Cryptography old and new

Join **Jonni Bidwell** on a journey of code making and breaking, mystery and intrigue...

For as long as there have been stories there have been secrets – words unspoken for tactical advantage or for fear of reprisal. Secrets often need to be sent afar, and their remaining secret en route is of paramount importance. So it was when Xerxes' attack on Sparta was thwarted by Demaratus (a Greek exile living in Persia, whose warning message was sent to Sparta hidden on an apparently blank wax tablet). And so it is when you send your credit card details across the ether to pay for gadgets, snacks or socks.

Most people will likely be familiar with a substitution cipher, in which one letter is replaced by another. The best-known of these is the Caesar cipher, in which each letter is replaced by one a fixed distance further down the alphabet, wrapping around when one runs out of letters. It is said that Julius Caesar used this method, replacing A with D, B with E, and so on, wrapping around with A replacing X, whereas his nephew Augustus favoured a shift of just one letter, in which A is replaced by B, B by C etc, but with no wraparound, so that Z is replaced by the symbol AA.

The *Kama Sutra* also describes, among other rather more interesting tricks, the art of mlecchita-vikalpa (secret writing). It details a substitution cipher in which letters are paired and interchanged by a fixed random scheme, so that lovers can "conceal the details of their liaisons". An even older substitution system is Atbash, originally found in old (circa 500 BC) Hebrew texts. Here the first letter of the alphabet, aleph, is replaced by the last, tav; the second, beth, by the second to last, shin, and so on, effectively reversing the alphabet. The latinic equivalent is interchanging A and Z, B and Y, and so forth. The ROT13 system (a Caesar cipher with a shift of 13) is still used on some websites and newsgroups to obfuscate plot spoilers, punchlines or naughty words.

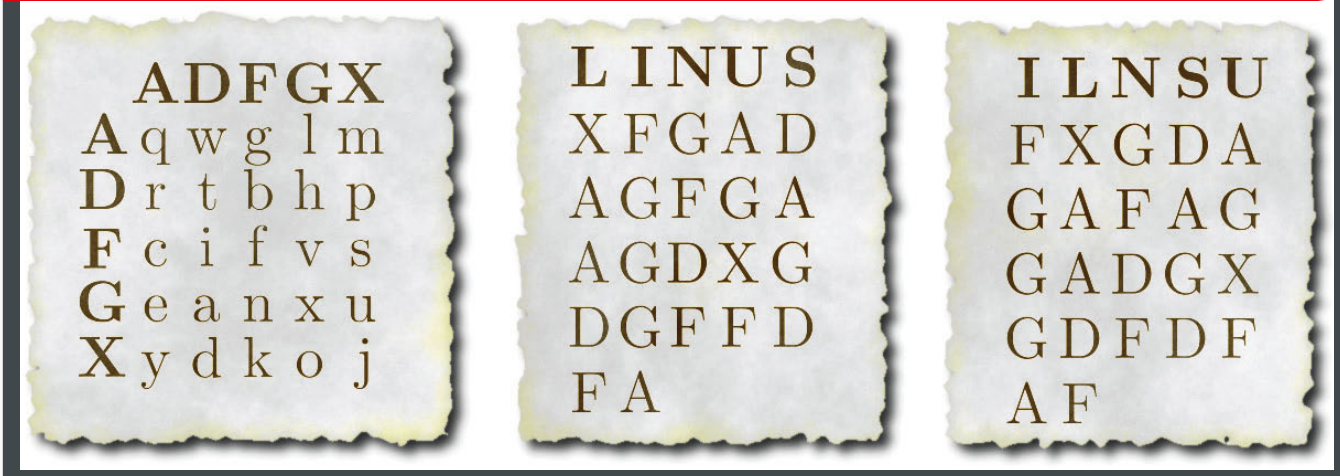
These monoalphabetic substitution ciphers (MSCs) are not in any way cryptographically secure by today's standards, but in their time they were likely effective enough – the highway bandits of Caesar's time being likely illiterate, unlike the masterful wordsmiths of the modern internet. These ciphers do contain a germ of the idea of the modern cryptographic key, though. Whether it's the length of the shift in a Caesar cipher, the dimensions of the Scytale, or the pairings used in the *Kama Sutra* (no, not *those* pairings), knowledge of the method of encryption, together with the key, allows one to decipher the message.

> **"The Kama Sutra describes, among other more interesting tricks, the art of secret writing."**

We have 26 possible keys (including the trivial zero-shift) for a Caesar cipher, whereas ROT13 and Atbash are essentially single-key systems. The *Kama Sutra* cipher has a fairly large keyspace – there are about 8 trillion (8 followed by 12 zeroes) unique ways of pairing the alphabet. The general MSC has an astounding number of possible combinations (26 factorial – about 4 followed by 26 zeroes – or a little more than 88-bits in modern binary terms), but size isn't everything... The Arab polymath Al-Kindi, in a ninth-century manuscript titled *On Deciphering Cryptographic Messages*, gave the first description of breaking MSCs by frequency analysis – exploiting the fact that in an 'average' message, some letters will occur more frequently than others.

For example, in English the letter 'e' occurs with a relative frequency of about 13%, followed by 't' with 9%, and so on. This is why Scrabble scoring is the way it is – the more common the letter, the less it scores. Other languages have different letters and frequencies, but the principle remains the same: replace the most frequently occurring letter in the ciphertext with the most frequently occurring letter in the language, then repeat for the next most frequent letter, and continue until you are able to fill in the blanks. The original message might not have exactly the same letter frequencies as the language, but provided it's long enough it will at least be close enough that decryption will be possible with a little tweaking.

Don't panic, Colonel



This triptych shows another WWI example: the ADFGX cipher (these letters were chosen because they're different in Morse code). The first plate is the fractionating key: it encodes each letter of our alphabet (sans the letter z because the **LXF** style guide doesn't like it) into a bigram, so that our message 'kernel panic' encodes to XF GA DA GF GA AG DX GD GF FD FA (the space is ignored). In the second plate, we fit this message onto a grid below a second keyword, 'LINUS', which is our transposition key. In practice, a longer transposition key would have been used, and both keys would be changed according to a daily code book. We rearrange the columns by putting the second key in alphabetical order, and then read off the ciphertext column-wise. Thus our encoded message is FGGGA XAADF GFDF DAGD AGXF.

The discovery of the 1586 Babington Plot (which sought to assassinate Queen Elizabeth I) led to Mary Queen of Scots and her co-conspirators being executed after their correspondence was decrypted by renowned codebreaker Thomas Phelippes. Letters between Mary and Babington had been encrypted by substitution using symbols mostly from the Greek alphabet, and Phelippes was able to forge an addendum to one of Mary's letters requesting the identities of the co-conspirators. Once they were thus incriminated, heads were off'd.

A milestone in the history of cryptography was the invention of the so-called Vigenère cipher in 1553. This was actually the work of cryptologist Giovan Battista Bellaso, who built on the ideas of Trithemius and Alberti. Vigenère did in fact publish a stronger autokeying cipher in 1586, but history has misattributed this earlier cipher to him. The cipher is a polyalphabetic substitution cipher which uses a keyword to switch cipher alphabets after each letter. Each letter is encrypted by a Caesar cipher with shift determined by the corresponding letter of the keyword. This (providing the keyword has more than one unique letter) thwarts traditional frequency analysis. The cipher was considered so strong that it was dubbed *le chiffre indéchiffrable*, and indecipherable it remained until work by Babbage and Kasiski in the mid-19th century. Their efforts centred on isolating the length of the key: once that is known then the ciphertext can be separated into as many chunks; each chunk will be encrypted by a different Caesar shift, which is easily dealt to by frequency analysis.

Later, this cipher was augmented with the letter V to make the imaginatively-titled ADFGVX cipher. In 1918, in a phenomenal tour-de-force, the French cryptanalyst Georges Painvin managed to decrypt an ADFGVX-encrypted message which revealed where the German forces were planning to attack Paris. Painvin lost 15kg of body weight over the course of this crypto-toil.

One may wonder if anyone can make a truly unbreakable cipher, and one may be shocked to learn that such a thing already exists. That it has been patented since 1917 may leave one so utterly aghast as to impinge permanently on one's health, but this is fact nonetheless. The chap responsible (for the patent at least) was Gilbert Vernam, and his invention is known as the One Time Pad. The trick is to ensure that there is as much key material as there is plaintext, that the key material is entirely random and perfectly secret, and no part of the key material is used more than once. In practical terms, though, Vernam's system is largely useless. Generating truly random material is difficult, as is distributing a huge amount of it in secret and ensuring its destruction post-use.

Enigmatic mathematics

Wartime cryptography relied heavily on codebooks which contained daily keys, and these had a bad habit of falling into enemy hands. Once such a breach occurred and news of it reached HQ, generals were faced with the tremendous logistical problem of alerting relevant personnel as to the breach and then manufacturing and distributing new key material. Long-range naval missions often failed to receive this, necessitating that messages be retransmitted using old keys. This exchange was sometimes intercepted, providing clues as to the new key. During World War I, the decrypting of the Zimmerman telegram (which invited Mexico to ally with Germany) was instrumental to American involvement in the war.

By World War II the Germans had upgraded the Enigma series of machines to present a sufficient cryptographic challenge to Bletchley Park. Polish researches had broken the original design as early as 1932, and just prior to the outbreak of war they shared their intelligence with the British. Alan Turing designed the Bombe machine, which by 1940 was doing a fine job of breaking Jerry comms.

The Enigma machine, despite having a huge number of rotor, plugboard and stecker settings, had a weakness in that a letter was never encrypted to itself. This vastly reduced the amount of work that the Bombe and the computers (usually women with a good eye for detail and skill at crossword puzzles) had to do. After a letter was typed on the Enigma, the cipher alphabet was changed by the rotor mechanism, in a manner not dissimilar from the Vigenère cipher. There were other layers of encryption too, but a lot of these were constant settings made redundant when Enigma machines were captured. By the end of the war there were around 200 Bombes in use throughout England. The Americans, being in a much better position for obtaining supplies, were able to build and design 125 much faster Bombes, and the Allies were able to farm out work to these remote behemoths via (encrypted) cable.

»

Turing's genius notwithstanding, much of the Enigma traffic was decrypted thanks to sloppy operational security. Message keys could have been changed with every transmission but were not, or when they were the change was only slight and easily guessed. Numbers were often spelled out, so 'einsing' was a common technique – looking for occurrences that might decrypt to 'eins'. If numerals had been allowed, this technique would have failed.

In the 1970s, two developments brought the cryptography game into the computer age. The first of these developments was the Data Encryption Standard, a block cipher based on work by Horst Feistel at IBM. Prior to its standardisation, it was slightly modified at the behest of the NSA. With no reasons being cited for these agency-mandated changes, suspicions were raised about a possible back door. Two decades later, it emerged that the opposite was true: the S-boxes of the original cipher were susceptible to a technique called 'differential cryptanalysis', which at the time (cryptography being considered a munition) was classified. The NSA changes made the cipher more resistant to the technique, although they did also recommend a smaller 48-bit, as opposed to 64-bit, key size. Being the first publicly available cipher, DES became the subject of intense scrutiny and in many ways bootstrapped serious academic study of cryptography.

While the thousands of pages of journal articles on the subject provide all manner of theoretical attacks on DES, by far its most serious weakness is the short key size. IBM and the NSA eventually compromised on a nominal 64-bit key, but eight of these 64 bits were redundant checksum bits. At the time of its introduction this was probably sufficient, but in the early 1990s machinery was proposed that could brute-force a key within hours. In 1997 an Internet-wide project successfully cracked a DES key for the first time. In 1998, the Electronic Frontier Foundation built a device (for a princely $250,000) which successfully cracked a key in a little over two days.

Among the other attacks on DES it's worth mentioning Matsui's 'linear cryptanalysis'. The attack involves building up approximations to parts of the cipher by finding modulo 2-linear expressions that hold with a probability significantly different from 0.5. By collecting a huge number (2^{43}) of plaintext-ciphertext pairs, one can deduce a sufficient number of bits of the key that the remainder can be brute-forced. Linear expressions can be found speedily thanks to the Walsh-Hadamard transform, and modern ciphers all are very careful to include a heavily nonlinear component to mitigate against these attacks. In some ways one can look at Matsui's work as an abstraction of basic letter frequency analysis, using characteristics of the cipher rather than the language, and 1s and 0s rather than characters.

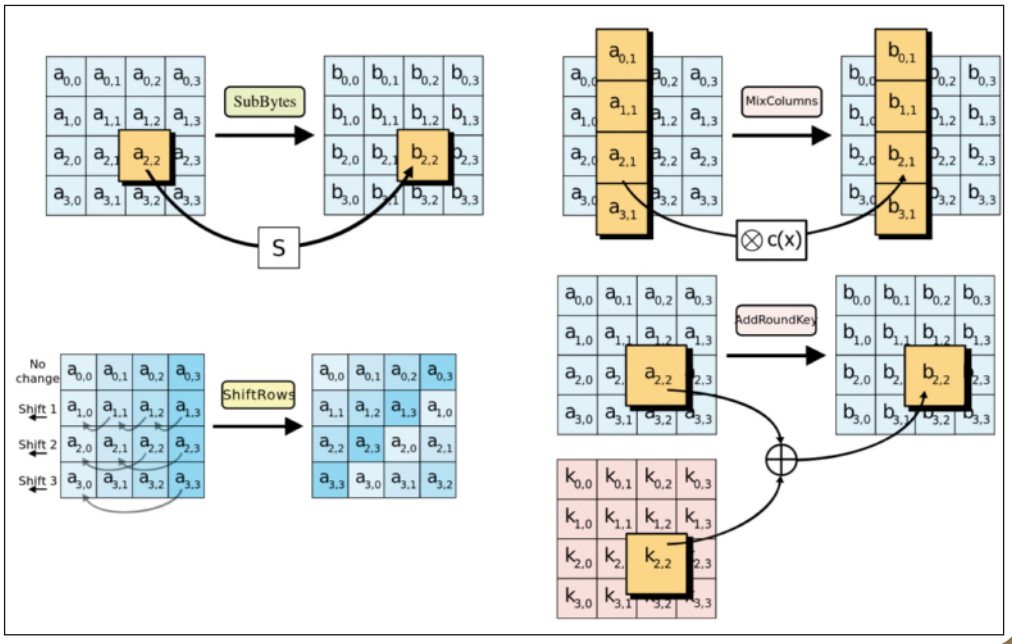
Going public

The other good thing to come out of the '70s was Public Key Cryptography. This finally solved the problem of being able to communicate securely without first having to meet in order to establish a shared secret. The method is called the Diffie-Hellman key exchange, after the gentlemen responsible for its invention. It exploits the chiral mathematics of finite fields, in which it's straightforward to exponentiate an element (that is, raise a number to a power), but very difficult to conduct the opposite process, known as the discrete logarithm. Thus field exponentiation is an example of a 'one way function'. The illustration (at the foot of the facing page) shows an example of the exchange between Alice and Bob, who are fairly ubiquitous in cryptographic literature. The shared secret $s=g^{ab}$ can be calculated by both Alice and Bob. An onlooker, Oscar say, can see the public keys *A* and *B*, and the exchange parameters *g* and *p*, but these are of no help in deducing the shared secret *s* unless one of the secret keys *a* or *b* is also known.

Once thusly established, the shared secret *s* can be used as an ephemeral encryption key for a symmetric cipher, such as DES. The secret keys *a* and *b* could at this point be destroyed, which would ensure so-called perfect forward secrecy, but a proper public key infrastructure would require that private and public keys remain largely immutable. Further, public keys should be as well-advertised as possible, to reduce chances that a man in the middle, say Mallory, could impersonate either party with a bogus public key: the key exchange provides confidentiality, but doesn't of itself guarantee authenticity. To achieve the latter, one needs to be sure of whose public keys belong to whom. To do this in general, one requires a trusted third party,

Advanced Encryption Standard

AES was introduced as a replacement for DES in 2001. To date it has defied all cryptanalytic efforts to find weaknesses. One reason for its selection was its relatively simple structure. There are four main layers, repeated over several rounds. With a bit of imagination, one can see echoes of the ADFGX cipher in the ShiftRows stage. The SubBytes stage is the only non-linear part of the cipher. Typically linear operations are much quicker to carry out, but without a non-linear stage a cipher will be trivial to break using the methods introduced by Matsui.



Development of modern principles

Over the last 150 years, a few key principles have been developed which (with small adjustments to allow for new technologies) still give a good idea of what the cryptography game is all about. The first is Kerckhoffs's [this apostrophe catastrophe brought to you by Wikipedia] principle: that knowledge of the encryption method alone should not be considered a threat to the security of the message. So long as the key is not compromised, this knowledge will be of no help. This is counter to the idea of security by obscurity, which, although it intuitively might seem reasonable, is considered bad form nowadays. The CSS copy-protection system used on DVDs was broken in 1999 after reverse engineering of the *Xing* software revealed a player key and the underlying algorithm (which turned out to be woefully poor). Likewise, the KeeLoq mechanism for remotely unlocking vehicles was broken in 2006 after part of its design was leaked.

Claude Shannon is often called the founder of Information Theory. In 1949 he introduced the ideas of Confusion and Diffusion for ciphers. Confusion advocates that the relationship between plaintext, ciphertext and key should be as complicated as possible. In terms of modern block ciphers this should mean each output bit depends in a non-linear manner on several key- and input bits. Diffusion refers to the idea that changing one key- or input bit should have a fairly drastic effect on the output. Ideal diffusion results in the strict avalanche criterion: that each output bit should change with probability 0.5 when one key- or input bit is flipped.

known as a Certificate Authority (CA), to act as a directory of keypair owners.

Since public key cryptography is such a different animal from its private counterpart, one can use various bits of mathematical trickery to reduce the search space to one significantly smaller than that of a brute-force attack. This being so, the classic public key algorithms all have much longer keys. For example, the AES algorithm is considered secure with a 128-bit key, but people are already concerned that 1,024-bit RSA keys are no longer secure. The new-fangled Elliptic Curve cryptography, based again on discrete logarithms but in a more abstract algebraic space, offers shorter keys, but still of the order of twice the security parameter.

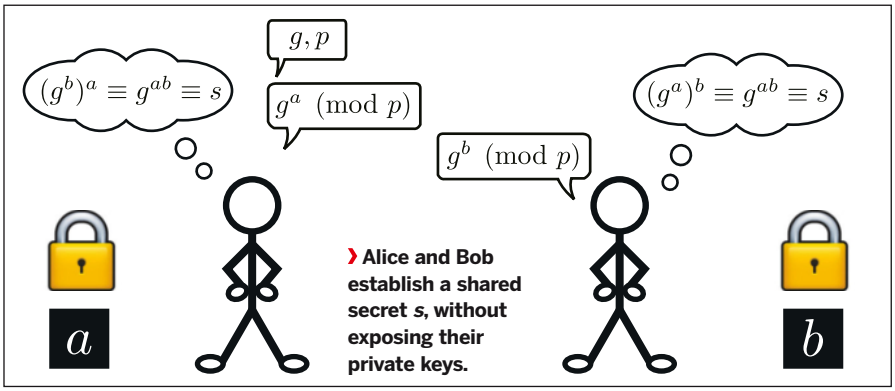
The security of all these public key systems rests on the supposed intractability of factoring integers and the discrete logarithm problem. While mathematicians have studied these problems extensively and come up with some good tricks for speeding up the process, they both remain sufficiently time-consuming to solve as to still be considered secure – at least on conventional hardware.

Up until 1992 cryptographic software was classified as a form of munitions in the US, and even after this date was governed by export restrictions. These precluded the export without licence of any software using a key length of more than 40 bits. This led to a lengthy criminal investigation of PGP founder Paul Zimmerman, which ended in nought. Zimmerman came up with novel ways of circumventing these restrictions, including publishing the source code as a book, protected by the First Amendment. Netscape was forced to release a crippled 'International Edition' which permitted only 40-bit SSL keys, in contrast to its 128-bit US edition.

Are you Shor?

In 1994, Peter Shor announced an algorithm which could be run on a quantum computer which would enable it to (among other things) factor integers and compute discrete logarithms much faster than a classical computer. While no one has yet succeeded in building the right kind of quantum computer, there's sufficient concern to give rise to a burgeoning field of study known as post-quantum cryptography.

Perhaps a more practical concern is the problem of producing secure keys in the first place. This relies on being able to produce a sufficiently random stream of bits, which computers are notoriously bad at. On Linux we have the **/dev/random** and **/dev/urandom** nodes (go on, run the *cat* command on them), which both harvest entropy gathered from (among other sources) keyboard and mouse input in order to augment a pseudorandom number generator (PRNG). This is why it's good practice to make erratic mouse gestures and batter the keyboard when running, for example, the *ssh-keygen* command.



Alice and Bob establish a shared secret *s*, without exposing their private keys.

A very early version of *Netscape* contained a weak PRNG that was seeded using the time of day and process ids. Since an attacker would be able make educated guesses as to these variables, the supposedly randomly generated SSL keys could be broken. In 2008 sysadmins were sent into a widespread panic when it was revealed that OpenSSL was generating weak keys, and had been doing so for two years. More recently, Ed Snowden has revealed that the NSA paid RSA security to use a generator called *Dual EC DRBG* as the default in their software. The constants that the NSA recommends to initialise this generator with are suspected to have been contrived in such a way as to provide a back door into the algorithm.

Besides ciphers, an important concept is that of a hash function. This scrambles an input to a fixed length output (so if the input is longer than the output there could be collisions) in a one-way manner. Hashed passwords in Linux are stored in **/etc/shadow**. Originally the MD5 hashing algorithm was used, but nowadays SHA-512 is becoming the standard. Often we hear news of hackers managing to obtain databases, which often contain hashed passwords. If you are in possession of a large database, the popular *John the Ripper* password cracker is able to weed out any weak passwords in a matter of minutes. For research purposes we ran it on a real world database (which has several thousand users), and managed to get 2,500 passwords over the course of a few hours. Other tools such as *oclHashcat* can leverage GPU power as well, so database security is important, as is changing your password if it is compromised.

In sum, we have seen great changes in how we encrypt our secrets, but it's important to see how we have been inspired by the past. Unfortunately, we make the same mistakes too – whenever security is breached, it is far more likely to be due to poor security practice than weaknesses in the cipher. Misconfigured servers, phishing attacks, malicious or lazy operators are by far the greater problem. LXF