Основи інформаційної безпеки

ПЗ №7

“ Асиметричне шифрування як засіб забезпечення конфіденційності інформації”

Підготував:

Студент групи МІТ-21

Пономаренко Владислав

(З-1)

(КОД)

using System;

using System.Security.Cryptography;

using System.IO;

using System.Text;

namespace Lab7x1

{

public class RSAWithRSAParameterKey

{

private RSAParameters \_publicKey;

private RSAParameters \_privateKey;

public void AssignNewKey()

{

using (var rsa = new RSACryptoServiceProvider(2048))

{

rsa.PersistKeyInCsp = false;

\_publicKey = rsa.ExportParameters(false);

\_privateKey = rsa.ExportParameters(true);

}

}

public byte[] EncryptData(byte[] dataToEncrypt)

{

byte[] cipherbytes;

using (var rsa = new RSACryptoServiceProvider())

{

rsa.PersistKeyInCsp = false;

rsa.ImportParameters(\_publicKey);

cipherbytes = rsa.Encrypt(dataToEncrypt, true);

}

return cipherbytes;

}

public byte[] DecryptData(byte[] dataToEncrypt)

{

byte[] plain;

using (var rsa = new RSACryptoServiceProvider())

{

rsa.PersistKeyInCsp = false;

rsa.ImportParameters(\_privateKey);

plain = rsa.Decrypt(dataToEncrypt, true);

}

return plain;

}

}

class Program

{

static void Main(string[] args)

{

var rsaParams = new RSAWithRSAParameterKey();

const string original = "Vlad Hi";

rsaParams.AssignNewKey();

Console.WriteLine(" Original Text = " + original);

Console.WriteLine();

var encrypted = rsaParams.EncryptData(Encoding.UTF8.GetBytes(original));

var decrypted = rsaParams.DecryptData(encrypted);

Console.WriteLine(" Encrypted Text = " + Convert.ToBase64String(encrypted));

Console.WriteLine();

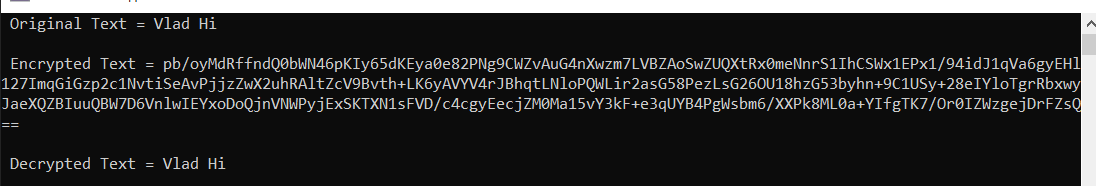
Console.WriteLine(" Decrypted Text = " + Encoding.Default.GetString(decrypted));

}

}

}

(РУЗУЛЬТАТ)



(З-2)

(КОД)

using System;

using System.IO;

using System.Security.Cryptography;

using System.Text;

namespace lab7\_2

{

class Program

{

public void AssignNewKey(string publicKey = "Public.xml",

string privateKey = "Private.xml")

{

using (var rsa = new RSACryptoServiceProvider(2048))

{

rsa.PersistKeyInCsp = false;

File.WriteAllText(publicKey, rsa.ToXmlString(false));

File.WriteAllText(privateKey, rsa.ToXmlString(true));

}

}

public byte[] EncryptData(byte[] dataToEncrypt, string publicKey = "Public.xml")

{

byte[] cb;

using (var rsa = new RSACryptoServiceProvider(2048))

{

rsa.PersistKeyInCsp = false;

rsa.FromXmlString(File.ReadAllText(publicKey));

cb = rsa.Encrypt(dataToEncrypt, false);

}

return cb;

}

public byte[] DecryptData(byte[] dataToDecrypt, string privateKey = "Private.xml")

{

byte[] p;

using (var rsa = new RSACryptoServiceProvider(2048))

{

rsa.PersistKeyInCsp = false;

rsa.FromXmlString(File.ReadAllText(privateKey));

p = rsa.Decrypt(dataToDecrypt, false);

}

return p;

}

static void Main(string[] args)

{

var rsap = new Program();

const string original = "Hey";

rsap.AssignNewKey();

var enc = rsap.EncryptData(Encoding.UTF8.GetBytes(original));

var dec = rsap.DecryptData(enc);

Console.WriteLine(" Original Text = " + original);

Console.WriteLine(" Encrypted Text = " + Convert.ToBase64String(enc));

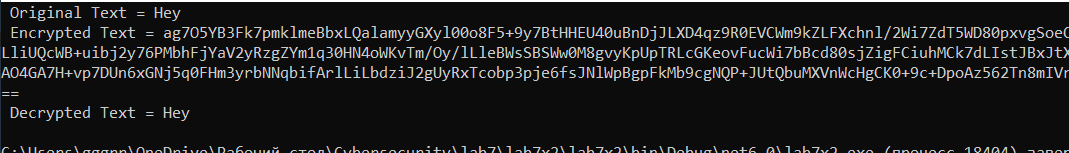
Console.WriteLine(" Decrypted Text = " + Encoding.Default.GetString(dec));

}

}

}

(РЕЗУЛЬТАТ)



(ПОСИЛАННЯ)

<https://github.com/VladPonomarenkoi/Cybersecurity>

(ВИСНОВОК)

у ході лабораторної роботи я навчився, зашифровувати та розшифровувати паролі за допомогою асиметричного шифрування. Навчився зберігати приватні та публічні ключі різними способами. Навчився зашифровувати повідомлення за допомогою публічних ключів, які зберігаються у файлі комп’ютера.