

Encoding byte array in C#

Published 3/23/2010 by [Vivek Thakur](#)

I noticed a developer encoding an already RSA encrypted byte array to a string format, so that he can save it in database. He was using this code:

```
byte[] message = Encoding.UTF8.GetBytes("string to be encrypted");
byte[] encrypted = rsaAlgo.PrivateEncryption(message);
string save_in_db = Encoding.UTF8.GetString(encrypted);
```

Then he simply wanted to decrypt save_in_db to its original value, using the following code:

```
byte[] message = Encoding.UTF8.GetBytes(save_in_db);
byte[] decrypted = myRsa.PublicDecryption(message );
string originalMsg = Encoding.UTF8.GetString(decrypted);
```

But unfortunately he did not get the original string back. Any guesses? The answer is simple, he first encrypted his original string using RSA and it generated random binary sequence. He then tried to encode these random binary values back to a string format, which of-course is not possible as some values may not have any string representation at all. So what is the solution?

He should use `Convert.ToBase64String` and `Convert.FromBase64String` instead of using `Encoding.UTF8.GetString()` method. For example:

```
byte[] message = Encoding.UTF8.GetBytes("string to be encrypted");
byte[] encrypted = rsaAlgo.PrivateEncryption(message);
string save_in_db = Convert.ToBase64String(encrypted);
```

And,

```
byte[] message = Convert.FromBase64String (save_in_db);
byte[] decrypted = myRsa.PublicDecryption(message );
string originalMsg = Encoding.UTF8.GetString(decrypted);
```

tags : c#, Encoding