# Security 2017 Opdracht 2: Cryptografie

Inleveren: UITSLUITEND via Blackboard. UITSLUITEND .dccx of .pdf formaat.

**Uiterste datum: 14-5-2016, 23:59uur.**

**Vraag 1. Symmetrische en asymmetrische encryptie**

* 1. **Wat is het verschil tussen symmetrische en asymmetrische encryptie?**

Bij symmetrisch is de sleutel voor encryptie ook de sleutel voor decryptie.

Asymmetrisch heeft een publieke sleutel en aan iedere kant een unieke sleutel.

**Auke wil een bericht naar Bauke sturen. Daarbij wil ze dat integrity is gegarandeerd.**

* 1. **Wat betekent integrity ?’**

Integrity is dat een programma goed en volledig werkt.

* 1. **Wat moet Auke doen om te zorgen dat Eve (een hacker) het bericht niet ongemerkt kan vervangen door een vals bericht?**

Ervoor zorgen dat de sleutel die gebruikt worden voor het encrypten/decrypten goed bewaakt wordt. Als de hacker dan een bericht stuurt en de desbetreffende hacker weet niet hoe het geencrypt of gedecrypt is, dan kan Auke wanneer zij het bericht decrypt zien dat het binnengekomen bericht niet klopt.

* 1. **Hoe kan Bauke de integrity van het verzonden bericht controleren?**

Door zijn unieke sleutel te gebruiken bij het decrypten. Als het bericht van Auke is dan kan hij het lezen, als het van Eve is dan niet (tenzij zijn unieke sleutel bekend is geworden bij haar).

**Auke had geen zin om te zorgen dat Evert (een andere hacker) het bericht niet kan onderscheppen en lezen.**

* 1. **Wat wordt bedoeld met confidentiality ?**

Confidentiality betekend dat wanneer iemand iets heeft wat alleen bedoelt is voor een specefieke groep mensen en niet verkrijgbaar is voor iemand hierbuiten.

* 1. **Wat had Auke beter kunnen doen (zodat Evert het bericht niet kan lezen) ?**

Zijn bericht encrypten voordat hij het upstuurde.

**Vraag 2. Certificaten en PKI**

1. **Wat voor informatie staat in een certificaat?**

Een bewijs van wie iemand is en of hij/zij bevoegd is voor iets.

1. **Wat is de functie van een certificaat?**

Ervoor zorgen dat alleen mensen die bevoegd zijn ervoor het kunnen gebruiken.

1. **Wat is een Registration Authority (RA)?**

Een autoriteit in een netwerk dat de gebruikersverzoeken voor een digitaal certificaat verifieert en de certificaat autoriteit recht geeft om het uit te geven.

1. **Gaat het hier om symmetrische of asymmetrische encryptie?**

Asymmetrisch.

1. **Je kan zelf ook een certificaat aanmaken. Waarom worden dan toch certificaten van een erkende Certification Authority (CA) gebruikt?**

Omdat dit veiliger is om te doen, want het hacken van een standaard gebruiker is makkelijker dan het hacken van de CA.

**Vraag 3. Programmeeropdracht**

Het Microsoft .NET framework bevat ondermeer een groot aantal cryptografische functies.   
  
Maak een eenvoudig C# console-programma met de volgende functionaliteit:

1. Invoeren van een sleutel (in de vorm van een string van hexadecimale getallen)
2. Invoeren van een te encrypten string
3. Encryptie van de string volgens TripleDES.
4. Uitvoer van de ge-encryptede string
5. De-cryptie van de ge-encryptede string

Test daarna dit programma in paren: Wissel de ge-encryptede strings uit, evenals de sleutels. Gebruik dit om de strings van de ander te decrypten (NB: NIET met het zelfde programma, maar met het programma dat de ander heeft gemaakt. Laat zien dat deze kruislingse encryptie goed werkt.  
**Wat moet je bij vraag 3 inleveren:** Alleen de c#-code van je programma en de console-uitvoer van encryptie en decryptie. Voeg beide toe aan het .docx of het .pdf bestand dat je inlevert.

# Code encrypt & decrypt

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Security.Cryptography;

namespace security\_test

{

class Program

{

static UTF8Encoding utf8 = new UTF8Encoding();

static TripleDES CreateDES(string key)

{

MD5 md5 = new MD5CryptoServiceProvider();

TripleDES des = new TripleDESCryptoServiceProvider();

des.Key = md5.ComputeHash(utf8.GetBytes(key));

des.Mode = CipherMode.ECB;

des.Padding = PaddingMode.PKCS7;

return des;

}

static void Main()

{

Console.WriteLine("ENTER KEY");

string key = Console.ReadLine();

TripleDES des = CreateDES(key);

//encrypt

Console.WriteLine("ENCRYPT (e) OR DECRYPT (d)?");

string answer = Console.ReadLine();

if (answer == "e")

{

Console.WriteLine("Type in your string");

ICryptoTransform ct = des.CreateEncryptor();

byte[] input = utf8.GetBytes(Console.ReadLine());

byte[] encrypted = ct.TransformFinalBlock(input, 0, input.Length);

Console.WriteLine(Convert.ToBase64String(encrypted));

}

//decrypt

else if (answer == "d")

{

Console.WriteLine("Type in your string");

string CypherText = Console.ReadLine();

byte[] b = Convert.FromBase64String(CypherText);

ICryptoTransform ct = des.CreateDecryptor();

byte[] input = ct.TransformFinalBlock(b, 0, b.Length);

string decrypted = utf8.GetString(input);

Console.WriteLine(decrypted);

}

else { Console.WriteLine("GEEN OPTIE."); }

Console.ReadLine();

}

}

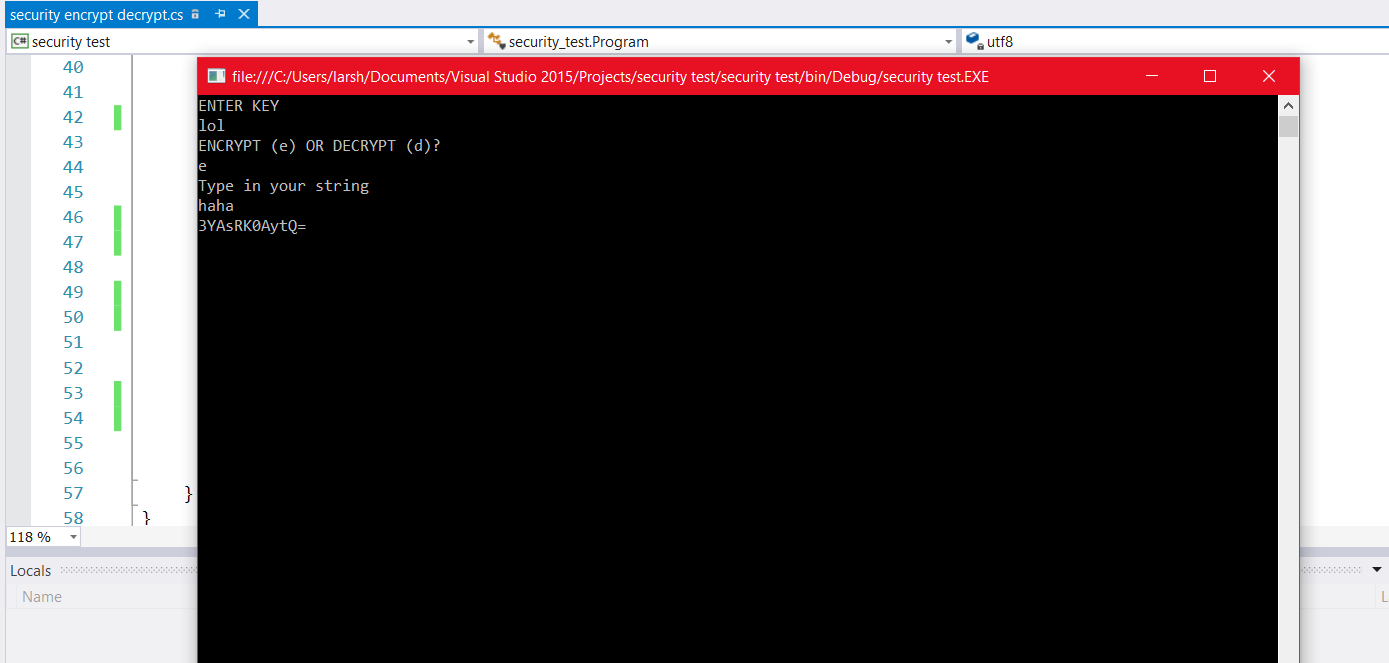
}

# Screenshots

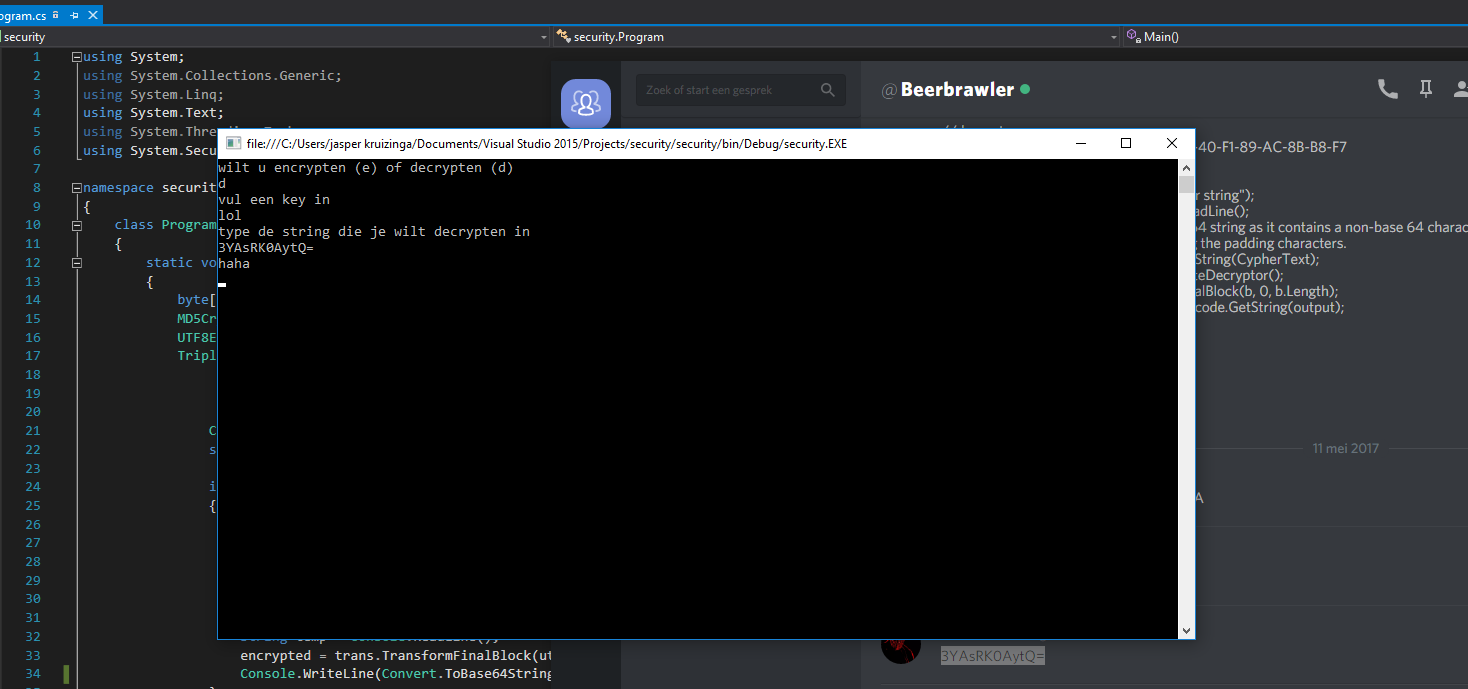
Lars Key = lol

Lars Encrypted string = D/JIyKbs/6VyQPGJrIu49w==

Encrypt (Lars):



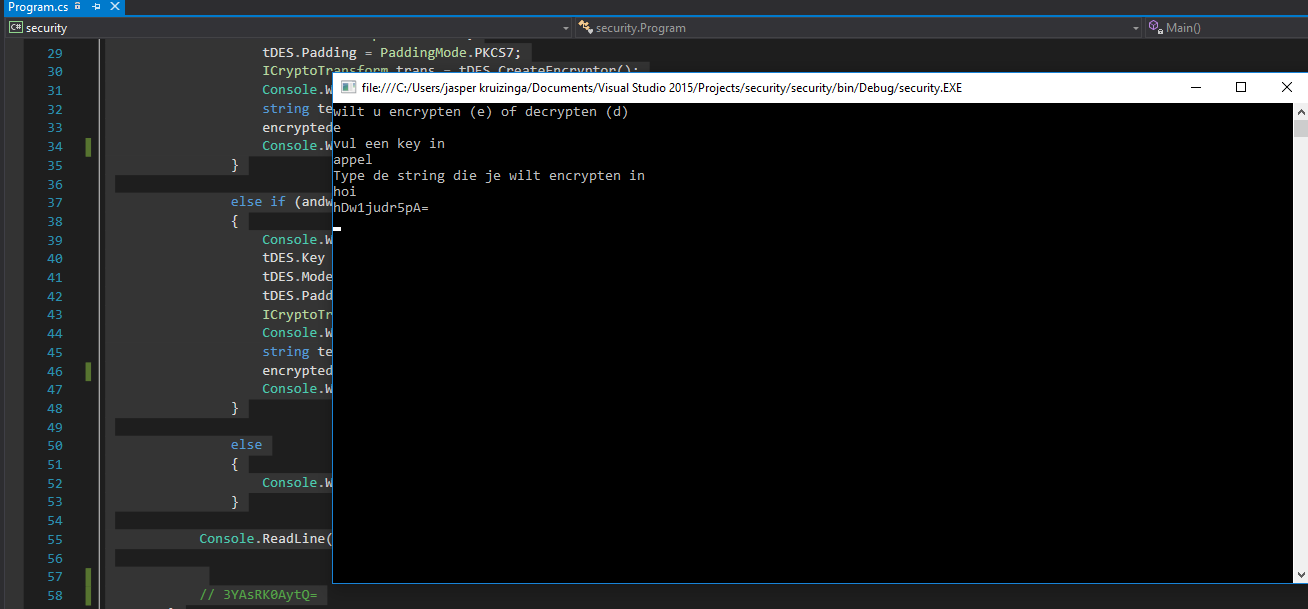
Decrypt (Jasper):



Jasper Key = appel

Jasper Encrpyted string = hDw1judr5pA=

Encrypt (Jasper):



Decrypt (Lars):

