Homebrewed Symmetric Block Encryption Algorithm

ITC 3431 L5 CRYPTOGRAPHY & NETWORK SECURITY

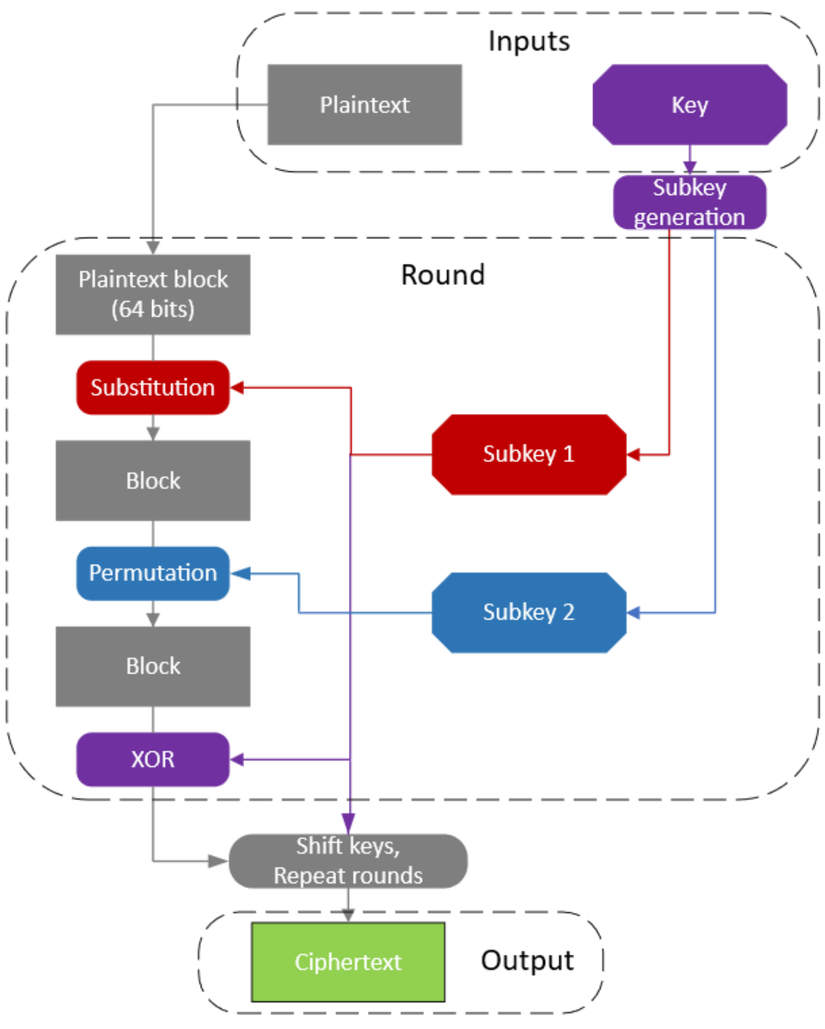
Introduction

The aim of the work was to write an encryption algorithm for text files. It should be symmetric (one key for encryption and decryption), producing unintelligible ciphertext that makes it highly difficult to determine the original message's contents but can be easily and quickly decrypted with knowledge of the original key.

The project derives two subkeys from a user-defined text key and uses them in permutation (moving characters around) and substitution (swapping characters for others) operations and XOR (exclusive-OR) bit manipulation.

The operations are performed in rounds, at the end of each of which the keys' bits are shifted and a new round begins, a total of 16 times (more is safer, but slower). Each round is performed on 64-bit blocks.

Process Overview



Algorithm

Key generation

$$k1, k2 = two\ halves\ of\ the\ key, parsed\ as\ numbers$$
$$ksum = k1 + k2$$
$$subkey1 = (k1 + ksum)\ mod\ k2$$
$$subkey2 = (k2 + ksum)\ mod\ k1$$

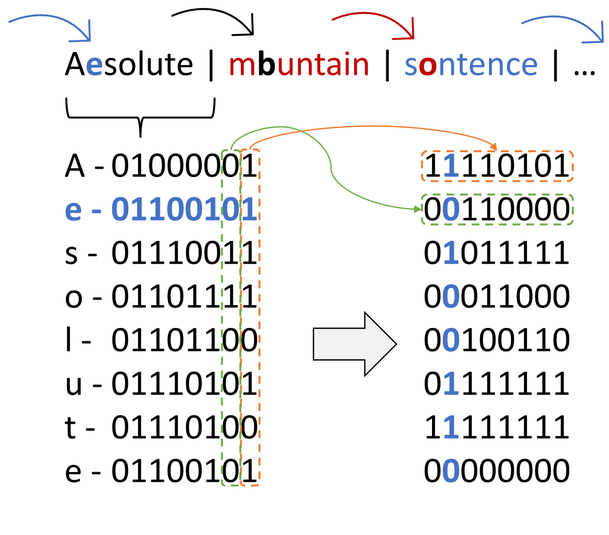
Permutation

Permutation is meant to propagate changes made by the ciphering process to neighboring bytes, diffusing results by "spreading" the tiniest change across the entire resulting text.

8-byte blocks:

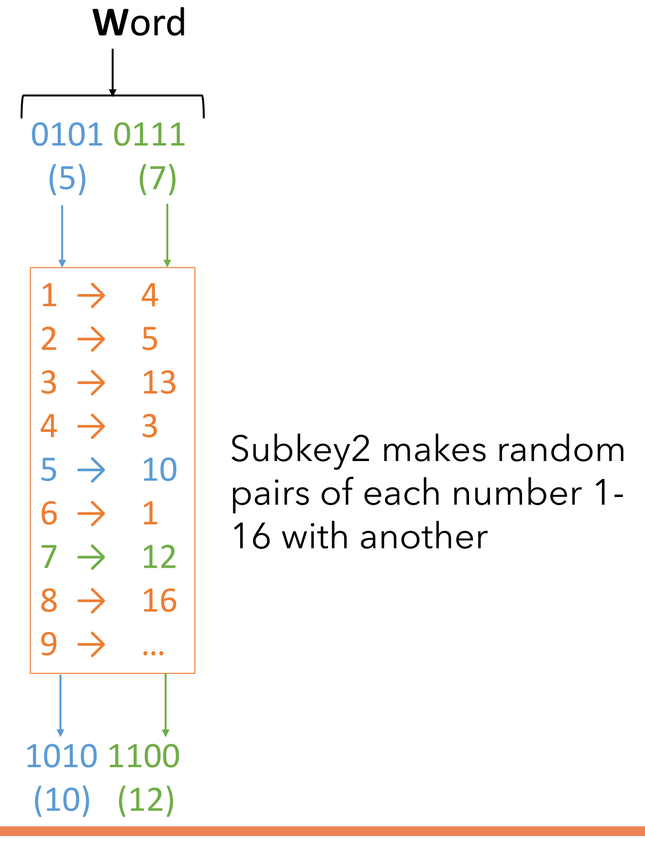
Absolute | mountain | sentence | …

Subkey1 picks a byte 1-8 to shift: **2**



Substitution

Substitution introduces perceived randomness and confusion by arbitrarily (yet predictably, with key) replacing parts of the text with entirely different ones.



Results

The result of the operations is an incoherent collection of base64-encoded characters that are confusing and visually indistinguishable from random text. When given English text with wildly varying occurrences of different characters the resulting ciphertext is almost uniformly diffused and disallows any statistical analysis.

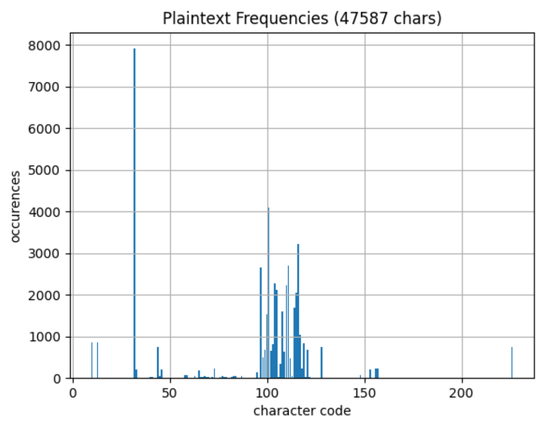
Miniscule changes of the plaintext or key in subsequent runs also lead to entirely different resulting ciphertexts, confirming the strong statistical security.

The algorithm operates from start to finish at an average rate of about 11.000 characters per second.

Plaintext

Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do: once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, "and what is the use of a book," thought Alice "without pictures or conversations?"

So she was considering in her own mind (as well as she could, for the hot day made her feel very sleepy and stupid), whether the pleasure of making a daisy-chain would be worth the trouble of getting up and picking the daisies, when suddenly a White Rabbit with pink eyes ran close by her.



Ciphertext

LeSHQ3ki47dcHuniXzwd25ZyD3AuiDrqB8WtMKUWDj8vuBViAcgx
OkDVyok/3ORb9MjFwqk2eslvsgZCuj1fHbA4A7NxQXdE8yBaPaEm
uSObXk3hhHYPqMK8sxbJYDe1kF/LXrrMmMy5u3lPttyzUWv23SqH
0jAMRmQ60+yqFFM3serl5uNdhsYo17k6avyFOI/48TRIaST4R3WlY
ZfbyndPm/XqvaXVMobXyoI96iVIfb4MsBo4TexFT90pRvIDTNwmRU
Hw0/VauoVZ+5mf0Qib5mQN5vLtPSNSkC+ZnRghCivNmfHAYT79/
RXR32W2hw79HOzwSD9p3+7LlMv+u0TpBzUvaRUdvNc+878jr4rI+
GtzZt41n7m27tjBMDSON5zpdWjkpFLZ+acUxE6UbP8+DkxYzRhfM
GGvfSUDYhHwwgl66u/mL/mHtpT5Ndc8qt+Wl4zHYDciU9jo4yK5O
GrQuJAGLmRawQ+eES2wd9NkAXwbmDkct2w2X1fhyiP/TxBqOaV
HqG9MXcbFMoGK+HnSRYsNUCQQYDNS4/wdwPpa0iJaCkN0kVuf
9ue4WkL8JjK3Pgui8vfIFilJGCHmUrBtmEqU3gvRjpw2c1dykNl9LdU
i4KkCjGi+ZFeOXilEDjo+lBtJKBuhtRnJ0pDGHyX2aet86u9rXPqvxEU
8yC1f+gxHp/CLD9i0CsRC+ymVBVlJSoNHecrDE+QtUQHZuHVkoO
iYs6uU3CEWwDQSysScw4pVWIXMbcIQf0ouOKwC84nCn/p2+kVnt
YguKPD7h1ziTO0SD66IATO6lS77rh8=