

Using Bluestem Authentication in the ASP.NET Environment

or How to Avoid Bluestem Code in Every Page

Dale Sinder - CITES

This document is available at
<http://staffdb.cites.uiuc.edu/BlueStem.NET/>

This document is available at <http://staffdb.cites.uiuc.edu/BlueStem.NET/>

Bluestem - University of Illinois WWW Identification Service

BlueStem.cs - For Bluestem Active Server Pages (ASP.NET) VB.NET, C#...

Namespace: Edu.Uiuc.Cites.Bluestem for dotNet 1.1 Edu.Uiuc.Ccso.Bluestem
for dotNET 1.0

(C) Copyright 1996-2003 - Board of Trustees of the University of Illinois

This module based on original work by Bill Reynen, U of I Foundation

Adapted for C# by Dale Sinder, Campus Information Technologies and
Educational Services

Dale Sinder - sinder@uiuc.edu

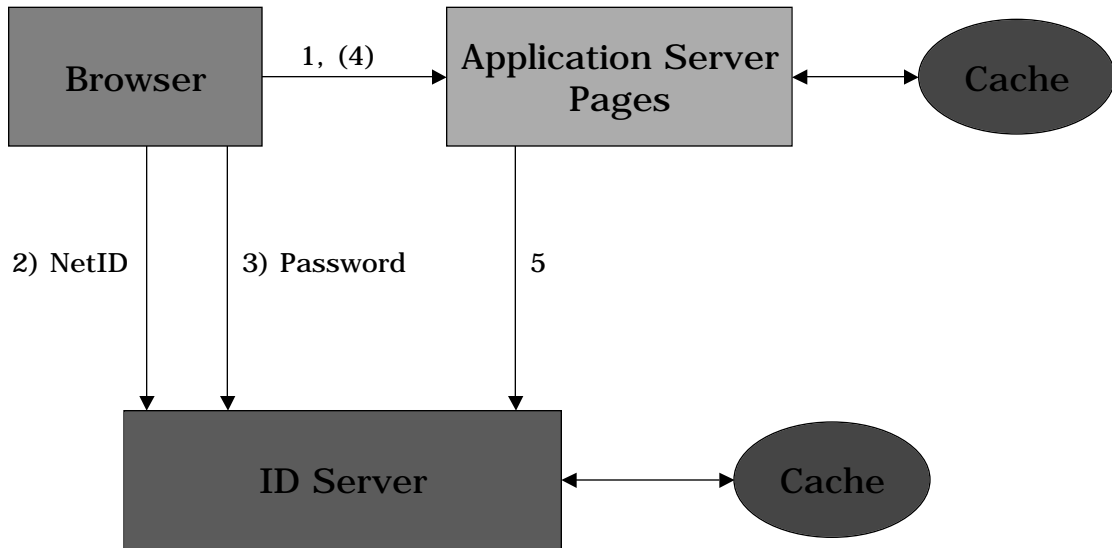
CITES - Campus Information Technologies and Educational Services

University of Illinois - Urbana-Champaign

Agenda

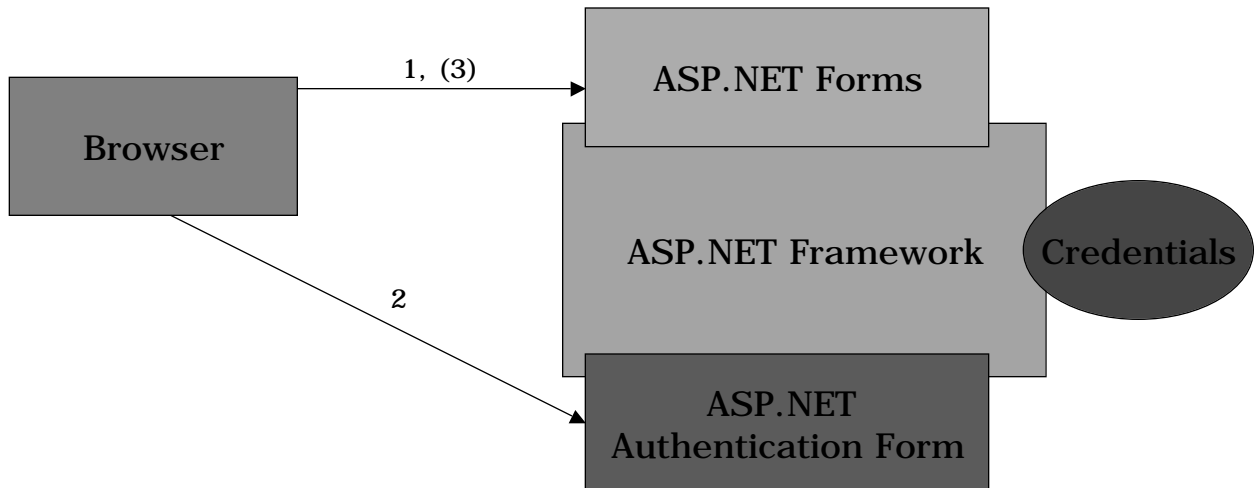
- Overview of Bluestem protocol
- Overview of ASP.NET Forms Authentication
- Overview of Putting the Two Together
- The Bluestem Class
- Example use in VB.NET and C#
- Summary - Resources

Bluestem Protocol Overview



1. A Browser makes a request for a Bluestem protected page. Code included in the page calls the little bluestem API to determine via cache if user is already authenticated. If so page is served to browser.
2. If user is not Authenticated the browser is redirected to the Login page on a Bluestem ID server.
3. User then enters password on a new next page.
4. Browser is redirected back to the page that was originally requested.

Forms Authentication Overview



1. Browser requests a page in an ASP.NET application that uses forms authentication. The ASP.NET framework checks if the user has already authenticated. If so, the Form contents are sent to the browser.
2. If the user is not already Authenticated the ASP.NET framework redirects the browser to the Authentication Form specified in the web.config file. This Form implements whatever is needed to authenticate the user and tells the Framework the users Identity.
3. Browser is redirected back to the Form originally requested.

The Bluestem Class

- Provides all the Bluestem functionality – You still must still install the Bluestem Distribution
- The Authentication Form you provide glues the Bluestem Class to your application and provides any custom authentication and authorization you need (perhaps by referring to a Database)

A Look at the Bluestem Class

The Are Two Versions:

One for dotNet 1.0

One for dotNET 1.1

Either version of the library can be with dotNET 1.0 and 1.1.

The differences in versions here are only for proper integration with Visual Studio .NET

A One Form Application Who Am I?

- Example in VB.NET
- Example in C#

Users Identity is:

Me.Context.User.Identity.Name

this.Context.User.Identity.Name

sinder@uiuc.edu/kerberos

Steps:

Download and unzip one of the Bluestem.NET distributions

Create a VB.NET or C# Web Application.

Add these sections to the Web.config file:

```
<appSettings>
  <add key="BluestemVirtDir" value="/scripts/bluestem" />
  <add key="BluestemLib" value="D:\Bluestem\lib" />
  <add key="BluestemLog" value="D:\Bluestem\Logs\bluestem" />
</appSettings>
```

Change locations above as needed for your Bluestem installation.

```
<authentication mode="Forms">
  <forms name=".ASPXFORMSAUTH" loginUrl="BlueStemLogin.aspx" />
</authentication>
<authorization>
  <deny users="?" />
  <allow users="*" />
</authorization>
```

For VB.NET the Global.asax.vb file should be modified to contain:

```
Sub Application_Start(ByVal sender As Object, ByVal e As EventArgs)
  Application["BluestemVirtDir"] = ConfigurationSettings.AppSettings("BluestemVirtDir").ToString()
  Application["BluestemLib"] = ConfigurationSettings.AppSettings("BluestemLib").ToString()
  Application["BluestemLog"] = ConfigurationSettings.AppSettings("BluestemLog").ToString()
End Sub
```

For C# modify Global.asax.cs to contain:

```
protected void Application_Start(Object sender, EventArgs e)
{
  Application["BluestemVirtDir"] =
ConfigurationSettings.AppSettings["BluestemVirtDir"].ToString();
  Application["BluestemLib"] =
ConfigurationSettings.AppSettings["BluestemLib"].ToString();
  Application["BluestemLog"] =
ConfigurationSettings.AppSettings["BluestemLog"].ToString();
}
```

Create a new Web Form Named "BlueStemLogin"

Model code in "BlueStemLogin" after the examples given in the distributions

Steps To Do Bluestem.NET

- Create Web Project
- Add Reference to Edu.Uiuc.Cites.Bluestem v1.1 or Edu.Uiuc.Ccso.Edu v1.0
- Edit Web.config:
 - Add <appSettings> Section:
 - <appSettings>
 - <add key="BluestemVirtDir" value="/scripts/bluestem" />
 - <add key="BluestemLib" value="D:\Bluestem\lib" />
 - <add key="BluestemLog" value="D:\Bluestem\Logs\bluestem" />
 - <add key="AppOff" value="false" />
 - </appSettings>

The appSettings are loaded into Application vars in Global.asax

- **Edit Authentication Section:**

```
< authentication mode= "Forms" >  
    < forms name= ".ASPXFORMSAUTH"  
        loginUrl= "BlueStemLogin.aspx" />  
< /authentication >
```

- **Edit Authorization Section:**

```
< authorization >  
    < deny users= "?" />  
    < allow users= "*" />  
< /authorization >
```

Deny unauthenticated users and allow authenticated users as a default.

- Add Location Section:

```
<location path="get_aspx_ver.aspx">  
  <system.web>  
    <authorization>  
      <allow users="*" />  
      <allow users="?" />  
    </authorization>  
  </system.web>  
</location>
```

get_aspx_ver.aspx must be exempt from authentication to avoid problems with loading into Visual Studio .NET

- Edit Global.asax (.vb / .cs):

Add to Application_Start:

```
Application("BluestemVirtDir") =  
    ConfigurationSettings.AppSettings("BluestemVirtDir").ToString()  
Application("BluestemLib") =  
    ConfigurationSettings.AppSettings("BluestemLib").ToString()  
    Application("BluestemLog") =  
    ConfigurationSettings.AppSettings("BluestemLog").ToString()
```

```
Application["BluestemVirtDir"] =  
    ConfigurationSettings.AppSettings["BluestemVirtDir"].ToString();  
Application["BluestemLib"] =  
    ConfigurationSettings.AppSettings["BluestemLib"].ToString();  
Application["BluestemLog"] =  
    ConfigurationSettings.AppSettings["BluestemLog"].ToString();
```

Read values from Web.config into Application vars.

-
- Create Web Form BlueStemLogin.aspx
 - Add code using the distributed Samples as a Model

```
Protected myBlueStem As Edu.Uiuc.Cites.Bluestem.BlueStem
Private Sub Page_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
Response.Cache.SetCacheability(HttpCacheability.NoCache)
myBlueStem = New Edu.Uiuc.Cites.Bluestem.BlueStem
Dim ca As Edu.Uiuc.Cites.Bluestem.CustomAuthenticate
Dim href As String
ca = New Edu.Uiuc.Cites.Bluestem.CustomAuthenticate(AddressOf
MyCallBack)
href = myBlueStem.Authenticate(ca)
If (href <> Nothing) Then
    ' handle intermediate panel setup here
    linkLabel.Text = "<a href=" + href + ">Click here to enter</a>" & _
        "<script language='javascript'>top.location.href='" & _
        href & "'</script>"
    linkLabel.Visible = True
    errorLabel.Visible = False
End If
End Sub
```

In Page_Load: Create a Bluestem Class and call the Authenticate method with a callback address for your custom authentication/authorization code.

```

Public Function MyCallBack() As Boolean
    Dim idfull, Username, domain, authmethod As String

    idfull = Me.myBlueStem.FullID
    Username = Me.myBlueStem.UserName
    domain = Me.myBlueStem.Domain
    authmethod = Me.myBlueStem.AuthMethod

    Session("UserName") = Username

    If (Not domain.Equals("uiuc.edu")) Then
        errorLabel.Text = "You are not permitted access<br>You must login in the
        <b>uiuc.edu</b> domain."
        errorLabel.Visible = True
        linkLabel.Visible = False
        Return False
    End If
    If (Not authmethod.Equals("kerberos")) Then
        errorLabel.Text = "You are not permitted access<br>You must login in using the
        <b>kerberos</b> authentication method."
        errorLabel.Visible = True
        linkLabel.Visible = False
        Return False
    End If
    ' Add any custom authentication or authorization code here based on user ID
    ' Return True if the user is OK, else Return False
    Return True
End Function

```

Set Session Variable to hold user name.

Check authentication domain and method.

Add any custom authentication or authorization code here based on user ID appropriate to your application.

Return True if the user is OK to use app, else Return False.

WebForm1.aspx:

```
<H3 align="center">You have been authenticated  
as&nbsp;UIUC NetID&nbsp;  
<asp:Label id="Label1" runat="server"></asp:Label></H3>  
<P align="center">(<asp:Label id="Label2"  
runat="server"></asp:Label>)</P>
```

WebForm1.aspx.vb

```
Private Sub Page_Load(ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles MyBase.Load  
Me.Label1.Text = Session("UserName").ToString()  
Me.Label2.Text = Me.User.Identity.Name  
End Sub
```

Display NetID from Session Variable and display full ID from User.Identity

Output of WebForm1

**You have been authenticated
as UIUC NetID sinder
(sinder@uiuc.edu/kerberos)**

Summary

- **Use of ASP.NET Forms Authentication and the Bluestem Helper Class isolates all Bluestem related code to one Form**
- **More Efficient after initial Authentication – No reading of cache files**

Resources

- **The Bluestem Class, Examples, and Documentation:**

<http://staffdb.cites.uiuc.edu/Bluestem.NET>

- **ASP.NET Security:**

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpcondesigningsecureaspnetapplications.asp>

- **Bluestem Protocol Details:**

<https://www-s.uiuc.edu/bluestem/notes/protocol.html>

- **Full Bluestem Documentation:**

<https://www-s.uiuc.edu/bluestem/notes/>