Program23 affine modulo 256, Hill modulo 256, S-DES.

#include <stdio.h>

#include <stdlib.h>

typedef unsigned char byte;

void generateRoundKeys(byte key, byte \*k1, byte \*k2) {

\*k1 = 0xF3;

\*k2 = 0xE3;

}

byte sdesEncrypt(byte plaintext, byte key) {

byte k1, k2;

generateRoundKeys(key, &k1, &k2);

return plaintext ^ k1;

}

void ctrEncrypt(byte \*plaintext, byte key, int length) {

byte counter = 0x00;

int i;

for ( i = 0; i < length; i++) {

byte encrypted = sdesEncrypt(counter, key);

plaintext[i] ^= encrypted;

counter++;

}

}

int main() {

byte key = 0xFD;

byte plaintext[] = {0x01, 0x02, 0x04};

int length = sizeof(plaintext);

int i;

printf("Plaintext: ");

for ( i = 0; i < length; i++) {

printf("%02X ", plaintext[i]);

}

ctrEncrypt(plaintext, key, length);

printf("\nEncrypted: ");

for (i = 0; i < length; i++) {

printf("%02X ", plaintext[i]);

}

ctrEncrypt(plaintext, key, length);

printf("\nDecrypted: ");

for (i = 0; i < length; i++) {

printf("%02X ", plaintext[i]);

}

printf("\n");

return 0;

}

OUTPUT:

Plaintext: 01 02 04

Encrypted: F2 F0 F5

Decrypted: 01 02 04