Презентация по лабораторной работе №7

НКНбд-01-21

Юсупов Эмиль Артурович

Введение

Цель работы

- Освоить на практике применение режима однократного гаммирования.

Выполнение работы

Выполнение работы

1. Проанализировали паттерны работы самого шифрования/дешифрования.
2. В программе мы занесли ASCII таблицу в вектор.
3. Сделали генератор случайного ключа.
4. Написали функции шифрования/дешифрования.
5. Прописали главную функцию со всей логикой.
6. Получили в консоль информацию.

Листинг программы

ASCII to Vector

```
void pushToVec(vector<char>* v) {
    for (int i = 0; i < 128; i++) {
        v->push_back(char(i));
    }
}
```

Figure 1: ASCII to Vector

Random key generator

```cpp
vector<char> generateRandomKey(const vector<char> *v, size_t len) {
    random_device rd;
    mt19937 mt(rd());
    uniform_int_distribution<> dist(0, v->size()-1);

    vector<char> key;

    for (int i = 0; i < len; i++) {
        key.push_back((*v)[dist(mt)]);
    }

    return key;
}
```

Figure 2: Random Key Generator

Encryption/Decryption methods

```cpp
vector<char> xorEncryption(vector<char> p, vector<char> k) {
    vector<char> enc;
    if (p.size() == k.size()) {
        for (int i = 0; i < p.size(); i++) {
            enc.push_back(p[i] ^ k[i]);
        }
    }
    return enc;
}


vector<char> findKey(const vector<char>& p, const vector<char>& enc) {
    return xorEncryption(p, enc);
}

vector<char> xorDecryption(const vector<char>& enc, const vector<char>& k) {
    return xorEncryption(enc, k);
}
```

Figure 3: Encryption/Decryption methods

Main

```cpp
int main()
{
    vector<char> v;
    // 97 - a, z - 122
    pushToVec(&v);

    std::string str;
    cout << "Enter the input: ";
    getline(cin, str);

    vector<char> arr(str.begin(), str.end());
    vector<char> key = generateRandomKey(&v, str.size());
    vector<char> enc = xorEncryption( arr, key);

    cout << "Original: " << str << endl;
    cout << "Generated: ";
    for (char c : key) { cout << c;}
    cout << endl << "Encrypted: ";
    for (char c : enc) { cout << c; }

    vector<char> decrypted = xorDecryption(enc, key);
    cout << "Decrypted: ";
    for (char c : decrypted) { cout << c; }

    vector<char> fKey = findKey(arr, enc);
    vector<char> decMes = xorDecryption(enc, fKey);
    cout << "\nDecrypted with key: ";
    for (char c : decMes) { cout << c; }
    cout << endl;

    return 0;
}
```

Output

```
Enter the input: Hello, World!
Original: Hello, World!
Generated: [/"lVaNnT4f
 Encrypted: JN9Mn9zt8PGDecrypted: Hello, World!
Decrypted with key: Hello, World!
```

Figure 5: Console output

Выводы

Выводы

Во время выполнения работы, мы получили навыки работы с режимом однократного гаммирования.

Спасибо за вниманиие