

Re: simple string encryption

Source:

<http://www.tech-archive.net/Archive/VB/microsoft.public.vb.general.discussion/2008-07/msg00179.html>

- *From:* "Mike Williams" <gagamomo@xxxxxxxxxxxxxxxx>
 - *Date:* Thu, 3 Jul 2008 17:34:00 +0100
-

"Matt Williamson" <ih8spam@xxxxxxxxxxxx> wrote in message
news:%23tzqHNS3IHA.4476@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

I'm working on a little project that requires lightweight string encryption and I found this: <http://www.devx.com/tips/Tip/5676> which seems to be a fast and small xor encryption algorithm. It works fine based upon the example, but if I dump the encrypted string into a textbox and then read the string from the textbox and feed it back to the routine, it doesn't decrypt the string. Does adding it to a textbox introduce any additional characters that throw off the routine?

Yes, it does. The reason is because strings in VB use two bytes per character and your code, as it stands, is messing about with both bytes of each character. This causes the VB TextBox to fail to understand the characters when you pass it the string, because a standard TextBox is not unicode aware and the character byte that it expects to be zero is something other than zero. This causes the TextBox to fail to recognise the character and so it changes it to a standard question mark and inserts those question mark characters into the TextBox. When you later transfer the contents of the TextBox to a String for running through the encryption routine again (to decrypt it) you are actually passing the routine a string consisting of just a bunch of question marks, which of course causes it to fail to perform its task. There are various ways around this problem but perhaps the simplest is to prevent the encryption routine from messing about with the "zero bytes" of each character (which it is only setting to the value 38 anyway and so it lends virtually nothing to the encryption itself). One way would be to change the code so that it converts into a Byte Array at one byte per character instead of two (using StrConv) but perhaps a simpler way would be to leave it as it is and simply Step 2 instead of 1 when running through the Byte Array. Try changing this part of the routine:

```
For INum = LBound(bytData) To UBound(bytData)
If INum Mod 2 Then
bytData(INum) = bytData(INum) Xor (bytKey(INum) _
+ KEY_OFFSET)
Else
bytData(INum) = bytData(INum) Xor (bytKey(INum) _
- KEY_OFFSET)
End If
Next INum
```

to :

Re: simple string encryption

```
For INum = LBound(bytData) To UBound(bytData) Step 2  
bytData(INum) = bytData(INum) Xor (bytKey(INum) _  
- KEY_OFFSET)  
Next INum
```

Of course, if you want the encryption routine to encrypt and decrypt Unicode characters then you actually do want it to "mess about with" both bytes of the characters (which it is already doing) but then of course you cannot pass the encrypted string to and from a standard TextBox, which is not Unicode aware.

Mike