

6. Lớp lưu trữ mã hóa CryptoStream

An ninh trong truyền tải dữ liệu luôn cần được cân nhắc.

.NET thực sự cung cấp lớp CryptoStream qua lớp cha Stream. Tuy nhiên chúng có thể dùng nó như lớp con của không gian System.IO.

CryptoStream cần ba tham số. Thứ nhất là lưu trữ dữ liệu vào mã hóa. Thứ hai, thuật toán mã hóa. Tham số sau cùng chỉ ra cách truy xuất hoặc ghi dữ liệu.

Có nhiều thuật toán mã hóa khác nhau được cung cấp qua giao tiếp ICryptoTransform. Ví dụ sau đây liệt kê các thuật toán mã hóa từ không gian System.Security.Cryptography.

Ta bắt đầu bằng đoạn mã sau đây.

```
using System;
using System.IO;
using System.Text;
using System.Security.Cryptography;

public class crypt
{
    public static void Main()
    {
```

Trên hết, ta yêu cầu người dùng chọn một trong số mà người dùng liệt kê. Các dịch vụ mã hóa thông qua các nhà cung cấp dịch vụ (Service Provider) được cung cấp qua lớp SymmetricAlgorithm và nó cho phép thực hiện mã hóa và giải mã kèm theo một mã khóa.

```
        Console.WriteLine("Moi ban chon dich vu Ma Hoa:");
        Console.WriteLine("1 = DES");
        Console.WriteLine("2 = RC2");
        Console.WriteLine("3 = AES");
        Console.WriteLine("4 = 3DES");
        Console.WriteLine("5 = Ma hoa doi xung co dien");
```

```
// Tạo biến để mã hóa des, nó sẽ nhận một kiểu dữ liệu để dùng
SymmetricAlgorithm des = null;
```

```

switch (Console.ReadLine())
{
    case "1": des = new DESCryptoServiceProvider();           break;
    case "2": des = new RC2CryptoServiceProvider();           break;
    case "3": des = new RijndaelManaged();                     break;
    case "4": des = new TripleDESCryptoServiceProvider();       break;
    case "5": des= SymmetricAlgorithm.Create();                 break;
    default:
        Console.WriteLine ("Moi chon so tu 1 den 5 !!!");
        return;
}

```

T o m t i t n g FileStream l u tr d l i u m ã h ó a và b a o g ó i nó t r o n g i t n g CryptoStream. T o g i a o t i p ICryptoTransform b n g p h n g t h c CreateEncryptor (t h u c l p SymmetricAlgorithm) x á c n h các h o t n g m ã h ó a c b n.

```

FileStream fs = new FileStream("SecretFile.dat", FileMode.Create,
                               FileAccess.Write);

ICryptoTransform desencrypt = des.CreateEncryptor();
CryptoStream cryptostream = new CryptoStream(fs, desencrypt,
                                              CryptoStreamMode.Write);

```

Bây g i t a b t u m ã h ó a m t t h o n g i p n g i n. T r c h t t h o n g i p c c h u y n t h à n h m n g b y t e b n g p h n g t h c GetByte() c a l p Encoding t r o n g k h o n g g i a n System.Text. M n g n à y r i s c g h i v à o CryptoStream b n g p h n g t h c Write().

```

string theMessage = "A top secret message";
byte[] bytearrayinput = Encoding.Unicode.GetBytes(theMessage);
Console.WriteLine("Original Message : {0} ", theMessage);
cryptostream.Write(bytearrayinput, 0, bytearrayinput.Length);
cryptostream.Close();
fs.Close();

```

Sau k h i ã ó n g l u n g fs t h à n h c o n g, t a c ó t h t h c h i n g i i m ã t h o n g i p ã m ã h ó a. D l i u m ã h ó a c l n l t c t f i l e và c h u y n i n g c l i n h b n g c.

```

/*****Thi hành giai mã*****/
// T o l u n g FileStream c f i l e m ã h ó a

```

```

        FileStream fsread = new FileStream("SecretFile.dat", FileMode.Open,
                                           FileAccess.ReadWrite);

        byte [] encByte = new byte[fsread.Length];
        fsread.Read(encByte, 0, encByte.Length);

```

âý, ta xác ñh chỉ u dài c ñ thì t c a m ñg byte b ñg thu c tính Length có trong lu ñg FileStream ñh m m c ích l y h t d ñi u có trong file. L ñ l t c d ñi u t file vào m ñg b ñg ph ñg th c Read(). Tr c khi gi i mã, ta hi ñ th toàn b ñ ñi dung mã hóa ra màn hình. Sau khi in ra màn hình, con tr ñu ñg file cu i cùng, ta b t bu c nó tr v v trí u lu ñg.

```

        Console.WriteLine ("Encrypted Message : " +
                             Encoding.ASCII.GetString(encByte));

        fsread.Position = 0;

```

Th t c gi i mã d ñi u không khác ñhi u so v i mã hóa. S khác bi t ch y u âý ch ñà cách dùng ph ñg th c CreateDecryptor(), nó t o i t ñg gi i mã. T o m i m ñg byte l u tr k t qu gi i mã. Ph ñg th c GetString() c a l p Encoding s bi ñ i m ñg byte thành chu i.

```

// T o i t ñg gi i mã t i t ñg des
ICryptoTransform desdecrypt = des.CreateDecryptor();
CryptoStream cryptostreamDecr = CryptoStream(fsread, desdecrypt,
                                               CryptoStreamMode.Read);

byte[] decrByte = new byte[fsread.Length];
cryptostreamDecr.Read(decrByte, 0, (int)fsread.Length);
string output = Encoding.Unicode.GetString(decrByte);
Console.WriteLine("Decrypted Message : {0}" ,output);
cryptostreamDecr.Close();
fsread.Close();
}
}

```