

# IO.Compression and Encryption Error

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.windowsforms/2006-11/msg00255.h>

---

- *From:* "AnikSol" <[anisol@xxxxxxxxxxxxxxxxxxxx](mailto:anisol@xxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Tue, 14 Nov 2006 23:41:06 +0800
- 

Hi,

Environment: VB 2005 Professional

Both ends, its the application which is doing the functionalities.  
At the despatch end, We have a text file which is to be compressed and encrypted and sent over to the receipient.  
At the receipient end, the file is decrypted and then decompressed.

Say the original file is CreateTestPort3.txt  
CreateTestPort3.txt: 1306 KB (original file)

TestGZipped:151 KB (compressing the original file using GZipStream)

EncryptText:151 KB (encrypting the compressed file)

DecryptText: 76 KB (decrypting the earlier file)

CreateTestPort3GOriginal: 0 KB (decompressing the file)

We receive an error at this stage:

Error : An unhandled exception of type 'System.IO.InvalidDataException' occurred in System.dll

Additional information: The magic number in GZip header is not correct. Make sure you are passing in a GZip stream.

Can anyone point out what are we doing wrong? If we just compress and decompress the file using the same code, its perfect. If we encrypt and decrypt the file using the same class, its perfect. But if we do the combination of compression and encryption, the file is totally messed up. Any reasons?

For the sake of testing, we did both the functionalities of receipient and despatch end in the same project...

## IO.Compression and Encryption Error

The code for above is here:

```
Dim SecKey As String = "Tashgktt"

Private Sub cmdComGZipped_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmdComGZipped.Click
Dim outfile, infile As FileStream

Dim zipStream As GZipStream

Dim sw As StreamWriter

Dim sr As StreamReader

Dim ms As MemoryStream

Dim info As FileInfo

infile = New FileStream("Y:\CreateTestPort3.txt", FileMode.Open,
FileAccess.Read)

outfile = New FileStream("Y:\TestGZipped.xip", FileMode.Create,
FileAccess.Write)

zipStream = New GZipStream(outfile, CompressionMode.Compress, False)

sr = New StreamReader(infile)

sw = New StreamWriter(zipStream)

Do While Not sr.EndOfStream

sw.WriteLine(sr.ReadLine)

Loop

sw.Close()

sr.Close()

zipStream.Close() ' important to close this first to flush compressed stream

outfile.Close() ' important to close this second to flush output stream

infile.Close()

info = New FileInfo("Y:\TestGZipped.Xip")

MsgBox(info.Length.ToString)

IO.Compression and Encryption Error
```

## IO.Compression and Encryption Error

//Encrypt the file

```
EncryptFile("Y:\TestGZipped.xip", "Y:\EncryptText.enc", SecKey)
```

End Sub

Private Sub cmdDecomGZipped\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdDecomGZipped.Click

Dim outfile, infile As FileStream '

Dim ZipStream As GZipStream

Dim sw As StreamWriter

Dim sr As StreamReader

Dim info As FileInfo

//Decrypt the file

```
DecryptFile("Y:\EncryptText.enc", "Y:\DecryptText.XIP", SecKey)
```

```
infile = New FileStream("Y:\DecryptText.XIP", FileMode.Open,  
FileAccess.Read)
```

```
outfile = New FileStream("Y:\CreateTestPort3GOriginal.txt", FileMode.Create,  
FileAccess.Write)
```

```
ZipStream = New GZipStream(infile, CompressionMode.Decompress, False)
```

```
sr = New StreamReader(ZipStream)
```

```
sw = New StreamWriter(outfile)
```

```
Do While Not sr.EndOfStream
```

```
sw.WriteLine(sr.ReadLine)
```

```
Loop
```

```
sw.Close()
```

```
sr.Close()
```

```
ZipStream.Close() ' important to close this first to flush compressed stream
```

```
outfile.Close() ' important to close this second to flush output stream
```

## IO.Compression and Encryption Error

```
infile.Close()
```

```
info = New FileInfo("Y:\CreateTestPort3GOriginal.txt")
```

```
MsgBox(info.Length.ToString)
```

```
End Sub
```

Error : An unhandled exception of type 'System.IO.InvalidDataException' occurred in System.dll

Additional information: The magic number in GZip header is not correct. Make sure you are passing in a GZip stream.

The Encrypt/Decrypt is done by the following:

```
Imports System.IO
```

```
Imports System.Security.Cryptography
```

```
Imports System.Runtime.InteropServices
```

```
Imports System.Text
```

```
Module EncryptDecrypt
```

```
' Call this function to remove the key from memory after it is used for security.
```

```
<DllImport("kernel32.dll")> _
```

```
Public Sub ZeroMemory(ByVal addr As IntPtr, ByVal size As Integer)
```

```
End Sub
```

```
' Function to generate a 64-bit key.
```

```
Function GenerateKey() As String
```

```
' Create an instance of a symmetric algorithm. The key and the IV are generated automatically.
```

```
Dim desCrypto As DESCryptoServiceProvider =  
DESCryptoServiceProvider.Create()
```

```
' Use the automatically generated key for encryption.
```

```
Return ASCIIEncoding.ASCII.GetString(desCrypto.Key)
```

IO.Compression and Encryption Error

## IO.Compression and Encryption Error

End Function

```
Sub EncryptFile(ByVal sInputFilename As String, _
ByVal sOutputFilename As String, _
ByVal sKey As String)
Dim fsInput As New FileStream(sInputFilename, _
FileMode.Open, FileAccess.Read)
Dim fsEncrypted As New FileStream(sOutputFilename, _
FileMode.Create, FileAccess.Write)
'Dim Key = sKey
'Dim IV = sKey
Dim DES As New DESCryptoServiceProvider()
'Set secret key for DES algorithm.
'A 64-bit key and an IV are required for this provider.
DES.Key = ASCIIEncoding.ASCII.GetBytes(sKey)
'Set the initialization vector.
DES.IV = ASCIIEncoding.ASCII.GetBytes(sKey)
'Create the DES encryptor from this instance.
Dim desencrypt As ICryptoTransform = DES.CreateEncryptor()
'Create the crypto stream that transforms the file stream by using DES
encryption.
Dim cryptostream As New CryptoStream(fsEncrypted, _
desencrypt, _
CryptoStreamMode.Write)
'Read the file text to the byte array.
Dim bytearrayinput(fsInput.Length - 1) As Byte
fsInput.Read(bytearrayinput, 0, bytearrayinput.Length)
IO.Compression and Encryption Error
```

## IO.Compression and Encryption Error

'Write out the DES encrypted file.

```
cryptostream.Write(bytearrayinput, 0, bytearrayinput.Length)
```

```
cryptostream.Close()
```

```
fsInput.Close()
```

```
fsEncrypted.Close()
```

End Sub

```
Sub DecryptFile(ByVal sInputFilename As String, _
```

```
ByVal sOutputFilename As String, _
```

```
ByVal sKey As String)
```

```
'Dim Key = sKey
```

```
'Dim IV = sKey
```

```
Dim DES As New DESCryptoServiceProvider()
```

```
'A 64-bit key and an IV are required for this provider.
```

```
'Set the secret key for the DES algorithm.
```

```
DES.Key() = ASCIIEncoding.ASCII.GetBytes(sKey)
```

```
'Set the initialization vector.
```

```
DES.IV = ASCIIEncoding.ASCII.GetBytes(sKey)
```

```
'Create the file stream to read the encrypted file back.
```

```
Dim fsread As New FileStream(sInputFilename, FileMode.Open, FileAccess.Read)
```

```
'Create the DES decryptor from the DES instance.
```

```
Dim desdecrypt As ICryptoTransform = DES.CreateDecryptor()
```

```
'Create the crypto stream set to read and to do a DES decryption transform  
on incoming bytes.
```

```
Dim cryptostreamDecr As New CryptoStream(fsread, desdecrypt,  
CryptoStreamMode.Read)
```

```
'Print out the contents of the decrypted file.
```

## IO.Compression and Encryption Error

```
Dim fsDecrypted As New StreamWriter(sOutputFilename)
fsDecrypted.Write(New StreamReader(cryptostreamDecr).ReadToEnd)
fsDecrypted.Flush()
fsDecrypted.Close()
fsread.Close()
cryptostreamDecr.Close()
End Sub
End Module
```