Make it

Advanced code-breaking

Cryptology tools are on your SuperDisc

For the full contents of your SuperDisc, see page 114

All previous articles are provided FREE on your SuperDisc

Breaking the toughest codes

From the World War II Enigma to today's three-star ciphers, we try to discover the unbreakable

You'll need this...

- **ENIGMA SIMULATOR V6.4**
  This emulator of the historic Enigma encryption machine lets you encrypt and decrypt messages, and also see how the machine works. Download it from **www.bit.ly/enisim**.
- **CRYPTOOL**
  This educational tool allows you to try out and learn about ciphers. You can get it from **www.cryptool.com**.
- **QUICKSTEGO**
  Use this free software (available from **www.quickcrypto.com/page22.htm**) to hide messages in images.

More on this topic

The previous parts to this series can be found in the Makeit folder on your SuperDisc.

- Last issue:
  We used Excel to explore the basics of cryptography.
- This issue:
  We tackle some of the more secure ciphers and crack the Enigma Code.

Last issue we started our tour of the science of cryptology by investigating some of the earliest ciphers. First we saw the Caesar Cipher, which shifts each letter by a fixed number of places in the alphabet. However, since there are only 25 different positions by which letters can be shifted, messages are easily cracked by trying out each of the keys in turn. We then turned our attention to the general mono-alphabetic substitution cipher in which any letter can be translated into any other letter. There are so many possible keys that the brute-force method of cracking a message would take billions of years, even with the fastest supercomputer. Regardless, we discovered that messages can easily be cracked by analysing the frequency of occurrence of the letters in the ciphertext.

We concluded part one by introducing the poly-alphabetic substitution cipher that overcomes this drawback by using several keys and cycling between them. Because a particular letter doesn't always end up as the same letter when encrypted, frequency analysis can't easily be used to detect common letters such as Es and Ts or letter pairs such as TH. One such cipher – the Vigenère cipher – become known as 'le chiffre indéchiffrable' (which in English means 'the uncrackable'). It hasn't lived up to this label, though, as we're about to see.

Cracking Vigenère

The Vigenère cipher is a particular instance of a poly-alphabetic substitution cipher based on the Caesar Cipher. It uses a keyword (because it's easy to remember) that dictates the sequence of Caesar shifts. So if the keyword was SHIFT, letters would be shifted 18 places (S being 18 places ahead of A in the alphabet), then seven (H being eight places ahead of A in the alphabet), then eight, then five, and then 19, before cycling back to 18 again.

The first job in cracking the Vigenère cipher is to determine the key length, which, in turn, defines the repeat cycle. The original method involved looking for repeating letter groups. So if we discovered that APJ occurs several times in the ciphertext, it's likely – although not certain – that it represents the same three letter group in the plaintext (perhaps THE, which is the most common three-letter group in the English language). If this is so, the repeats will be separated by a multiple of the key length.

By looking for several such repeating groups, and bearing in mind that the key is normally fairly short, it's possible to guess the key length. Once the key length is known, and bearing in mind that there are only 26 possible shift values, it's comparatively easy to unlock the cipher. Even the more general poly-alphabetic substation cipher could be cracked using frequency analysis. So, for our key length of five, frequency analysis would be used separately on letters 1, 6, 11, 16, 21, letters 2, 7, 12, 17, 22 and letters 3, 8, 13, 18, 23 and so on. ▸

Spotlight on... The one-time pad

The story of cryptology is one of ever-more complicated ciphers as everyone tries to stay one step ahead of advances in cryptanalysis. Ironically, therefore, there's a totally uncrackable cipher that really couldn't be much simpler.

If a cryptographic key is shorter than the message, the ciphertext will contain some patterns that, given sufficient computing power, could be exploited by a cryptanalyst. The solution is probably obvious – use a key that's as long as the message and is never reused. This form of cryptography employs something called a one-time pad: a book containing random numbers from 0 to 25. The first number is used to implement a Caesar Shift on the first letter to be encrypted, the second number is used for the second letter and so on. So long as those numbers are genuinely random (so software-generated pseudo random numbers are out of the question), the resultant ciphertext can only be decrypted by someone with another copy of the pad. Cracking it is impossible.

One-time pads have been used in espionage but, despite their 100 per cent security, they have one important drawback. The security is only guaranteed if the originator of the pad can get a copy to the other party in the absolute certainty that nobody else has seen it. Delivering it by hand – potentially an expensive proposition – is the only option. ■

▲ **The one-time pad might be low tech but it provides totally secure communication – at least in theory.**

Do this...

Hide messages in images

An alternative to cryptography is steganography. This involves hiding a message in something innocuous in such a way that someone intercepting it doesn't realise it contains concealed information. An early example is hiding text within a letter such that the hidden message can be read by extracting, for example, every 10th letter.

Bringing this up to date, messages can be hidden within images. This could be done by reducing the colour depth of a 24-bit TIFF image to 21 bits. The reduction from 16.7 million to 2 million wouldn't be noticed if viewed normally, but three bits per pixel would be freed up and these could be used to store a message.

You can try this using QuickStego, which will both hide a message in an image and subsequently reveal that message. Just open an image and a text file, click on 'Hide Text' and then 'Save Image'. ■

▲ **QuickStego can hide secret messages in an image such as a JPEG file.**

► If you doubt that the Vigenère cipher can be cracked so easily, why don't you put an automatic Vigenère cracker to the test? First you need some Vigenère ciphertext of a reasonable length. To give the cryptanalysis software a fighting chance, make sure the keyword isn't too long. We suggest at least 250 words of plaintext and a key no longer than six letters.

Unless you particularly want to encrypt the text by hand or write an Excel workbook, use the online Vigenère encrypting tool at **www.sharkysoft.com/misc/vigenere**. To simplify things, just copy some text from the web and then keep the punctuation and make all the letters upper case. You could also format the text into five-letter groups.

When you've encrypted the text, copy and paste it into the Vigenère cracker at **www.bit.ly/aMZJho**. Make sure that the specified range of key lengths includes the length of key you used and click on 'Analyse ciphertext'. If our experiences are typical, there's a very good chance that it'll reproduce your keyword and go on to decrypt your ciphertext. So much for le chiffre indéchiffrable...

Enigma cribs

An important trick in cracking Enigma was to guess at words or phrases that an encrypted message might contain. These were called cribs. When this wasn't possible, the British would sometimes 'leak' information that they were pretty confident would then turn up in German Enigma-encrypted messages. ■

Introducing Enigma

Clearly, a poly-alphabetic substitution cipher becomes more difficult to crack as the repeat cycle increases in length. In the case of ciphers like Vigenère that are encrypted and decrypted by hand, the repeat cycle was kept fairly short for obvious reasons. But with mechanisation, the possibility for errors is reduced and, as a result, a much longer repeat cycle becomes practical. The pinnacle of achievement in poly-alphabetic substitution ciphers could perhaps be the Enigma machine, which was used by German forces for encrypting messages before transmission by Morse Code during World War II.

▲ **This Enigma Simulator both looks and works like the famous WWII electromechanical cipher machine.**

▲ **The Enigma Simulator also provides us with a glimpse of what went on behind the scenes.**

A familiar name to most, Enigma was an electromechanical machine that featured a keyboard and an array of lights, one for each letter. Encrypting a message involved typing it on the keyboard and recording the sequence of letters displayed on the lights for each key depression. The electrical connection between the keys and the lights was both complicated and constantly changing, as illustrated in the simplified wiring diagram on page 67 of an Enigma variation which, for clarification, has only four keys and four lamps (for the letters A, B, C and D).

When a key is pressed an electrical signal is routed to a patch panel where some, but not all, of the letters are translated to a different letter. Next it progresses to the first rotor, which translates each letter to a different letter. The signal from this first rotor progresses to a second rotor, where a similar but different translation takes place. The signal from the second rotor passes to a third rotor, which also creates a translation. The signal now progresses to the reflector, where it's routed back to a different position in the third rotor. This passes back through the second and finally the first rotor, from where it's routed back through the patch panel and finally to the lamps.

Where things really get complicated, though, is that the first rotor moves on one position after each key depression, the second rotor progresses after every full rotation of the first rotor, and the third rotor progresses after every full rotation of the second rotor.

In the case of our simplified Enigma, which has just four positions on each rotor, the machine would return to its starting position every 4 x 4 x 4 (so 64 in total) letters. With the complete Enigma, though, there are 26 x 26 x 26 combinations of the three rotors, so the machine only returns to the same position every 17,567 letters.

▲ **The Vigenère cipher – once thought to be ultimately secure – is now far from uncrackable, as this online utility quickly proves.**

To get a feel for this ground-breaking machine, try out Enigma Simulator v6.4, which provides a good representation of the real machine at a function level and also shows how the machine actually looked. The simulator comes with a comprehensive manual from which you can learn all its intricacies. However, you don't need to consult the manual to start encrypting a message, noting that the same plaintext letter rarely ends up as the same ciphertext letter, and that the rotors do indeed rotate with each key depression.

For now all we're going to say about cracking Enigma is that it involved a massive effort by some of Britain's most talented mathematicians, perhaps most notably computer pioneer Alan Turing. It was the first instance of automation being used in cryptanalysis, the machine in question being a specialised electromechanical computer called the Bombe. The logic behind the process is involved, as is the working of the Bombe, but if you do want to delve further then there's a Bombe simulator available for you to have a play with at **www.bit.ly/bTKKTY.**

Spotlight on… Quantum cryptography

As you can read in the 'The one-time pad' box on page 65, the one-time pad technique is 100 per cent secure but there are practical problems with key distribution. This has restricted its use to military espionage.

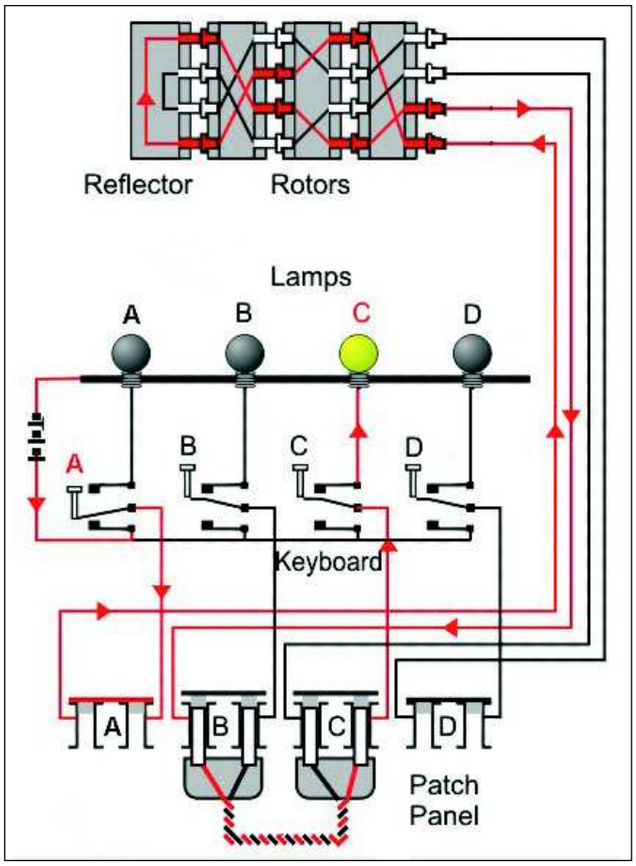
Quantum cryptography is the one-time pad brought up to date by allowing the key – basically a long string of random numbers – to be transmitted securely over a communication link. Normally any such transmission would pose a serious risk of the key being intercepted. But the strange properties of fundamental particles such as photons mean the key can be transmitted in such a way that the recipient is able to tell if the key has fallen victim to an eavesdropper. However, the fact that today's technology requires either a fibre optic or a laser link for it to work, and the range is limited, means that it'll be some time yet before this technique becomes widely available. ■

▲ **Quantum cryptography currently requires laser beams and precision optical components.**

Modern ciphers

To complete our whistle-stop tour of the history of cryptology, we're now going to jump forwards to a cipher that's representative of today's state-of-the-art ciphers: DES (Data Encryption Standard). Developed by IBM in the 1970s, DES uses a 56-bit key to operate on blocks of 64 bits of plaintext. It's no longer considered secure because the phenomenal increase in computer speed since the '70s means that it's susceptible to a brute-force attack. This involves trying out all of the 72,057,594,037,927,936 possible keys. However, it's still used as part of the triple-DES algorithm that involves encrypting a message three times, each time employing DES with a different key, and it also provides an inkling of the level of complexity involved in today's more secure ciphers.

To cut a long story short, encrypting a message using DES involves repeating a sequence of cryptographic transformations 16 times, in each case using a different sub-key derived from the full 56-bit key. That sequence of transformations includes bit-level substitutions, exclusive OR-ing with the sub-key and bit-level transpositions. The end result is that the 64-bits of data in the block are well and truly scrambled.



▲ **This wiring diagram shows how the letter A translates to C in a simplified version of the Enigma machine.**

Cracking DES

The 56-bit DES cipher was first publicly cracked in 1997 in 140 days using a distributed computing architecture – in other words, harnessing the power of lots of PCs communicating over the web. In 1999, using specially designed hardware, a message was cracked in under 24 hours. ■

We declined even to try to create an Excel workbook to demonstrate DES (although it has been done – see **www.bit.ly/95YBQs**). If you want to get a feel for how it works, we suggest you use CrypTool. From the Indiv. Procedures menu, select 'Visualisation of Algorithms' and then 'DES…'. The animation will start up in a separate window that you should maximise before stepping through the tutorial using the navigation controls in the bottom left-hand corner.

You can also use CrypTool to encrypt messages in DES and lots of other ciphers. Select 'New' from the File menu and an empty text box will appear. Type your plaintext in here and, if you want to view it as hexadecimal – which is a more appropriate way to view data that will be encrypted using a block cipher like DES – select 'Show as Hexdump' from the View menu. Each two-character hexadecimal number represents eight bits, so each line shows two 64-bit blocks. Now, in the Crypt/Decrypt menu select 'Symmetric (modern)' and then 'DES (ECB)…'. In the Key Entry: DES (ECB) dialog box, enter the key as eight two-character hexadecimal numbers (this is really 64 bits, but the algorithm will only use 56 of them) before clicking on 'Encrypt'. A new window will display the resulting ciphertext. PCP

Although he remains fascinated with cryptology, Mike Bedford drew the line at implementing modern ciphers like DES in Excel. ***feedback@pcplus.co.uk***