

How to write a referrer spam application with C#

By Scott Horne (<http://web-professor.net>)

This tutorial covers how to write a referrer spam generator in C#. You can download all the project files from [this zip](#) and load it as a project in Visual Studio.

What is referrer spam

Referrer spam makes it appear as if visitors have been sent to a site from another site when they have not. There are several goals that can be sought after by the spammer. One is to grab the attention of the website owner and cause them to visit your site in hopes of talking about the spammers site in a positive way. Another goal would be to get indexed by a search engine if the search engine robot reads the websites logs. Lastly this can be done to generate commercial interest in a site. This is a not meant to be a complete overview of referrer spam if you require more information please visit [sydk8 forums](#) and ask around in the forums.

What you need to start

It assumes that you already have a version of Visual Studio installed and know a minimal amount of C#. You can get a free version of [Visual Studio Express from Microsoft](#). If you don't know C# yet:

1. Visual Studio has excellent documentation included
2. Charles Petzold has a [free C# primer](#)
3. O'Reilly C# books are always good for [C/C++ programmers](#) and for [new programmers](#) .
4. [Petzolds book](#) is a classic and is an inevitable buy for anyone who is serious about C#.

Ok got an idea on how to write a basic C# app already ? Lets GO!

Getting Started

The first part of our project setting up the form and controls we will be using to do our evil deeds for us.

Start a project:

1. Click File -> New Project
2. Select "Windows Application" and name the application "Referrer Spam"
3. Select the Form: Change the Text property to "Referrer Spam", Change the name to RefSpam
4. From your toolbox drag a button in and place it at the top left.
 - Change the Text property to read "Start"
 - Change the Name property to read "btnStart"
5. From your toolbox drag a button in and place it to the right of the Start button.
 - Change the Text property to read "Stop"
 - Change the Name property to read "btnStop"
6. From your toolbox drag in 2 labels and 2 text boxes
 - Position them on the right side of the form and alternate them from top to bottom label, textbox, label, textbox.
 - Select the top label: Change the Text property to read "From URL", Change the Name to "lblFromURL"
 - Select the top text box: Change the Name property to "txtFromURL"
 - Select the bottom label: Change the Text property to "Target URLs", change the Name

- property to "lblTargetURLs"
- Select the bottom textbox: Change the Name property to "txtTargetURLs", Change the multiline property to true, Resize the text box so that the bottom of the box meets the bottom of the form, and change the Anchor property to read "Top, Left, Bottom"
- 7. From the toolbox drag in a webbrowser control and place it to the right of your text boxes
 - In the properties for the webbrowser control change the Name to "wbBrowser"
 - Change the anchor property to read "Left, Right, Bottom, Top"
- 8. Drag a label and place it above the webbrowser control.
 - Change the Text Property to "Current Target"
 - Change the Name to "lblBrowser"

There you have it a complete working shell of everything we need for a basic referral spam app. Go ahead and punch f5 to spin that beauty up if you want. Exit out when your done admiring your work and lets get this thing finished.

Getting started wiring it up

Now we need to put it all together so that when you hit the start button you will get some results. We will be doing minimal of error checking to keep the tutorial simple.

Right hand click your application design view and select view code. From here you will see the applications code which is pretty minimal.

The class is going to contain three instance variables, an array of strings that will hold the target urls, a int that indicates which target url we are to browse to next and a bool that will let the object know when to stop browsing.

Inside that class add the following to meet those requirements:

```
private string[] TargetUrls; // array of urls that are the targets
private int CurrentUrl; // the next position in the list of target urls to be browsed
private bool KeepBrowsing; // sentinal value that indicates if the instance of RefSpam should
continue to browse
```

Then we need to add the constructor to initialize all those values.

```
public RefSpam()
{
    InitializeComponent();
    CurrentUrl = 0;
    KeepBrowsing = false;
}
```

Inorder to iterate through the array of of target urls we need to add a function that will walk over the array and go back to the begining of the loop when done. We can acheive this by adding a function that will do just that and return the current url as a string for use by the function that loads the site in the browser.

Add to the RefSpam class:

```
private string getNextTargetUrl()
{
    string url;
    if (TargetUrls.Length > 0)
    {
        // did we reach the end ?
```

```

        if (CurrentUrl >= TargetUrls.Length)
        {
            // lets restart at the top of the list
            CurrentUrl = 0;
        }

        url = TargetUrls[CurrentUrl];
        CurrentUrl++;

    }
    else
    {
        url = "";
    }
    return url;
}

```

Now to consume the urls spit out by the getNextTargetURL we add the actual function that navigates our webbrowser control and send the referral url we want to send.

```

private void goToNextTargetUrl()
{
    wbBrowser.Navigate( getNextTargetUrl(), null, null, "Referrer: " + txtFromURL + "\n" );
}

```

After the document at the URL loads we want the browser to go to the next target URL. To achieve this we add an event handler to the webbrowser that gets called when ever the browser completes loading. To do this right hand click the screen and select "View Designer". Select the webbrowser control and in the properties click the lightning bolt to see the events that we can add handlers for on that control. Double click the DocumentCompleted... this will cause the creation of a function to handle that event. In that function we create the following:

```

private void wbBrowser_DocumentCompleted(object sender,
WebBrowserDocumentCompletedEventArgs e)
{
    if (KeepBrowsing == true)
    {
        goToNextTargetUrl();
    }
}

```

Testing to see if it should go on it then calls goToNextTargetUrl() to advance the browser to the next target.

We also need a way to alert the user that there was an error. In the Referral spammer class we are going to add a function call "NotifyError". After the constructor function add the following:

Function:

```

private void NotifyError()
{
    MessageBox.Show("There has been an error, check to make sure all fields have been entered");
}

```

This function will display message to the user in case of an error when we call it.

Start the spam

Now right hand click the screen again and select "View Designer" from the menu. Then double click the Start button, this automatically creates a function that will fire when we click the button and drops us into that empty function so we can flesh it out. Our first job is to check the mayke sure our textfields are filled in and if they aren't then warn the user. (*We will assume that if there is text in them that they are valid URLs. This is something that you might want to validate on your own later.*) We create a local sentinel variable called error of type bool and set it to false. Then if either text control is empty we set it to true, display an error to our user and exit the function. If its not true then we perform the actual work of the program. Lastly we import the urls from the control txtTargetURLs and start the browsing by calling goToNextTarget. Lets see what the code looks like right now.

Code so far:

```
private void btnStart_Click(object sender, EventArgs e)
{
    bool error = false;

    // testing the length of the text in the "From URL" box
    if (txtFromURL.Text.Length > 0)
    {
    }
    else
    {
        error = true;
    }

    // testing the see the number of lines of text in the "Target URLs" box
    if (txtTargetURLs.Lines.Length > 0)
    {
    }
    else
    {
        error = true;
    }

    // was there an error ?
    if (error)
    {
        // tell the user
        NotifyError();
    }
    else
    {
        // here is where all the action happens baby !!!
        TargetUrls = txtTargetURLs.Lines;
        goToNextTargetUrl();
    }

    return;
}
```

Stop the Spam

To do this right hand click the screen and select "View Designer". Double clickt he stop button and fill in the following to stop the browser from navigating.

```
private void btnStop_Click(object sender, EventArgs e)
{
```

```
    KeepBrowsing = false;  
}
```

Thats it your done ! You now have completed an application that will leave a fake referral on the target sites. Build and test that bad boy out.

Files:

- [Project Files](#)
- Just [install the application](#) (install only works in Internet Explorer)

Please visit <http://web-professor.net> for more tools, tips, and scripts by [Web Professor](#)