption problem is "hard" in this sense, then the phenomenon of exponential growth becomes the friend of Alice and Bob. For example, suppose they are using ordinary decimal numerals as keys, and their keys are ten digits long. If they suspect that Eve's computers are getting powerful enough to search through all possible keys, they can switch to 20-digit keys. The amount of time Eve would require goes up by a factor of 10^{10} = 10,000,000,000. Even if Eve's computers were powerful enough to crack any 10-digit key in a second, it would then take her more than 300 years to crack a 20-digit key!

Exhaustive search is always *one* way for Eve to discover the key. But if Alice encrypts her message using a substitution or Vigenère cipher, the encrypted message will have patterns that enable Eve to find the key far more quickly. The trick is to find a means of encrypting the message so that the ciphertext reveals no patterns from which the key could be inferred.

The Key Agreement Protocol

The crucial invention was the concept of a *one-way computation*—a computation with two important properties: It can be done quickly, but it can't be undone quickly. To be more precise, the computation quickly combines two numbers x and y to produce a third number, which we'll call $x * y$. If you know the value of $x * y$, there is no quick way to figure out what value of y was used to produce it, even if you also know the value of x. That is, if you know the values of x and the result z, the only way to find a value of y so that $z = x * y$ is trial and error search. Such an exhaustive search would take time that grows exponentially with the number of digits of z—practically impossible, for numbers of a few hundred digits. Diffie and Hellman's one-way computation also has an important third property: $(x * y) * z$ always produces the same result as $(x * z) * y$.

The key agreement protocol starts from a base of public knowledge: how to do the computation $x * y$, and also the value of a particular large number g. (See the Endnotes for the details.) All this information is available to the entire world. Knowing it, here is how Alice and Bob proceed.

1. Alice and Bob each choose a random number. We'll call Alice's number a and Bob's number b. We'll refer to a and b as Alice and Bob's *secret keys*. Alice and Bob keep their secret keys secret. *No one except Alice knows the value of a, and no one except Bob knows the value of b.*
2. Alice calculates $g * a$ and Bob calculates $g * b$. (Not hard to do.) The results are called their *public keys* A and B, respectively.
3. Alice sends Bob the value of A and Bob sends Alice the value of B. It doesn't matter if Eve overhears these communications; A and B are not secret numbers.
4. When she has received Bob's public key B, Alice computes $B * a$, using her secret key a as well as Bob's public key B. Likewise, when Bob receives A from Alice, he computes $A * b$.

Even though Alice and Bob have done different computations, they have ended up with the same value. Bob computes $A * b$, that is, $(g * a) * b$ (see Step 2—A is $g * a$). Alice computes $B * a$, that is, $(g * b) * a$. Because of the