**Задание**

Написать программу для шифрования/дешифрования файлов на языке С#

**Текст программы**

using System;

using System.IO;

using System.Security.Cryptography;

class FileEncryptor

{

static void Main()

{

string inputFile = "input.txt";

string encryptedFile = "encrypted.bin";

string decryptedFile = "decrypted.txt";

// Генерация случайного ключа и вектора инициализации

byte[] key = GenerateRandomBytes(32); // 256 бит (32 байта)

byte[] iv = GenerateRandomBytes(16); // 128 бит (16 байт)

// Шифрование файла

EncryptFile(inputFile, encryptedFile, key, iv);

Console.WriteLine("File encrypted successfully.");

// Дешифрование файла

DecryptFile(encryptedFile, decryptedFile, key, iv);

Console.WriteLine("File decrypted successfully.");

}

// Шифрование файла

static void EncryptFile(string inputFile, string outputFile, byte[] key, byte[] iv)

{

using (Aes aesAlg = Aes.Create())

{

// Устанавливаем ключ и вектор инициализации для алгоритма AES

aesAlg.Key = key;

aesAlg.IV = iv;

// Открываем поток для чтения из исходного файла

using (FileStream inputFileStream = new FileStream(inputFile, FileMode.Open))

{

// Открываем поток для записи в зашифрованный файл

using (FileStream outputFileStream = new FileStream(outputFile, FileMode.Create))

{

// Создаем объект, осуществляющий шифрование

using (ICryptoTransform encryptor = aesAlg.CreateEncryptor())

{

// Создаем поток для шифрования

using (CryptoStream cryptoStream = new CryptoStream(outputFileStream, encryptor, CryptoStreamMode.Write))

{

// Копируем данные из исходного файла в поток шифрования

inputFileStream.CopyTo(cryptoStream);

}

}

}

}

}

}

// Дешифрование файла

static void DecryptFile(string inputFile, string outputFile, byte[] key, byte[] iv)

{

using (Aes aesAlg = Aes.Create())

{

// Устанавливаем ключ и вектор инициализации для алгоритма AES

aesAlg.Key = key;

aesAlg.IV = iv;

// Открываем поток для чтения из зашифрованного файла

using (FileStream inputFileStream = new FileStream(inputFile, FileMode.Open))

{

// Открываем поток для записи в дешифрованный файл

using (FileStream outputFileStream = new FileStream(outputFile, FileMode.Create))

{

// Создаем объект, осуществляющий дешифрование

using (ICryptoTransform decryptor = aesAlg.CreateDecryptor())

{

// Создаем поток для дешифрования

using (CryptoStream cryptoStream = new CryptoStream(outputFileStream, decryptor, CryptoStreamMode.Write))

{

// Копируем данные из зашифрованного файла в поток дешифрования

inputFileStream.CopyTo(cryptoStream);

}

}

}

}

}

}

// Генерация случайных байтов для ключей и векторов инициализации

static byte[] GenerateRandomBytes(int length)

{

using (RNGCryptoServiceProvider rngCrypto = new RNGCryptoServiceProvider())

{

byte[] randomBytes = new byte[length];

rngCrypto.GetBytes(randomBytes);

return randomBytes;

}

}

}

**Результат работы**

