Playfair Ciphers

Playfair ciphers are a type of *block cipher*: the ciphertext character that replaces a particular plaintext character in the encryption will depend in part on an adjacent character in the plaintext. They are named for an English lord, Lyon Playfair, who advocated their use, but they were invented by Charles Wheatstone (1854).

Encryption is accomplished using a square array of characters, constructed from the encryption key. Because our set of plaintext characters is the 26-letter English alphabet, for us this array will be 5×5, with 2 of the 26 characters occupying a single position in the array. Typically, these two characters are i and j, since usually it is easy to distinguish from the context which of these two letters was intended in the plaintext.

The encryption key for a Playfair cipher is a word, i.e., a finite sequence of characters taken from the set of plaintext characters. This keyword will determine the positioning of the characters in the encryption array as is demonstrated in the following example.

Suppose the encryption key is the word *larkspur* and we want to encrypt the message "rocky mountain meadow" using a Playfair cipher with this keyword. First, we construct the 5×5 array below:

L	A	R	K	S
P	U	B	C	D
E	F	G	H	I/J
M	N	O	Q	T
V	W	X	Y	Z

In constructing the above array, we began with the characters of the keyword, but omitted any repetitions of characters. Once we had exhausted the characters of the keyword, we completed the array using the remaining members of the character set, in alphabetical order.

Now that we have the encryption array, we proceed with encryption of the plaintext message as follows. We separate the characters in the plaintext into pairs, to get

ro ck ym ou nt ai nm ea do w⋆

In order for the encryption to proceed, we must do something about the "⋆" that appears in the last pair. It is customary to replace it with a plaintext letter that will be identified easily as extraneous when the message is received. Here, as is customary, we'll use the letter "x."

ro ck ym ou nt ai nm ea do wx

Each pair of characters, (*digram*), in the above plaintext will be replaced with a pair of ciphertext characters according to the following procedure.

- If the two letters in the digram both appear in the same column of the encryption array, each letter of the digram is replaced with the letter immediately below it in the encryption array. (For example, the digram "ro" encrypts to BX.) If there is no letter below, the corresponding ciphertext character is taken from the top of the column (for example, the digram "aw" would have encrypted to UA).
- If the two letters in the digram both appear in the same row of the encryption array, each letter of the digram is replaced with the letter immediately to its right in the encryption array. If there is no letter to the right, the corresponding ciphertext character is taken from the far left of the row (for example, the digram "nt" encrypts to OM).
- If the two plaintext letters in the digram are neither in the same row nor in the same column of the encryption array, each is replaced by the letter in its own row that shares a column with the other letter of the plaintext digram. (For example, the plaintext digram "ym" is replaced with the ciphertext VQ, and the plaintext digram "ou" is replaced with the ciphertext NB.)

L	A	R	K	S
P	U	B	C	D
E	F	G	H	I/J
$\boxed{M}$	N	O	$\boxed{Q}$	T
$\boxed{V}$	W	X	$\boxed{Y}$	Z

L	A	R	K	S
P	$\boxed{U}$	$\boxed{B}$	C	D
E	F	G	H	I/J
M	$\boxed{N}$	$\boxed{O}$	Q	T
V	W	X	Y	Z

Continuing through all of the plaintext digrams in our example, we obtain the ciphertext

BXHCVQNBOMSFONFLBTXY.

There is a complication that arises when a plaintext message contains a doubled character. Because of the way encryption of digrams is done, we cannot allow a digram to contain two copies of the same letter. The solution is to insert a character easily identified as extraneous (usually "x") between any doubled letters, and then separate the modified plaintext into digrams. For example, if our plaintext had been "sunny mountain meadow," separation into digrams would have yielded

su nx ny mo un ta in me ad ow

and playfair encryption with keyword *larkspur* would have produced ADOWQWNQFWNSFTVMSUNX.

For practice, try encrypting the plaintext "hike the foothills" using a Playfair cipher with encryption key *primrose*, and see if you can get the ciphertext GMFCYTFNIZMUGMQVVD.

The following ciphertext was produced using a Playfair cipher with the encryption key *larkspur*:

ILMILDRKRY.

See if you can decrypt it by reversing the encryption algorithm.

Playfair ciphers represent an improvement in security over substitution ciphers, but it is still relatively easy to attack them using a slightly more sophisticated form of frequency analysis. (The frequencies in English of the various digrams are well-known — can you guess what is the most common digram?) An entertaining, accurate, and surprisingly detailed discussion of a playfair attack appears in Dorothy L. Sayers' mystery novel *Have His Carcase*, (Victor Gollancz Ltd, 1932). Cryptanalysis of playfair ciphers is also explained in detail in Chapter 7, section II of the US Army Field Manual, (1990), which can be accessed at

http://www.umich.edu/∼umich/fm-34-40-2/ch7.pdf