**Міністерство Освіти І НАУКИ України**

**Національний університет "Львівська політехніка"**

Інститут **КНІТ**

Кафедра **ПЗ**

### ЗВІТ

До лабораторної роботи № 4

**З дисципліни:** *“Безпека програм та даних”*

**На тему:** *“Створення програмної реалізації алгоритму шифрування з відкритим ключем RSA з використанням Microsoft CryptoAPI”*

**Лектор:**

доц. каф. ПЗ

Сенів М. М.

**Виконав:**

ст. гр. ПЗ-44

Вонс Ю. А.

**Прийняв:**

асист. каф. ПЗ

Самбір А. А.

« \_\_\_\_ » \_\_\_\_\_\_\_\_ 2019 р.

∑= \_\_\_\_\_ .

Львів – 2019

**Тема роботи:** Створення програмної реалізації алгоритму шифрування з відкритим ключем RSA з використанням Microsoft CryptoAPI.

**Мета роботи:** Ознайомитись з методами і засобами криптографії з відкритим ключем, навчитись створювати програмні засоби з використанням криптографічних інтерфейсів.

**TЕОРЕТИЧНІ ВІДОМОСТІ**

RSA – це алгоритм з відкритим ключем, призначений як для шифрування, так і для аутентифікації. Алгоритм заснований на використанні того факту, що задача факторизації є складною, тобто легко перемножити два числа, у той час як не існує поліноміального алгоритму знаходження простих множників великого числа.

Основна ідея шифрування з відкритим ключем – використовувати ключі парами, що складаються з ключа шифрування та ключа дешифрування, які неможливо обчислити один з одного.

1. Кожен користувач генерує пару ключів: один для шифрування і один для дешифрування.

2. Кожен користувач публікує свій ключ шифрування, розміщує його у відкритому для всіх доступі. Другий ключ, парний до відкритого, зберігається в таємниці.

3. Якщо користувач А збирається надіслати повідомлення користувачеві В, він шифрує повідомлення відкритим ключем користувача В.

4. Коли користувач В отримує повідомлення, він дешифрує його за допомогою свого приватного ключа. Інший користувач не зможе дешифрувати повідомлення, оскільки приватний ключ В відомий тільки користувачеві В.

**ЗАВДАННЯ**

З використання функцій CryptoAPI створити програмну реалізацію алгоритму шифрування RSA.

**ТЕКСТ ПРОГРАМИ**

using System;

using System.Windows;

using Microsoft.Win32;

using System.IO;

namespace RSA

{

public partial class MainWindow : Window

{

private static Algorithm\_RSA obj;

public MainWindow()

{

InitializeComponent();

obj = new Algorithm\_RSA();

obj.Init();

}

private void button1\_Click(object sender, RoutedEventArgs e)

{

OpenFileDialog openFileDialog = new OpenFileDialog();

openFileDialog.ShowDialog();

TextBoxOriginalFilePath.Text = openFileDialog.FileName;

}

private void button2\_Click(object sender, RoutedEventArgs e)

{

DateTime time1 = DateTime.Now;

obj.EncryptFIle(TextBoxOriginalFilePath.Text, @"C:\Users\hp\Desktop\Безпека програм та даних\RSA\enc\_try.txt");

DateTime time2 = DateTime.Now;

TimeSpan TimeToCrypted = time2.Subtract(time1);

string answer = string.Format("{0:D3} мілісекунд", TimeToCrypted.Milliseconds);

MessageBox.Show("Файл зашифровано за " + answer);

using (var sr = new StreamReader(@"C:\Users\hp\Desktop\Безпека програм та даних\RSA\enc\_try.txt"))

{

var str = sr.ReadToEnd();

textBox.Text = Convert.ToString(str);

}

}

private void button3\_Click(object sender, RoutedEventArgs e)

{

DateTime time1 = DateTime.Now;

obj.DecryptFile(@"C:\Users\hp\Desktop\Безпека програм та даних\RSA\enc\_try.txt", @"C:\Users\hp\Desktop\Безпека програм та даних\RSA\decr\_try.txt");

DateTime time2 = DateTime.Now;

TimeSpan TimeToDecrypted = time2.Subtract(time1);

string answer = string.Format("{0:D3} мілісекунд", TimeToDecrypted.Milliseconds);

MessageBox.Show("Файл розшифровано за " + answer);

using (var reader = new StreamReader(@"C:\Users\hp\Desktop\Безпека програм та даних\RSA\decr\_try.txt"))

{

var str = reader.ReadLine();

textBox.Text = str.ToString();

}

}

}

}

using System;

using System.IO;

using System.Security.Cryptography;

using System.Text;

namespace RSA

{

public class Algorithm\_RSA

{

private RSACryptoServiceProvider RSAService;

public string pub\_key { get; set; }

public string pr\_key { get; set; }

public void Init()

{

RSAService = new RSACryptoServiceProvider(1024);

byte[] key = RSAService.ExportCspBlob(false);

File.WriteAllBytes(@"C:\Users\hp\Desktop\Безпека програм та даних\RSA\private\_key.xml", key);

key = RSAService.ExportCspBlob(true);

File.WriteAllBytes(@"C:\Users\hp\Desktop\Безпека програм та даних\RSA\public\_key.xml", key);

var writer = new StreamWriter(@"C:\Users\hp\Desktop\Безпека програм та даних\RSA\private\_key.xml");

pr\_key = RSAService.ToXmlString(true);

writer.Write(pr\_key);

writer.Close();

writer = new StreamWriter(@"C:\Users\hp\Desktop\Безпека програм та даних\RSA\public\_key.xml");

pub\_key = RSAService.ToXmlString(false);

writer.Write(pub\_key);

writer.Close();

}

public string Encrypt(string data)

{

RSAService.FromXmlString(pub\_key);

var dataBytes = Encoding.UTF8.GetBytes(data);

var encryptedData = RSAService.Encrypt(dataBytes, false);

return Convert.ToBase64String(encryptedData);

}

public byte[] Encrypt(byte[] data)

{

return RSAService.Encrypt(data, false);

}

public string Decrypt(string data)

{

var dataBytes = Convert.FromBase64String(data);

RSAService.FromXmlString(pr\_key);

var result = RSAService.Decrypt(dataBytes, false);

return Encoding.UTF8.GetString(result);

}

public void EncryptFIle(string file, string outFile)

{

RSAService.FromXmlString(pub\_key);

var f = new FileStream(file, FileMode.Open);

var buffer = new byte[78];

var fw = new FileStream(outFile, FileMode.Create);

while (f.Read(buffer, 0, buffer.Length) != 0)

{

var a = Encrypt(buffer);

fw.Write(a, 0, a.Length);

}

f.Close();

fw.Close();

}

public void DecryptFile(string inputFile, string outputFile)

{

RSAService.FromXmlString(pr\_key);

var f = new FileStream(inputFile, FileMode.Open);

var buffer = new byte[128];

var fw = new FileStream(outputFile, FileMode.Create);

while (f.Read(buffer, 0, buffer.Length) != 0)

{

var a = RSAService.Decrypt(buffer, false);

fw.Write(a, 0, a.Length);

}

f.Close();

fw.Close();

}

}

}

**РЕЗУЛЬТАТИ**

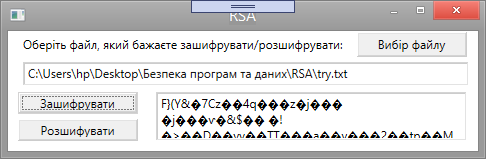


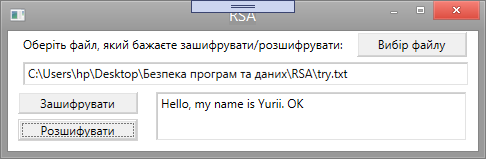
Рис. 1. Результат шифрування файлу

Рис. 2. Результат розшифрування файлу

**ВИСНОВКИ**

Виконуючи дану лабораторну роботу, я ознайомився на практиці з методами криптографічного забезпечення конфіденційності за допомогою асиметричного алгоритму шифрування RSA та створив програмну реалізацію алгоритму шифрування з відкритим ключем RSA з використанням Microsoft CryptoAPI.