Информационная безопасность. Лабораторная работа № 8 на тему “Элементы криптографии. Шифрование (кодирование) различных исходных текстов одним ключом”

Горбунова Ярослава Михайловна

RUDN University, Moscow, Russian Federation

Содержание

Содержание

Цели и задачи

Цели и задачи

Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом

Выполнение

Выполнение

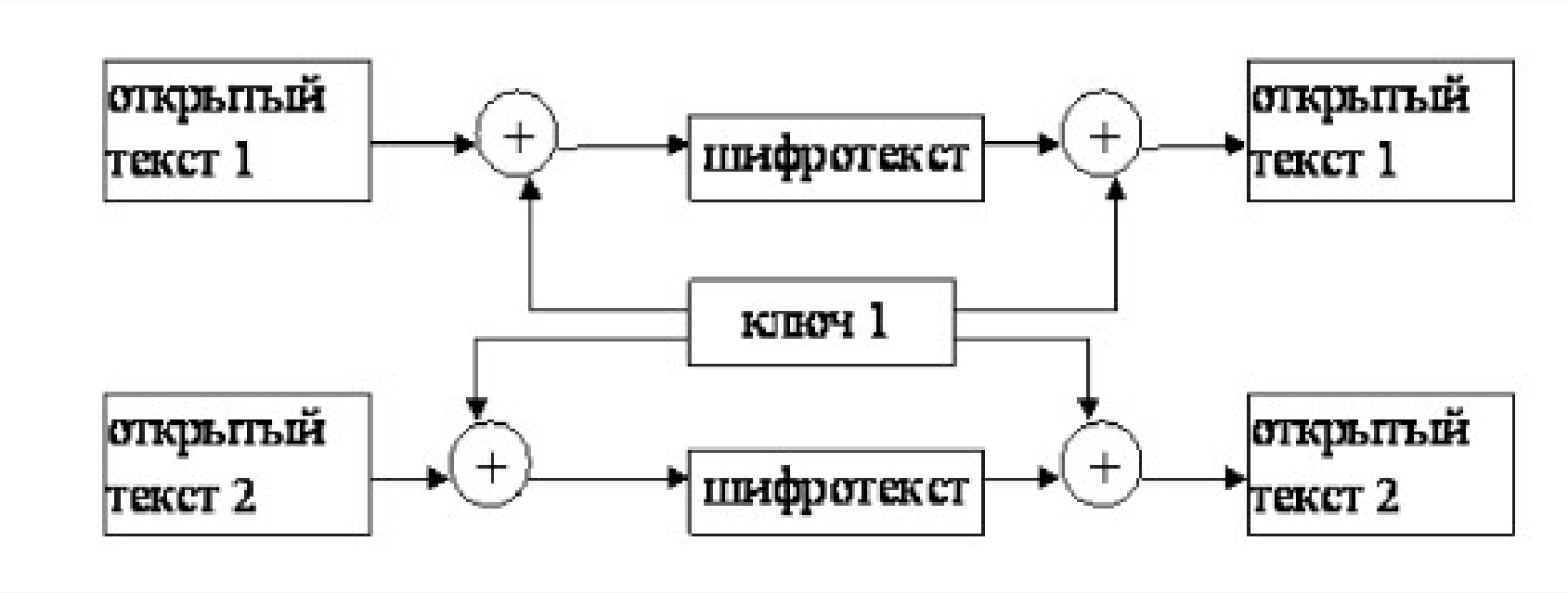

Figure 1: Общая схема шифрования двух различных текстов одним ключом

Выполнение

$$C1 = P1 \oplus K$$

$$C2 = P2 \oplus K \quad (8.1)$$

$$C1 \oplus C2 = P1 \oplus K \oplus P2 \oplus K = P1 \oplus P2$$

$$C1 \oplus C2 \oplus P1 = P1 \oplus P2 \oplus P1 = P2 \quad (8.3)$$

Выполнение

```
/*
Gorbunova Y.M., NFI-01-19, 2022

Single gamming
Principle:
1) text ^ gamma = ciphertext;
2) ciphertext ^ gamma = text;
3) gamma = ciphertext ^ text;
where "^" is additions modulo 2 (XOR)

ciphertext_1 = text_1 ^ gamma
ciphertext_2 = text_2 ^ gamma

4) ciphertext_1 ^ ciphertext_2 ^ text_1 = text_2
   ciphertext_1 ^ ciphertext_2 ^ text_2 = text_1
*/


#include <iostream>
#include <string>
#include <cstring>
#include <bitset>

using namespace std;

const int m = 1024;  // array dimension
const int n = 8;  // bitset dimension

// Print for unmodified text
void print_bitset_text(char arr[]) {
    for (unsigned int i = 0; i < strlen(arr); i++)
        cout << bitset<n>((unsigned char)arr[i]) << " ";
}

```

Figure 2: Программа (1)

Выполнение

```
35    // Print for modified text
36    void print_bitset_gamma(char arr[], char text[]) {
37        for (unsigned int i = 0; i < strlen(text); i++)
38            cout << bitset<n>((unsigned char)arr[i]) << " ";
39    }
40
41    int main()
42    {
43        // Introduce variables
44        //char text_1[m] = "НаВашисходящийот1204";
45        char text_1[m] = "NaVasishodysiyot1204";
46        //char text_2[m] = "ВСеверныйфилиалБанка";
47        char text_2[m] = "VSevernyifilialBanka";
48        int gam[m] = {
49            0x05, 0x0C, 0x17, 0x7F, 0x0E, 0x4E, 0x37, 0xD2, 0x94, 0x10,
50            0x09, 0x2E, 0x22, 0x57, 0xFF, 0xC8, 0x0B, 0xB2, 0x70, 0x54
51        };  // given key (int)
52        char gamma[m];  // key (char)
53        char gam_text_1[1024], gam_text_2[1024];  // ciphertexts
54
55        // Convert from integer to char
56        for (unsigned int i = 0; i < size(gam); i++) {
57            gamma[i] = (char)gam[i];
58        }
59
60        cout << "----------------------------------------------------------------" << endl << "Part 1" << endl;
61
62        cout << "    Bitset of the key (gamma) :\n";
63        print_bitset_text(gamma);
64        cout << endl;
65
66        // Text 1
67        cout << "    Text 1 for single gamming: \n" << text_1 << endl;
68
```

Figure 3: Программа (2)

Выполнение

```
69      cout << "    Bitset of the Text 1 :\n";
70      print_bitset_text(text_1);
71      cout << endl;
72
73      cout << "    Ciphertext computation...\n";
74
75      // Convert text to ciphertext
76      for (unsigned int i = 0; i < strlen(text_1); i++) {
77          gam_text_1[i] = text_1[i] ^ gamma[i];
78      }
79      cout << "    Bitset of the Ciphertext (gam_text_1) :\n";
80      print_bitset_gamma(gam_text_1, text_1);
81      cout << endl;
82
83      // Text 2
84      cout << "    Text 2 for single gamming: \n" << text_2 << endl;
85
86      cout << "    Bitset of the Text 2 :\n";
87      print_bitset_text(text_2);
88      cout << endl;
89
90      cout << "    Ciphertext computation...\n";
91
92      // Convert text to ciphertext
93      for (unsigned int i = 0; i < strlen(text_2); i++) {
94          gam_text_2[i] = text_2[i] ^ gamma[i];
95      }
96      cout << "    Bitset of the Ciphertext (gam_text_2) :\n";
97      print_bitset_gamma(gam_text_2, text_2);
98
99      cout << endl;
100     cout << "-----------------------------------------------------------------" << endl;
101     cout << endl << "-----------------------------------------------------------------" << endl << "Part 2" << endl;
102
```

Figure 4: Программа (3)

Выполнение

```
103     cout << "    1) Accept Text 2 is unknown. Define Text 2 via cybertext 1, cybertest 2 and Test 1: \n";
104     cout << "    Text 2 computation...\n";
105     // Define text_2
106     cout << "    Check of correct computation work...\n    Obtained Text 2 after single gamming" << endl;
107     for (unsigned int i = 0; i < strlen(text_1); i++) {
108         cout << static_cast<char>(bitset<n>((unsigned char)(gam_text_1[i] ^ gam_text_2[i] ^ text_1[i])).to_ulong() + 256);
109     }
110     cout << endl;
111
112     cout << "    2) Accept Text 1 is unknown. Define Text 1 via cybertext 1, cybertest 2 and Test 2: \n";
113     cout << "    Text 1 computation...\n";
114     // Define text_2
115     cout << "    Check of correct computation work...\n    Obtained Text 1 after single gamming" << endl;
116     for (unsigned int i = 0; i < strlen(text_2); i++) {
117         cout << static_cast<char>(bitset<n>((unsigned char)(gam_text_1[i] ^ gam_text_2[i] ^ text_2[i])).to_ulong() + 256);
118     }
119     cout << endl << "----------------------------------------------------------------" << endl;
120 }
121
```

Figure 5: Программа (4)

Выполнение

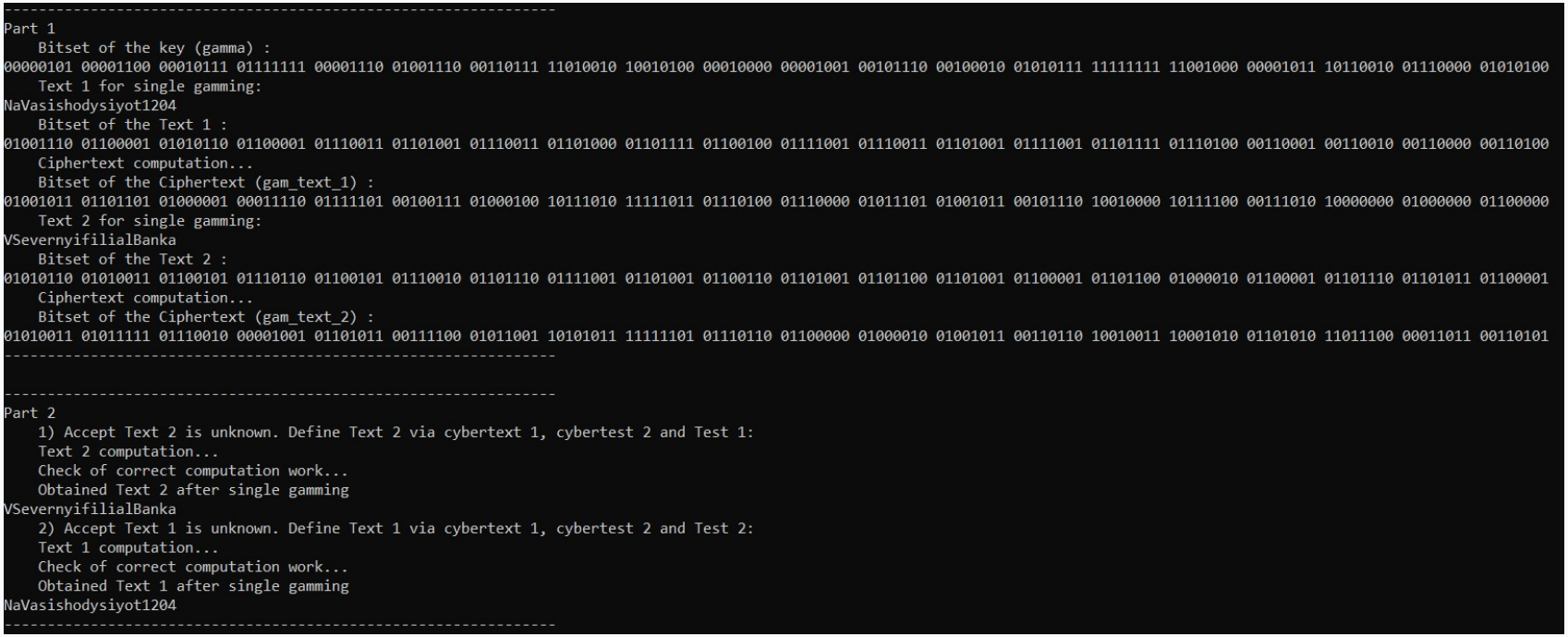

Figure 6: Вывод работы программы

Результаты

Результаты

Освоено на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом

Список литературы

Список литературы

1. Методические материалы курса