

Re: Copying string to byte array

Source:

<http://www.tech-archive.net/Archive/VB/microsoft.public.vb.general.discussion/2005-04/msg01196.html>

- *From:* Tony Proctor <tony_proctor@xx>
 - *Date:* Tue, 12 Apr 2005 01:51:03 -0700
-

Sorry, only just became aware of this old thread that talked about the use of Strings and the CryptEncrypt + CryptDecrypt APIs.

In general, binary data should not be held in String variables. There are 2 main reasons for this: a) not all character codes are valid in a given character set (incl. Unicode), and b) those codes will get mangled if the character set is changed (e.g. a Unicode-to-ANSI conversion when passing it to an API)

In the specific case of the Microsoft CryptoAPI, I can confirm that every single VB6 example I've seen on the Web is wrong, and uses String data for the input/output buffer (apologies if anyone has noticed and has also written a corrected version that I haven't seen). They're all passing a Unicode String variable, and using the character count as both the buffer and data lengths. The documentation is explicit that it requires a plain memory buffer and byte lengths, but I suspect the examples have all been inherited from one original bad source a long time ago.

OK, so why is it wrong? Well, whatever you put into your String is being converted to the ANSI character set when passed to the API (during encryption). This appears to work because most people have only ever tested "Latin 1" characters. Hence, the converted text then only occupies one byte per character, and the setting of the buffer/data lengths to the character count just happens to be right. There are 3 ways that this can fail, and it's not hard to force any of them:-

- 1) If you try to run it in a locale that isn't using a single-byte ANSI character set then the buffer/data lengths will no longer be correct
- 2) If you encrypt with one code page active, but decrypt whilst a different one is active then you won't get back what you started with
- 3) The conversion between the Unicode and ANSI character sets may fail due to an illegal character code being generated. This is unlikely to happen if you stick with the same single-byte ANSI character set, but can happen in scenarios (1) and (2)

In reality, the data buffer should be a byte array, and the data/buffer lengths should always be in bytes. Since encrypting the Unicode text directly will more often than not be encrypting a lot of redundant NUL characters, I

Re: Copying string to byte array

prefer to convert it to UTF-8 before encryption, and from UTF-8 after decryption. This removes the sensitivity to the current ANSI code page setting by selecting what is effectively a universal MBCS.

Tony Proctor

"Sam Hobbs" wrote:

> "Jim Mack" <jmack@xxxxxxxxxxxxxxxx> wrote in message
> news:O9kLO2Q%23EHA.3700@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
>>
>> VB strings cannot contain arbitrary binary data. This is the reason
>> (stated by MS) that byte arrays were introduced into the language.
>
> Where does Microsoft state that VB strings cannot contain arbitrary binary
> data? I found a KB article that says it will work. See Article ID:122289
> ("Passing Structures in OLE Automation") that says that serializing a
> structure into a BSTR will work "if both the controller and server are
> 32-bit and both support Unicode", which is what is relevant here. Note that
> for this context, "serializing" means storing. Also note that (beginning
> with VB 4 at least) VB uses the BSTR OLE automation data type for strings.
>
> Unless you are aware of something as definitive or more definitive stating
> the opposite, then you should not post comments that waste people's time.
>
>> Don't even bother trying, it's a well-travelled road that leads to
>> misery. :-)
>>
>> Bottom line: for arbitrary binary data, use byte arrays and never let it
>> touch a VB string -- not for file I/O, not for API parameters -- never.
>
> Then please explain why strings are used for the Platform SDK "CryptEncrypt"
> function's input and output data. At least every sample I have seen does,
> including Microsoft samples. The input and output data is the fifth
> ("pbData") parameter for CryptEncrypt.
>
> I seldom use the word "wrong" to state explicitly that there is a problem
> with an answer, but in this case, I truly believe this answer to definitely
> be wrong.
>
> I was not sure which newsgroup to ask this question in, but I will post a
> question in the microsoft.public.vb.winapi newsgroup about use of strings
> for the CryptEncrypt data.
>
>
>
>

• Prev by Date: [paramètres régionaux](#)

Re: Copying string to byte array

- Next by Date: ***Fax Component***
- Previous by thread: ***paramètres régionaux***
- Next by thread: ***Fax Component***
- Index(es):
 - ◆ ***Date***
 - ◆ ***Thread***