

Tech Note

LHM

SonicWALL LHM Scripts Reference

This document contains the complete text of the following Lightweight Hotspot Messaging (LHM) scripts:

- [Chooser.aspx Script](#)
- [Logout.aspx Script](#)
- [Accept Script](#)
- [ADAuth Script](#)
- [Guestbook Script](#)
- [LHMQuiz Script](#)
- [PayPal Script](#)
- [Random Script](#)

Chooser.aspx Script

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Xml" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Security" %>
<%@ Import namespace="System.Security.Cryptography.X509Certificates"%>

<script language="VB" runat="server">

Dim ip as String
Dim sessionId as String
Dim mac as String
Dim ufi as String
Dim mgmtBaseUrl as String
Dim clientRedirectUrl as String
Dim req as String
Dim hmac as String
Dim customCode as String
Dim qString as String

Sub Page_Load(src as Object, e as EventArgs)

    'Grab the querystring one element at a time since we need to do a custom URL encode on the req variable
    sessionId=Request.QueryString("sessionId")
    ip=Request.QueryString("ip")
    mac=Request.QueryString("mac")
    ufi=Request.QueryString("ufi")
    mgmtBaseUrl=Request.QueryString("mgmtBaseUrl")
    clientRedirectUrl=Request.QueryString("clientRedirectUrl")
    req=Request.QueryString("req")
    hmac=Request.QueryString("hmac")
    customCode=Request.QueryString("cc")

    'SonicWALL URL Encode routine is different from Microsoft - this is the SonicWALL method
    req=Replace(req,"%","%25")
    req=Replace(req,".", "%3A")
    req=Replace(req," ", "%20")
    req=Replace(req,"?", "%3F")
    req=Replace(req,"+", "%2B")
    req=Replace(req,"&", "%26")
    req=Replace(req,"=", "%3D")

    'Rebuild the querystring variable
```



Tech Note

```
& "&req=" & req

'Add the optional hmac and cc vars if they are there.
If hmac <> "" Then
    qString+="&hmac=" & hmac
End If

If customCode <> "" Then
    qString+="&cc=" & customCode
End If

'Bind the directory data
Dim lhmDir As New DirectoryInfo(Server.MapPath("."))
lhmList.DataSource = lhmDir.GetDirectories
lhmList.DataBind()

End Sub

</script>

<STYLE>
body {
    font-size: 10pt;
    font-family: verdana,helvetica,arial,sans-serif;
    color:#000000;
    background-color:#9CBACE;
}

tr.heading {
    background-color:#006699;
}

.button {
    border: 1px solid #000000;
    background-color: #ffffff;
}

tr.hidden {
    font-size: 5pt;
    color:#9CBACE;
}

</STYLE>

<HTML>
<HEAD>
<TITLE>LHM Script Chooser</TITLE>
</HEAD>

<BODY>

<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;</td>
    </tr>
    <tr class="heading">
        <td width="50%" valign="center"><font color="white"><b>LHM Script Chooser</b></font></td>
        <td><center></center></td>
        <td width="50%" align="right" valign="center"><font color="white"><b></b>&nbsp;</font></td>
    </tr>
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;</td>
    </tr>
</table>

<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr><td><br></td></tr>
    <tr><td><H3>Please select one of the LHM Scripts below</H3></td></tr>
    <tr><td>Your original querystring information will be passed to the target script, and it will open in a new window.</td></tr>
    <tr><td><br></td></tr>
</table>

<asp:Repeater id="lhmList" runat="server">
    <ItemTemplate >
        <li><a href = <%=# DataBinder.Eval(Container.DataItem, "Name").ToString() &
"/default.aspx?" & qString & " target=""_blank"" %> >
```



Tech Note

```
<%# DataBinder.Eval(Container.DataItem, "Name").ToString() %>
</a>
</li>
</ItemTemplate>
</asp:Repeater>

<table>
<tr class="hidden">
<td>default.aspx?sessionId=0b712fd83b9f5313db5af1cea6b1004f&ip=10.50.165.231&mac=00:0e:35:bd:c9:37&ufi=0006b11184300&mgmtBaseUrl=https://10.50.165.193:4043/&clientRedirectUrl=https://10.50.165.193:444/&req=http%3A/www.google.com/ig</td></tr>
</table>

</BODY>
</HTML>
```

Logout.aspx Script

The following logout.aspx script is identical for all of the following scripts (Accept, ADAUth, Guestbook, etc.). Each of the scripts can use the logout.aspx script if so configured in their myvars.aspx script.

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Xml" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Security" %>
<%@ Import namespace="System.Security.Cryptography.X509Certificates"%>

<!-- #INCLUDE file="myvars.aspx" -->

<script language="VB" runat="server">

'This class allows SSL certs signed by unknown CAs to be accepted.
'This is necessary for the POST to the SonicWALL authorizing the LHM session.
Public Class acceptAllCerts
    Implements System.Net.ICertificatePolicy
    Public Function CheckValidationResult(ByVal srvPoint As ServicePoint, _
        ByVal cert As X509Certificate, ByVal request As WebRequest, ByVal problem As Integer) _
        As Boolean Implements ICertificatePolicy.CheckValidationResult
        Return True
    End Function
End Class

Dim sessionId as String
Dim mgmtBaseUrl as String
Dim eventId as String = "&eventId=1"

'Grab the code and the session lifetime from the generator page
Sub Page_Load(src as Object, e as EventArgs)
    sessionId=Request.QueryString("sessId")
    mgmtBaseUrl=Request.QueryString("mgmtBaseUrl")
    sessTimer=Request.QueryString("sessTimer")

    'Use the override class in myvars.aspx to accept untrusted certificates from the SonicWALL
    'This is necessary for the POST to the SonicWALL authorizing the LHM session.
    System.Net.ServicePointManager.CertificatePolicy = New acceptAllCerts

    'When the page loads, make the loggedIn span visible
    loggedIn.Visible=True
    loggedOut.Visible=False

    Me.Button1.Attributes.Add("OnClick", "self.close()")

End Sub

'The Logout button
Sub btnSubmit_Click(Sender As Object, E As EventArgs)

    'Let the user know that we are setting up the session, just in case it takes more than a second
    LHMResult.Text = "Authorizing session. Please wait."

    'The LHM cgi on the SonicWALL - this does not change
    Dim loginCgi as String = "externalGuestLogoff.cgi"
```



Tech Note

```
'Assemble the data to post back to the SonicWALL to authorize the LHM session
Dim loginParams as String = "sessId=" & sessionId & eventId
```

```
'Combine mgmtBaseUrl from the original redirect with the login cgi
Dim postToSNWL as String = mgmtBaseUrl & loginCgi
```

```
'Convert the loginParams to a well behaved byte array
Dim byteArray As Byte() = Encoding.UTF8.GetBytes(loginParams)
```

```
Try
```

```
    'Make the loggedOut span visible
    loggedIn.Visible=False
    loggedOut.Visible=True
```

```
    'Create the webrequest to the SonicWALL
    Dim toSNWL as WebRequest = WebRequest.Create(postToSNWL)
```

```
    'Calculate the length of the byte array
    toSNWL.ContentLength = byteArray.Length
```

```
    'Set the method for the webrequest to POST
    toSNWL.Method = "POST"
```

```
    'Set the content type
    toSNWL.ContentType = "application/x-www-form-urlencoded"
```

```
    'Open the request stream
    Dim dataStream As Stream = toSNWL.GetRequestStream()
```

```
    'Write the byte array to the request stream
    dataStream.Write(byteArray, 0, byteArray.Length)
```

```
    'Close the Stream object
    dataStream.Close()
```

```
    'Get the response
    Dim snwlReply As WebResponse = toSNWL.GetResponse()
```

```
    'Display the status - looking for 200 = OK.
    'Response.Write(CType(snwlReply, HttpWebResponse).StatusCode)
```

```
    'Grab the response and stuff it into an xml doc for possible review
    Dim snwlResponse as XmlDocument = New XmlDocument()
    snwlResponse.Load(snwlReply.GetResponseStream())
```

```
    'Set the xPath to the SNWL reply, and get the response
    Dim codePath as String = "SonicWALLAccessGatewayParam/LogoffReply/ResponseCode"
```

```
    'Response.Write(snwlResponse.SelectSingleNode(codePath).InnerXml)
```

```
    'Response code 150 - Logout Succeeded
    If snwlResponse.SelectSingleNode(codePath).InnerXml = "150"
        LHMResult.Text = "<br><b><font color=""green"">Your session has been logged out.<br><br>Thank you for using LHM Guest
```

```
Services.</font></b>"
```

```
    'Response code 251 - Bad HMAC.
    ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "251"
```

```
        LHMResult.Text = "<br><b><font color=""red"">Session logout failed:</font></b> The request failed message authentication. Sorry for the
inconvenience. Please close and relaunch your browser to try again."
```

```
    'Response code 253 - Invalid SessionID.
    ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "253"
```

```
        LHMResult.Text = "<br><b><font color=""red"">Session logout failed:</font></b> The request failed to match a known session identity.
Sorry for the inconvenience. Please close and relaunch your browser to try again."
```

```
    'Response code 254 - Invalid CGI.
    ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "254"
```

```
        LHMResult.Text = "<br><b><font color=""red"">Session logout failed:</font></b> The request was missing an essential parameter. Sorry
for the inconvenience. Please close and relaunch your browser to try again."
```

```
    'Response code 255 - Internal Error.
    ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "255"
```

```
        LHMResult.Text = "<br><b><font color=""red"">Session logout failed:</font></b> The request failed due to an unspecified error. Sorry for
the inconvenience. Please close and relaunch your browser to try again."
```

```
End If
```

```
'Close the streams
```



Tech Note

```
        dataStream.Close()
        snwlReply.Close()

        'If there is some asp.net error trying to talk to the SonicWALL, print it in the same color as the background.
        Catch ex as Exception
            catchError.Text = "<font color=""9CBACE"">" & ex.ToString & "</font>"
            LHMResult.Text = "<br><b><font color=""red"">Session logout failed:</font></b> The request failed due to an unspecified error. Sorry for
the inconvenience. Please close and relaunch your browser to try again. If the problem persists, please notify an attendant."
            End Try
        End Sub

</script>
<STYLE>
body {
    font-size: 10pt;
    font-family: verdana,helvetica,arial,sans-serif;
    color:#000000;
    background-color:#9CBACE;
}

tr.heading {
    font-size: 10pt;
    background-color:#006699;
}

tr.smalltext {
    font-size: 8pt;
}

.button {
    border: 1px solid #000000;
    background-color: #ffffff;
    font-size: 8pt;
}
</STYLE>

<HTML>
<HEAD>
<TITLE>LHM Logout Page</TITLE>

<SCRIPT LANGUAGE="Javascript">

//Javascript Seconds Countdown Timer
var SecondsToCountDown = <%= sessTimer%>;
var originalTime=" ";

function CountDown()
{
    clockStr="";

    dayStr=Math.floor(SecondsToCountDown/86400)%100000
    if(dayStr>0){
        if(dayStr>1){
            dayStr+=" days ";
        } else dayStr+=" day ";
        clockStr=dayStr;
    }
    hourStr=Math.floor(SecondsToCountDown/3600)%24
    if(hourStr>0){
        if(hourStr>1){
            hourStr+=" hours ";
        } else hourStr+=" hour ";
        clockStr+=hourStr;
    }
    minuteStr=Math.floor(SecondsToCountDown/60)%60
    if(minuteStr>0){
        if(minuteStr>1){
            minuteStr+=" minutes ";
        } else minuteStr+=" minute ";
        clockStr+=minuteStr;
    }
    secondStr=Math.floor(SecondsToCountDown/1)%60
    if(secondStr>0){
        if(secondStr>1){
            secondStr+=" seconds ";
        } else secondStr+=" second ";
        clockStr+=secondStr;
    }
}
```



Tech Note

```
if(SecondsToCountDown > 0)
{
    --SecondsToCountDown;
}

if(originalTime.length < 2)
{
    originalTime = clockStr;
}

// Make sure the form is still there before trying to set a value
if(document.frmValidator){
    document.frmValidator.originalTime.value = originalTime;
    document.frmValidator.countdown.value = clockStr;
}

setTimeout("CountDown()", 1000);
if(SecondsToCountDown == 0)
{
    document.frmValidator.countdown.value = "Session Expired";
}
}

//Disable right-click so that the window doesn't get refreshed since the countdown is clientside.
document.oncontextmenu = disableRightClick;
function disableRightClick()
{
    return false;
}

//Disable F5 key, too, on IE at least.
function noF5()
{
    var key_f5 = 116;
    if (key_f5==event.keyCode)
    {
        event.keyCode=0;
        return false;
    }
    return false;
}

document.onkeydown=noF5
document.onmousedown=disableRightClick

</SCRIPT>

</HEAD>

<BODY onload='CountDown()'>
<span id="loggedIn" runat="server">
<form id="frmValidator" runat="server">
<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr class="heading">
        <td colspan=2 align="center">&nbsp;</td>
    </tr>
    <tr class="heading">
        <td colspan=2 align="center"><font color="white"><b>SonicWALL LHM Logout Window</b></font></td>
    </tr>
    <tr class="heading">
        <td colspan=2 align="center">&nbsp;</td>
    </tr>
    <tr class="smalltext"><td><br></td></tr>
    <tr class="smalltext">
        <td>Original Session Time:</td>
        <td><asp:textbox width=250 id="originalTime" runat="server" /></td>
    </tr>
    <tr class="smalltext">
        <td>Remaining Session Time:</td>
        <td><asp:textbox width=250 id="countdown" runat="server" /></td>
    </tr>
    <tr class="smalltext">
        <td colspan=2><br>You may use this window to manually logout your session at any time, or you may safely close this window if you prefer to let your
session timeout automatically.</font></td>
    </tr>
</tr>
```



Tech Note

```
<td colspan=2><center><asp.button id="btnSubmit" class="button" text=" Logout " onClick="btnSubmit_Click" runat="server" /></center></td>
</tr>
</table>
</form>
</span>

<span id="loggedOut" runat="server">
<form id="logout" runat="server">
<table width="100%" border="0" cellpadding="2" cellspacing="0">
  <tr class="heading">
    <td colspan=2 align="center">&nbsp;</td>
  </tr>
  <tr class="heading">
    <td colspan=2 align="center"><font color="white"><b>SonicWALL LHM Logout Window</b></font></td>
  </tr>
  <tr class="heading">
    <td colspan=2 align="center">&nbsp;</td>
  </tr>
  <tr>
    <td><asp.Label id=LHMResult runat="server" /></td>
  </tr>
  <tr>
    <td><asp.Label id=catchError runat="server" /></td>
  </tr>
  <tr><td><br></td></tr>
  <tr>
    <td><center><asp.button id="Button1" class="button" text=" Close " runat="server" /></center></td>
  </tr>
</table>
</form>
</span>

</BODY>
</HTML>
```



Tech Note

Accept Script

[myvars.aspx](#)

[default.aspx](#)

myvars.aspx

```
<script language="VB" runat="server">

'Set the logoutPopup window flag - 0 = no popup, 1 = popup
'The use of the logoutPopup in this script is encouraged because the login event is non-exclusive.
Dim logoutPopup as String = "1"

'Set the LHM Session Timeout
Dim sessTimer as String = "3600"

'Set the LHM Idle Timeout
Dim idleTimer as String = "300"

'Set the username to record for LHM session since this does not gather one. Set to userName="useMAC" to use the MAC address.
Dim userName="useMAC"
'Dim userName = "LHM Guest User"

'Set the secret for use with optional HMAC auth, as configured in the Extern Guest Auth config on the SonicWALL
Dim strHmac as String = "password"

'Set the digest method for the HMAC, either MD5 or SHA1
Dim hmacType as String = "MD5"
'Dim hmacType as String = "SHA1"

'Set the logo image to use
Dim logo as String = "sonicwall.gif"

'-----End of Configurable Settings-----

</script>
```

default.aspx

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Xml" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Security" %>
<%@ Import namespace="System.Security.Cryptography.X509Certificates"%>

<!-- #INCLUDE file="myvars.aspx" -->

<script language="VB" runat="server">

'This class allows SSL certs signed by unknown CAs to be accepted.
'This is necessary for the POST to the SonicWALL authorizing the LHM session.
Public Class acceptAllCerts
    Implements System.Net.ICertificatePolicy
    Public Function CheckValidationResult(ByVal srvPoint As ServicePoint, _
        ByVal cert As X509Certificate, ByVal request As WebRequest, ByVal problem As Integer) _
        As Boolean Implements ICertificatePolicy.CheckValidationResult
        Return True
    End Function
End Class

'Sample LHM redirect querystring:
'http://127.0.0.1/lhm/accept/default.aspx?sessionId=0b712fd83b9f5313db5af1cea6b1004f&ip=10.50.165.231&mac=00:0e:35:bd:c9:37&ufi=0006b11184300&mgmtBaseUrl=ht
tps://10.50.165.193:4043/&clientRedirectUrl=https://10.50.165.193:444/&req=http%3A/www.google.com/ig

Dim ip as String
Dim sessionId as String
Dim mac as String
Dim ufi as String
Dim mgmtBaseUrl as String
Dim clientRedirectUrl as String
```



Tech Note

```
Dim req as String
Dim hmac as String
Dim customCode as String
```

```
Sub Page_Load(src as Object, e as EventArgs)
```

```
    LHMResult.Text=""
    catchError.Text=""
```

```
    ip=Request.QueryString("ip")
    sessionId=Request.QueryString("sessionId")
    mac=Request.QueryString("mac")
    ufi=Request.QueryString("ufi")
    mgmtBaseUrl=Request.QueryString("mgmtBaseUrl")
    clientRedirectUrl=Request.QueryString("clientRedirectUrl")
    req=Request.QueryString("req")
    hmac=Request.QueryString("hmac")
    customCode=Request.QueryString("cc")
```

'customCode grabs the "cc=" querystring value sent by the SonicWALL. This allows you to use the same 'page (e.g. this page) for the "Session Expiration" (?cc=2), "Idle Timeout" (?cc=3) and "Max Sessions" (?cc=4) page.

```
If customCode <> "" Then
    Select Case customCode
        Case "2"
            LHMResult.Text="<br><H3><font color=""red"">Your LHM session has expired. You may try to initiate a new session.</font></H3>"
        Case "3"
            LHMResult.Text="<br><H3><font color=""red"">You have exceeded your idle timeout. Please log back in.</font></H3>"
        Case "4"
            LHMResult.Text="<br><H3><font color=""red"">The maximum number of sessions has been reached. Please try again later.</font></H3>"
    End Select
End If
```

```
'Set the userName to the grabbed client MAC address if so configured in myvars
If userName = "useMAC" Then
    userName = mac
End If
```

'Use the override class in myvars.aspx to accept untrusted certificates from the SonicWALL
'This is necessary for the POST to the SonicWALL authorizing the LHM session.
System.Net.ServicePointManager.CertificatePolicy = New acceptAllCerts

'Note - the routine below for handling the hmac requires the use of the SonicSSL.dll and libeay.dll libraries.
'The DLL must be copied to the IIS server, and the SonicSSL dll must be registered with "regsvr32 sonicssl.dll"
If hmac <> "" Then

```
    'SonicWALL URL Encode routine is different from Microsoft - this is the SonicWALL method
    req=Replace(req,"%","%25")
    req=Replace(req,"%","%3A")
    req=Replace(req,"%","%20")
    req=Replace(req,"%?","%3F")
    req=Replace(req,"%+","%2B")
    req=Replace(req,"%&","%26")
    req=Replace(req,"%=","%3D")
```

```
    Dim strHmacText as String
    Dim objCrypto as Object
    Dim strHmacGenerated
    Dim loginError as String
```

```
    'Initialize the Crypto object
    objCrypto = Server.CreateObject("SonicSSL.Crypto")
```

```
    'The text to be encoded
    strHmacText = sessionId & ip & mac & ufi & mgmtBaseUrl & clientRedirectUrl & req
```

'Calculate the hash with a key strHmac, the return value is a string converted from the output sha1 binary.
'The hash algorithm (MD5 or SHA1) is configured in myvars and in the Extern Guest Auth config on the SonicWALL

```
If hmacType = "MD5" Then
    strHmacGenerated = objCrypto.hmac_md5(strHmacText, strHmac)
Else
    strHmacGenerated = objCrypto.hmac_sha1(strHmacText, strHmac)
End If
```

```
If strHmacGenerated <> hmac Then
    Dim hmacFail as String
    hmacFail = "<font color=""red"">The HMAC failed validation. Please notify an attendant.</font><br><br>"
    hmacFail+="<font color=""9CBACE"">Received HMAC: " & hmac &
    "<br>Calculated HMAC: " & strHmacGenerated & "<br>"
```



Tech Note

```
        hmacFail+="Make sure the digest functions on the SonicWALL and LHM server match.<br>"
        hmacFail+="Also make sure the shared secret on the SonicWALL and myvars match</font>"
        catchError.Text=hmacFail
    End If

End If

End Sub

Sub btnSubmit_Click(Sender As Object, E As EventArgs)

    'Let the user know that we are setting up the session, just in case it takes more than a second
    LHMResult.Text = "Authorizing session. Please wait."

    'The LHM cgi on the SonicWALL - this does not change
    Dim loginCgi as String = "externalGuestLogin.cgi"

    'Assemble the data to post back to the SonicWALL to authorize the LHM session
    Dim loginParams as String = "sessId=" & sessionId & "&userName=" & Server.URLEncode(userName) & "&sessionLifetime=" & sessTimer & "&idleTimeout=" &
idleTimer

    'Combine mgmtBaseUrl from the original redirect with the login cgi
    Dim postToSNWL as String = mgmtBaseUrl & loginCgi

    'Convert the loginParams to a well behaved byte array
    Dim byteArray As Byte() = Encoding.UTF8.GetBytes(loginParams)

    Try
        'Create the webrequest to the SonicWALL
        Dim toSNWL as WebRequest = WebRequest.Create(postToSNWL)

        'Calculate the length of the byte array
        toSNWL.ContentLength = byteArray.Length

        'Set the method for the webrequest to POST
        toSNWL.Method = "POST"

        'Set the content type
        toSNWL.ContentType = "application/x-www-form-urlencoded"

        'Open the request stream
        Dim dataStream As Stream = toSNWL.GetRequestStream()

        'Write the byte array to the request stream
        dataStream.Write(byteArray, 0, byteArray.Length)

        'Close the Stream object
        dataStream.Close()

        'Get the response
        Dim snwlReply As WebResponse = toSNWL.GetResponse()

        'Display the status - looking for 200 = OK.
        'Response.Write(CType(snwlReply, HttpWebResponse).StatusCode)

        'Grab the response and stuff it into an xml doc for possible review
        Dim snwlResponse as XmlDocument = New XmlDocument()
        snwlResponse.Load(snwlReply.GetResponseStream())

        'Set the xPath to the SNWL reply, and get the response
        Dim codePath as String = "SonicWALLAccessGatewayParam/AuthenticationReply/ResponseCode"

        'Response.Write(snwlResponse.SelectSingleNode(codePath).InnerXml)

        'Response code 50 - Login Succeeded
        If snwlResponse.SelectSingleNode(codePath).InnerXml = "50"

            'Do we want to provide a logout popup window?
            If logoutPopup = "1" Then
                'Popup hack using Javascript for logout window
                Dim sb As New System.Text.StringBuilder()
                sb.Append("<script language='javascript'>")
                sb.Append("window.open('logout.aspx?sessId=")
                sb.Append(Server.URLEncode(CStr(sessionId)))
                sb.Append("&mgmtBaseUrl=")
                sb.Append(Server.URLEncode(CStr(mgmtBaseUrl)))
                sb.Append("&sessTimer=")
                sb.Append(Server.URLEncode(CStr(sessTimer)))
            End If
        End If
    End Try
End Sub
```



Tech Note

```
sb.Append(", 'logOut', toolbar=no,")
sb.Append(" addressbar=no, menubar=no,")
sb.Append(" width=400, height=250");")
sb.Append("<")
sb.Append("/")
sb.Append("<script>")
RegisterStartupScript("stp", sb.ToString)
End If

LHMResult.Text = "<br><b><font color=""green"">Session Authorized:</font></b> You may now go to the URL you originally requested: <a
target=""_blank"" href="" & req & "">" & req & "</a>"

'Response code 51 - Session Limit Exceeded
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "51"
LHMResult.Text = "<br><b><font color=""red"">Session Limit Reached:</font></b> The maximum number of guest session has been
reached. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 100 - Login Failed.
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "100"
LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> Your session cannot be created at this time. Sorry for
the inconvenience. Please close and relaunch your browser to try again."

'Response code 251 - Bad HMAC.
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "251"
LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed message
authentication. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 253 - Invalid SessionID.
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "253"
LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed to match a known
session identity. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 254 - Invalid CGI.
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "254"
LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization was missing an essential
parameter. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 255 - Internal Error.
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "255"
LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed due to an
unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again."

End If

'Close the streams
dataStream.Close()
snwlReply.Close()

'If there is some asp.net error trying to talk to the SonicWALL, print it in the same color as the background.
Catch ex as Exception
catchError.Text = "<font color=""9CBACE"">" & ex.ToString & "</font>"
LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed due to an
unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again. If the problem persists, please notify an attendant."
End Try
End Sub

</script>

<STYLE>
body {
font-size: 10pt;
font-family: verdana,helvetica,arial,sans-serif;
color:#000000;
background-color:#9CBACE;
}

tr.heading {
background-color:#006699;
}

.button {
border: 1px solid #000000;
background-color: #ffffff;
}
</STYLE>

<HTML>
```



Tech Note

```
<HEAD>
<TITLE>LHM Accept Script</TITLE>
</HEAD>

<BODY>
<form id="frmValidator" runat="server">

<table width="100%" border="0" cellpadding="2" cellspacing="0">
  <tr class="heading">
    <td colspan=3 align="center"><font color="white">&nbsp;</td>
  </tr>
  <tr class="heading">
    <td width="50%" valign="center"><font color="white"><b>Welcome <%= ip%></b></font></td>
    <td align="center"></center></td>
    <td width="50%" align="right" valign="center"><font color="white"><b>Powered by SonicWALL LHM</b>&nbsp;</font></td>
  </tr>
  <tr class="heading">
    <td colspan=3 align="center"><font color="white">&nbsp;</td>
  </tr>
</table>

<table width="100%" border="0" cellpadding="2" cellspacing="0">
  <tr>
    <td align="left">
      By clicking the <b>Accept</b> button below, you accept the following terms of service:<br><br><b>
      1. You will not try to download bad things.<br>
      2. You will not try to upload bad things.<br>
      3. You will not try to use all the bandwidth so that others have none.<br>
      4. You will be happy when the SonicWALL blocks bad things from reaching you.</b><br><br>
    </td>
  </tr>
  <tr>
    <td><br><asp.button id="btnSubmit" class="button" text=" Accept " onClick="btnSubmit_Click" runat="server" /></td>
  </tr>
  <tr>
    <td><asp.Label id=LHMResult runat="server" /></td>
  </tr>
  <tr>
    <td><asp.Label id=catchError runat="server" /></td>
  </tr>
</table>
</form>
</BODY>
</HTML>
```



Tech Note

ADAuth Script

[myvars.aspx](#)
[default.aspx](#)

myvars.aspx

```
<script language="VB" runat="server">

'Set the logoutPopup window flag - 0 = no popup, 1 = popup
'The use of the logoutPopup in this script is encouraged because the login event is non-exclusive.
Dim logoutPopup as String = "1"

'Set the LDAP server IP or Name
Dim myLdapServer as String = "10.50.128.40"

'Set the LDAP domain
Dim myLdapDomain as String = "sv.us.sonicwall.com"

'Set the retrAttr to 0 to use default session and idle timeouts
'Set the retrAttr to 1 to try to retrieve the session and idle timeouts from LDAP attributes.
Dim retrAttr as String = "1"

'Set useCN=1 to use common name (e.g. "joe levy", non-Active Directory LDAP) for attribute retrieval (retrAttr).
'Set useCN=0 to use saMACcountName (e.g. "jlevy", Active Directory / Windows) for attribute retrieval.
Dim useCN as String = "0"

'If using retrAttr=1, you must define the ldap attributes from which to retrieve the values
'Set the ldap attribute from which to retrieve the session timeout value (use is optional)
Dim sessAttr as String = "pager"

'Set the ldap attribute from which to retrieve the idle timeout value (use is optional)
Dim idleAttr as String = "mobile"

'If retrAttr=0, of if no attributes value can be retrieved, use the following timeout values
'Set the default LHM Session Timeout (for when no attributes is retrieved)
Dim sessTimer as String = "3600"

'Set the default LHM Idle Timeout (for when no attributes is retrieved)
Dim idleTimer as String = "300"

'Set the secret for use with optional HMAC auth, as configured in the Extern Guest Auth config on the SonicWALL
Dim strHmac as String = "password"

'Set the digest method for the HMAC, either MD5 or SHA1
Dim hmacType as String = "MD5"
'Dim hmacType as String = "SHA1"

'Set the logo image to use
Dim logo as String = "sonicwall.gif"

'-----End of Configurable Settings-----
</script>
```



Tech Note

default.aspx

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Xml" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Math" %>
<%@ Import Namespace="System.DirectoryServices" %>
<%@ Import Namespace="System.Collections" %>
<%@ Import Namespace="System.Security" %>
<%@ Import namespace="System.Security.Cryptography.X509Certificates"%>
<%@ Assembly name="System.DirectoryServices, Version=1.0.3300.0, Culture=neutral,PublicKeyToken=b03f5f7f11d50a3a"%>
```

```
<!-- #INCLUDE file="myvars.aspx" -->
```

```
<script language="VB" runat="server">
```

'This class allows SSL certs signed by unknown CAs to be accepted.

'This is necessary for the POST to the SonicWALL authorizing the LHM session.

Public Class acceptAllCerts

Implements System.Net.ICertificatePolicy

Public Function CheckValidationResult(ByVal srvPoint As ServicePoint, _

ByVal cert As X509Certificate, ByVal request As WebRequest, ByVal problem As Integer) _

As Boolean Implements ICertificatePolicy.CheckValidationResult

Return True

End Function

End Class

'Sample LHM redirect querystring:

'http://127.0.0.1/lhm/adauth/default.aspx?sessionId=0b712fd83b9f5313db5af1cea6b1004f&ip=10.50.165.231&mac=00:0e:35:bd:c9:37&ufi=0006b11184300&mgmtBaseUrl=ht
tps://10.50.165.193:4043/&clientRedirectUrl=https://10.50.165.193:444/&req=http%3A/www.google.com/ig

Dim ip as String

Dim sessionId as String

Dim mac as String

Dim ufi as String

Dim mgmtBaseUrl as String

Dim clientRedirectUrl as String

Dim req as String

Dim hmac as String

Dim customCode as String

Sub Page_Load(src as Object, e as EventArgs)

LHMResult.Text=""

catchError.Text=""

authResult.Text=""

ip=Request.QueryString("ip")

sessionId=Request.QueryString("sessionId")

mac=Request.QueryString("mac")

ufi=Request.QueryString("ufi")

mgmtBaseUrl=Request.QueryString("mgmtBaseUrl")

clientRedirectUrl=Request.QueryString("clientRedirectUrl")

req=Request.QueryString("req")

hmac=Request.QueryString("hmac")

customCode=Request.QueryString("cc")

'customCode grabs the "cc=" querystring value sent by the SonicWALL. This allows you to use the same

'page (e.g. this page) for the "Session Expiration" (?cc=2), "Idle Timeout" (?cc=3) and "Max Sessions" (?cc=4) page.

If customCode <> "" Then

Select Case customCode

Case "2"

LHMResult.Text="
<H3>Your LHM session has expired. You may try to initiate a new session.</H3>"

Case "3"

LHMResult.Text="
<H3>You have exceeded your idle timeout. Please log back in.</H3>"

Case "4"

LHMResult.Text="
<H3>The maximum number of sessions has been reached. Please try again later.</H3>"

End Select

End If

'Use the override class in myvars.aspx to accept untrusted certificates from the SonicWALL

'This is necessary for the POST to the SonicWALL authorizing the LHM session.

System.Net.ServicePointManager.CertificatePolicy = New acceptAllCerts



Tech Note

'Note - the routine below for handling the hmac requires the use of the SonicSSL.dll and libeay.dll libraries.
'The DLL must be copied to the IIS server, and the SonicSSL dll must be registered with "regsvr32 sonicssl.dll"
If hmac <> "" Then

```
'SonicWALL URL Encode routine is different from Microsoft - this is the SonicWALL method
req=Replace(req,"%","%25")
req=Replace(req,".","%3A")
req=Replace(req," ","%20")
req=Replace(req,"?","%3F")
req=Replace(req,"+","%2B")
req=Replace(req,"&","%26")
req=Replace(req,"=","%3D")

Dim strHmacText as String
Dim objCrypto as Object
Dim strHmacGenerated
Dim loginError as String

'Initialize the Crypto object
objCrypto = Server.CreateObject("SonicSSL.Crypto")

'The text to be encoded
strHmacText = sessionId & ip & mac & ufi & mgmtBaseUrl & clientRedirectUrl & req

'Calculate the hash with a key strHmac, the return value is a string converted from the output sha1 binary.
'The hash algorithm (MD5 or SHA1) is configured in myvars and in the Extern Guest Auth config on the SonicWALL
If hmacType = "MD5" Then
    strHmacGenerated = objCrypto.hmac_md5(strHmacText, strHmac)
Else
    strHmacGenerated = objCrypto.hmac_sha1(strHmacText, strHmac)
End If

If strHmacGenerated <> hmac Then
    Dim hmacFail as String
    hmacFail = "<font color=""red"">The HMAC failed validation. Please notify an attendant.</font><br><br>"
    hmacFail+="<font color=""9CBACE"">Received HMAC: " & hmac & "<br>Calculated HMAC: " & strHmacGenerated & "<br>"
    hmacFail+="Make sure the digest functions on the SonicWALL and LHM server match.<br>"
    hmacFail+="Also make sure the shared secret on the SonicWALL and myvars match</font>"
    catchError.Text=hmacFail
End If
```

End If

End Sub

sub OnBtnClearClicked (Sender As Object, e As EventArgs)

```
txtName.Text = ""
txtPassword.Text = ""
authResult.Text=""
LHMResult.Text=""
catchError.Text=""
```

end sub

Sub btnSubmit_Click(Sender As Object, E As EventArgs)

'Try to connect to LDAP with the user supplied attributes
Try

```
Dim ldapPath as String = "LDAP://" & myLdapServer
Dim ldapUser as String = myLdapDomain & "\" & txtName.Text
Dim validateUser as New DirectoryEntry(ldapPath,ldapUser,txtPassword.Text)
```

'This is the actual authentication piece
Dim nativeCheck as Object = validateUser.NativeObject

'If retrAttr is set in the myvars file, attempt to retrieve the session and idle values from LDAP
If retrAttr = "1" Then

```
Dim mySearch as New DirectorySearcher(validateUser)
```

```
'Check the myvars for selecting either sAMAccountName or cn
If useCN = "0" Then
    mySearch.Filter = "(sAMAccountName=" & Server.URLEncode(txtName.Text) & ")"
Else
    mySearch.Filter = "(cn=" & Server.URLEncode(txtName.Text) & ")"
End If
```

```
mySearch.PageSize="1"
mySearch.PropertiesToLoad.Add(sessAttr)
mySearch.PropertiesToLoad.Add(idleAttr)
Dim adResult as SearchResult
```



Tech Note

```
'If we get results on the attribute query, set timer values
adResult = mySearch.FindOne
If Not (adResult is Nothing) Then
    If (adResult.Properties.Contains(sessAttr)) Then
        'Check to see if the LDAP value returned is a number
        Dim isNumber as New RegEx("^d+$")
        If (isNumber.IsMatch(adResult.Properties(sessAttr)(0).ToString())) Then
            sessTimer=adResult.Properties(sessAttr)(0).ToString()
        End If
    End If 'End If sessAttr
    If (adResult.Properties.Contains(idleAttr)) Then
        'Check to see if the LDAP value returned is a number
        Dim isNumber as New RegEx("^d+$")
        If (isNumber.IsMatch(adResult.Properties(idleAttr)(0).ToString())) Then
            idleTimer=adResult.Properties(idleAttr)(0).ToString()
        End If
    End If 'End if idleAttr
End If 'End if adResult is present
End If 'End if retrAttr is in use

authResult.Text="<font color=""green""><b>Credentials Accepted.</b></font><br>Session Lifetime: " & round(sessTimer/60) & " minutes.<br>Idle
Timer: " & round(idleTimer/60) & " minutes."

'Auth succeeded - move on to LHM Auth
LHM()

Catch ex as Exception
    catchError.Text = "<font color=""9CBACE"">" & ex.ToString & "</font>"
    authResult.Text="<font color=""Red""><b>Credentials Rejected.</b></font><br>Please enter a valid username and password. "
End Try

End Sub

Sub LHM()

'Let the user know that we are setting up the session, just in case it takes more than a second
LHMResult.Text = "Authorizing session. Please wait."

'The LHM cgi on the SonicWALL - this does not change
Dim loginCgi as String = "externalGuestLogin.cgi"

'Assemble the data to post back to the SonicWALL to authorize the LHM session
Dim loginParams as String = "sessId=" & sessionId & "sessionId=" & Server.URLEncode(txtName.Text) & "&sessionLifetime=" & sessTimer &
"&idleTimeout=" & idleTimer

'Combine mgmtBaseUrl from the original redirect with the login cgi
Dim postToSNWL as String = mgmtBaseUrl & loginCgi

'Convert the loginParams to a well behaved byte array
Dim byteArray As Byte() = Encoding.UTF8.GetBytes(loginParams)

Try
    'Create the webrequest to the SonicWALL
    Dim toSNWL as WebRequest = WebRequest.Create(postToSNWL)

    'Calculate the length of the byte array
    toSNWL.ContentLength = byteArray.Length

    'Set the method for the webrequest to POST
    toSNWL.Method = "POST"

    'Set the content type
    toSNWL.ContentType = "application/x-www-form-urlencoded"

    'Open the request stream
    Dim dataStream As Stream = toSNWL.GetRequestStream()

    'Write the byte array to the request stream
    dataStream.Write(byteArray, 0, byteArray.Length)

    'Close the Stream object
    dataStream.Close()

    'Get the response
    Dim snwlReply As WebResponse = toSNWL.GetResponse()

    'Display the status - looking for 200 = OK.
```



Tech Note

```
'Response.Write(CType(snlwReply, HttpWebResponse).StatusCode)

'Grab the response and stuff it into an xml doc for possible review
Dim snlwResponse as XmlDocument = New XmlDocument()
snlwResponse.Load(snlwReply.GetResponseStream())

'Set the xPath to the SNWL reply, and get the response
Dim codePath as String = "SonicWALLAccessGatewayParam/AuthenticationReply/ResponseCode"

'Response.Write(snlwResponse.SelectSingleNode(codePath).InnerXml)

'Response code 50 - Login Succeeded
If snlwResponse.SelectSingleNode(codePath).InnerXml = "50"

    'Do we want to provide a logout popup window?
    If logoutPopup = "1" Then
        'Popup hack using Javascript for logout window
        Dim sb As New System.Text.StringBuilder()
        sb.Append("<script language='javascript'">")
        sb.Append("window.open('logout.aspx?sessId=")
        sb.Append(Server.URLEncode(CStr(sessionId)))
        sb.Append("&mgmtBaseUrl=")
        sb.Append(Server.URLEncode(CStr(mgmtBaseUrl)))
        sb.Append("&sessTimer=")
        sb.Append(Server.URLEncode(CStr(sessTimer)))
        sb.Append("'",'logOut','toolbar=no,')
        sb.Append("addressbar=no,menubar=no,")
        sb.Append("width=400,height=250';")
        sb.Append("<")
        sb.Append("/")
        sb.Append("<script>")
        RegisterStartupScript("stp", sb.ToString)
    End If

    LHMResult.Text = "<br><b><font color=""green"">Session authorized:</font></b> You may now go to the URL you originally
requested: <a target=""_blank"" href=""& req & """"> & req & "</a>"

'Response code 51 - Session Limit Exceeded
ElseIf snlwResponse.SelectSingleNode(codePath).InnerXml = "51"
    LHMResult.Text = "<br><b><font color=""red"">Session Limit Reached:</font></b> The maximum number of guest session has
been reached. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 100 - Login Failed.
ElseIf snlwResponse.SelectSingleNode(codePath).InnerXml = "100"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> Your session cannot be created at this time.
Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 251 - Bad HMAC.
ElseIf snlwResponse.SelectSingleNode(codePath).InnerXml = "251"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed
message authentication. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 253 - Invalid SessionID.
ElseIf snlwResponse.SelectSingleNode(codePath).InnerXml = "253"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed to match
a known session identity. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 254 - Invalid CGI.
ElseIf snlwResponse.SelectSingleNode(codePath).InnerXml = "254"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization was missing
an essential parameter. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 255 - Internal Error.
ElseIf snlwResponse.SelectSingleNode(codePath).InnerXml = "255"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed due to
an unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again."

End If

'Close the streams
dataStream.Close()
snlwReply.Close()

'If there is some asp.net error trying to talk to the SonicWALL, print it in the same color as the background.
Catch ex as Exception
    catchError.Text = "<font color=""9CBACE"">" & ex.ToString & "</font>"
```



Tech Note

```
LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed due to
an unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again. If the problem persists, please notify an attendant."
End Sub
End Try
</script>

<STYLE>
body {
font-size: 10pt;
font-family: verdana,helvetica,arial,sans-serif;
color:#000000;
background-color:#9CBACE;
}

tr.heading {
background-color:#006699;
}

.button {
border: 1px solid #000000;
background-color: #ffffff;
}
</STYLE>

<HTML>
<HEAD>
<TITLE>LHM ADAuth Script</TITLE>
</HEAD>

<BODY>
<form id="frmValidator" runat="server">

<table width="100%" border="0" cellpadding="2" cellspacing="0">
<tr class="heading">
<td colspan=3 align="center"><font color="white">&nbsp;</td>
</tr>
<tr class="heading">
<td width="50%" valign="center"><font color="white"><b>LDAP/AD LHM Authentication</b></font></td>
<td align="center"></center></td>
<td width="50%" align="right" valign="center"><font color="white"><b>Powered by SonicWALL LHM</b>&nbsp;</font></td>
</tr>
<tr class="heading">
<td colspan=3 align="center"><font color="white">&nbsp;</td>
</tr>
</table>

<table width="90%" border="0" cellpadding="2" cellspacing="0">
<tr>
<td><b>Welcome <%= ip%> to SonicWALL's LHM AD/LDAP Authenticator.</b><br><br>Enter your LDAP or Active Directory username and password
to obtain secure guest internet access.<br><br>If your domain account specifies session timeout values, those values will be applied to your account, otherwise you will
receive the default one hour (60 minutes) of access with a five minute idle timeout.<br>
</td>
</tr>
</table>
<table width="100%" border="0" cellpadding="2" cellspacing="0">
<tr class="heading">
<td colspan=3 align="center"><font color="white">Authentication domain: <%=myLdapDomain%></td>
</tr>
</table>
<table width="100%" border="0" cellpadding="2" cellspacing="0">
<tr>
<td width="30%"><br>Enter your login name:</td>
<td width="30%"><asp:TextBox id="txtName" runat="server" /></td>
<td width="40%"><asp:RequiredFieldValidator id="valTxtName" ControlToValidate="txtName" ErrorMessage="Please enter your name."
runat="server" /></td>
</tr>
<tr>
<td width="30%"><br>Enter your password:</td>
<td width="30%"><asp:TextBox id="txtPassword" textmode="password" runat="server" /></td>
<td width="40%"><asp:RequiredFieldValidator id="valTxtPassword" ControlToValidate="txtPassword" ErrorMessage="Please enter your password."
runat="server" /></td>
</tr>
<tr>
<td></td><td><asp:Label id=authResult runat="server" />&nbsp;</td>
</tr>
<tr>
<td></td>
</tr>
</table>
```



Tech Note

```
<tr><td><asp.button id="btnSubmit" class="button" text=" Submit " onClick="btnSubmit_Click" runat="server" />  
      &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~  
</td><td><asp.button id="btnClear" class="button" text=" Clear All " CausesValidation="False" onClick="OnBtnClearClicked" runat="server" />  
</td></tr>  
<tr>  
<td colspan=2><asp.Label id=LHMResult runat="server" /></td>  
</tr>  
<tr>  
<td colspan=2><asp.Label id=catchError runat="server" /></td>  
</tr>  
</table>  
</form>  
</BODY>  
</HTML>
```



Tech Note

Guestbook Script

[myvars.aspx](#)

[default.aspx](#)

myvars.aspx

```
<script language="VB" runat="server">

'Set the logoutPopup window flag - 0 = no popup, 1 = popup
'The use of the logoutPopup in this script is discouraged because although the login event
'is non-exclusive, the login event produces data where redundancy is undesirable.
Dim logoutPopup as String = "0"

'Set the LHM Session Timeout
Dim sessTimer as String = "3600"

'Set the LHM Idle Timeout
Dim idleTimer as String = "300"

'Set the secret for use with optional HMAC auth, as configured in the Extern Guest Auth config on the SonicWALL
Dim strHmac as String = "password"

'Set the digest method for the HMAC, either MD5 or SHA1
Dim hmacType as String = "MD5"
'Dim hmacType as String = "SHA1"

'Set the logo image to use
Dim logo as String = "sonicwall.gif"

'-----End of Configurable Settings-----

</script>
```

default.aspx

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Xml" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Data" %>
<%@ Import Namespace="System.Data.OleDb" %>
<%@ Import Namespace="System.Security" %>
<%@ Import namespace="System.Security.Cryptography.X509Certificates"%>

<!-- #INCLUDE file="myvars.aspx" -->

<script language="VB" runat="server">

'This class allows SSL certs signed by unknown CAs to be accepted.
'This is necessary for the POST to the SonicWALL authorizing the LHM session.
Public Class acceptAllCerts
    Implements System.Net.ICertificatePolicy
    Public Function CheckValidationResult(ByVal srvPoint As ServicePoint, _
        ByVal cert As X509Certificate, ByVal request As WebRequest, ByVal problem As Integer) _
        As Boolean Implements ICertificatePolicy.CheckValidationResult
        Return True
    End Function
End Class

'Sample LHM redirect querystring:
'http://127.0.0.1/lhm/guestbook/default.aspx?sessionId=0b712fd83b9f5313db5af1cea6b1004f&ip=10.50.165.231&mac=00:0e:35:bd:c9:37&ufi=0006b11184300&mgmtBaseUr
l=https://10.50.165.193:4043/&clientRedirectUrl=https://10.50.165.193:444/&req=http%3A/www.google.com/ig

Dim ip as String
Dim sessionId as String
Dim mac as String
Dim ufi as String
Dim mgmtBaseUrl as String
Dim clientRedirectUrl as String
Dim req as String
```



Tech Note

```
Dim hmac as String
Dim customCode as String
```

```
Sub Page_Load(src as Object, e as EventArgs)
```

```
    LHMResult.Text=""
    catchError.Text=""
```

```
    ip=Request.QueryString("ip")
    sessionId=Request.QueryString("sessionId")
    mac=Request.QueryString("mac")
    ufi=Request.QueryString("ufi")
    mgmtBaseUrl=Request.QueryString("mgmtBaseUrl")
    clientRedirectUrl=Request.QueryString("clientRedirectUrl")
    req=Request.QueryString("req")
    hmac=Request.QueryString("hmac")
    customCode=Request.QueryString("cc")
```

```
'customCode grabs the "cc=" querystring value sent by the SonicWALL. This allows you to use the same
'page (e.g. this page) for the "Session Expiration" (?cc=2), "Idle Timeout" (?cc=3) and "Max Sessions" (?cc=4) page.
```

```
If customCode <> "" Then
    Select Case customCode
        Case "2"
            LHMResult.Text="<br><H3><font color=""red"">Your LHM session has expired. You may try to initiate a new session.</font></H3>"
        Case "3"
            LHMResult.Text="<br><H3><font color=""red"">You have exceeded your idle timeout. Please log back in.</font></H3>"
        Case "4"
            LHMResult.Text="<br><H3><font color=""red"">The maximum number of sessions has been reached. Please try again later.</font></H3>"
    End Select
End If
```

```
'Use the override class in myvars.aspx to accept untrusted certificates from the SonicWALL
```

```
'This is necessary for the POST to the SonicWALL authorizing the LHM session.
```

```
System.Net.ServicePointManager.CertificatePolicy = New acceptAllCerts
```

```
'Note - the routine below for handling the hmac requires the use of the SonicSSL.dll and libeay.dll libraries.
```

```
'The DLL must be copied to the IIS server, and the SonicSSL dll must be registered with "regsvr32 sonicssl.dll"
```

```
If hmac <> "" Then
```

```
    'SonicWALL URL Encode routine is different from Microsoft - this is the SonicWALL method
```

```
    req=Replace(req,"%","%25")
    req=Replace(req,"%","%3A")
    req=Replace(req,"%","%20")
    req=Replace(req,"%?","%3F")
    req=Replace(req,"%+","%2B")
    req=Replace(req,"%&","%26")
    req=Replace(req,"%=","%3D")
```

```
    Dim strHmacText as String
    Dim objCrypto as Object
    Dim strHmacGenerated
    Dim loginError as String
```

```
    'Initialize the Crypto object
```

```
    objCrypto = Server.CreateObject("SonicSSL.Crypto")
```

```
    'The text to be encoded
```

```
    strHmacText = sessionId & ip & mac & ufi & mgmtBaseUrl & clientRedirectUrl & req
```

```
    'Calculate the hash with a key strHmac, the return value is a string converted from the output sha1 binary.
```

```
    'The hash algorithm (MD5 or SHA1) is configured in myvars and in the Extern Guest Auth config on the SonicWALL
```

```
If hmacType = "MD5" Then
```

```
    strHmacGenerated = objCrypto.hmac_md5(strHmacText, strHmac)
```

```
Else
```

```
    strHmacGenerated = objCrypto.hmac_sha1(strHmacText, strHmac)
```

```
End If
```

```
If strHmacGenerated <> hmac Then
```

```
    Dim hmacFail as String
```

```
    hmacFail = "<font color=""red"">The HMAC failed validation. Please notify an attendant.</font><br><br>"
```

```
    hmacFail+="<font color=""9CBACE"">Received HMAC: " & hmac & "<br>Calculated HMAC: " & strHmacGenerated & "<br>"
```

```
    hmacFail+="Make sure the digest functions on the SonicWALL and LHM server match.<br>"
```

```
    hmacFail+="Also make sure the shared secret on the SonicWALL and myvars match</font>"
```

```
    catchError.Text=hmacFail
```

```
End If
```

```
End If
```



Tech Note

End Sub

```
sub OnBtnClearClicked (Sender As Object, e As EventArgs)
```

```
    txtName.Text = ""
    txtAddress.Text = ""
    txtCity.Text = ""
    txtState.Text = ""
    txtZip.Text = ""
    txtPhone.Text = ""
    txtEMail.Text = ""
    txtURL.Text = ""
    txtComment.Text = ""
    LHMResult.Text=""
    catchError.Text=""
```

```
end sub
```

```
Sub btnSubmit_Click(Sender As Object, E As EventArgs)
```

```
    Try
```

```
        'Try to write the submitted info to the database file
```

```
        Dim strConn as string = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & server.mappath("guestbook.mdb") & ";"
```

```
        Dim MySQL as string = "INSERT INTO Guestbook (Name, Address, City, State, Zip, Phone, EMail, URL, Comment) VALUES ('" & txtName.Text & "','" & txtAddress.Text & "','" & txtCity.Text & "','" & txtState.Text & "','" & txtZip.Text & "','" & txtPhone.Text & "','" & txtEMail.Text & "','" & txtURL.Text & "','" & txtComment.Text & "');"

```

```
        Dim MyConn as New OleDbConnection (strConn)
        Dim cmd as New OleDbCommand (MySQL, MyConn)
        MyConn.Open ()
        cmd.ExecuteNonQuery ()
        MyConn.Close ()
```

```
        Catch ex as Exception
```

```
            catchError.Text = "<font color=""9CBACE"">" & ex.ToString & "</font>"
```

```
    End Try
```

```
    'Let the user know that we are setting up the session, just in case it takes more than a second
```

```
    LHMResult.Text = "Authorizing session. Please wait."
```

```
    'The LHM cgi on the SonicWALL - this does not change
```

```
    Dim loginCgi as String = "externalGuestLogin.cgi"
```

```
    'Assemble the data to post back to the SonicWALL to authorize the LHM session
```

```
    Dim loginParams as String = "sessId=" & sessionId & "&userName=" & Server.URLEncode(txtName.Text) & "&sessionLifetime=" & sessTimer & "&idleTimeout=" & idleTimer
```

```
    'Combine mgmtBaseUrl from the original redirect with the login cgi
```

```
    Dim postToSNWL as String = mgmtBaseUrl & loginCgi
```

```
    'Convert the loginParams to a well behaved byte array
```

```
    Dim byteArray As Byte() = Encoding.UTF8.GetBytes(loginParams)
```

```
    Try
```

```
        'Create the webrequest to the SonicWALL
```

```
        Dim toSNWL as WebRequest = WebRequest.Create(postToSNWL)
```

```
        'Calculate the length of the byte array
```

```
        toSNWL.ContentLength = byteArray.Length
```

```
        'Set the method for the webrequest to POST
```

```
        toSNWL.Method = "POST"
```

```
        'Set the content type
```

```
        toSNWL.ContentType = "application/x-www-form-urlencoded"
```

```
        'Open the request stream
```

```
        Dim dataStream As Stream = toSNWL.GetRequestStream()
```

```
        'Write the byte array to the request stream
```

```
        dataStream.Write(byteArray, 0, byteArray.Length)
```

```
        'Close the Stream object
```

```
        dataStream.Close()
```

```
        'Get the response
```

```
        Dim snwlReply As WebResponse = toSNWL.GetResponse()
```

```
        'Display the status - looking for 200 = OK.
```

```
        'Response.Write(CType(snwlReply, HttpWebResponse).StatusCode)
```



Tech Note

```
'Grab the response and stuff it into an xml doc for possible review
Dim snwlResponse as XmlDocument = New XmlDocument()
snwlResponse.Load(snwlReply.GetResponseStream())

'Set the xPath to the SNWL reply, and get the response
Dim codePath as String = "SonicWALLAccessGatewayParam/AuthenticationReply/ResponseCode"

'Response.Write(snwlResponse.SelectSingleNode(codePath).InnerXml)

'Response code 50 - Login Succeeded
If snwlResponse.SelectSingleNode(codePath).InnerXml = "50"

    'Do we want to provide a logout popup window?
    If logoutPopup = "1" Then
        'Popup hack using Javascript for logout window
        Dim sb As New System.Text.StringBuilder()
        sb.Append("<script language='javascript'>")
        sb.Append("window.open('logout.aspx?sessId=")
        sb.Append(Server.URLEncode(CStr(sessionId)))
        sb.Append("&mgmtBaseUrl=")
        sb.Append(Server.URLEncode(CStr(mgmtBaseUrl)))
        sb.Append("&sessTimer=")
        sb.Append(Server.URLEncode(CStr(sessTimer)))
        sb.Append("&logOut','toolbar=no,")
        sb.Append("addressbar=no,menubar=no,")
        sb.Append("width=400,height=250');")
        sb.Append("<")
        sb.Append("/")
        sb.Append("<script>")
        RegisterStartupScript("stp", sb.ToString)
    End If

    LHMResult.Text = "<br><b><font color='green'>Session authorized:</font></b> You may now go to the URL you originally requested: <a
target=''_blank'' href=''' & req & ''''> & req & '</a>"

'Response code 51 - Session Limit Exceeded
ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "51"
    LHMResult.Text = "<br><b><font color='red'>Session Limit Reached:</font></b> The maximum number of guest session has been
reached. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 100 - Login Failed.
ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "100"
    LHMResult.Text = "<br><b><font color='red'>Session creation failed:</font></b> Your session cannot be created at this time. Sorry for
the inconvenience. Please close and relaunch your browser to try again."

'Response code 251 - Bad HMAC.
ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "251"
    LHMResult.Text = "<br><b><font color='red'>Session creation failed:</font></b> The request for authorization failed message
authentication. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 253 - Invalid SessionID.
ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "253"
    LHMResult.Text = "<br><b><font color='red'>Session creation failed:</font></b> The request for authorization failed to match a known
session identity. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 254 - Invalid CGI.
ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "254"
    LHMResult.Text = "<br><b><font color='red'>Session creation failed:</font></b> The request for authorization was missing an essential
parameter. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 255 - Internal Error.
ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "255"
    LHMResult.Text = "<br><b><font color='red'>Session creation failed:</font></b> The request for authorization failed due to an
unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again."

End If

'Close the streams
dataStream.Close()
snwlReply.Close()

'If there is some asp.net error trying to talk to the SonicWALL, print it in the same color as the background.
Catch ex as Exception
    catchError.Text = "<font color='9CBACE'>" & ex.ToString & "</font>"
    LHMResult.Text = "<br><b><font color='red'>Session creation failed:</font></b> The request for authorization failed due to an
unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again. If the problem
persists, please notify an attendant."
```



Tech Note

```
End Sub
End Try
</script>

<STYLE>
body {
font-size: 10pt;
font-family: verdana,helvetica,arial,sans-serif;
color:#000000;
background-color:#9CBACE;
}

tr.heading {
background-color:#006699;
}

.button {
border: 1px solid #000000;
background-color: #ffffff;
}
</STYLE>

<HTML>
<HEAD>
<TITLE>LHM Guestbook Script</TITLE>
</HEAD>

<BODY>
<form id="frm\Validator" runat="server">

<table width="100%" border="0" cellpadding="2" cellspacing="0">
<tr class="heading">
<td colspan=3 align="center"><font color="white">&nbsp;</td>
</tr>
<tr class="heading">
<td width="50%" valign="center"><font color="white"><b>LHM Guestbook</b></font></td>
<td colspan=2 align="center"></td>
<td width="50%" align="right" valign="center"><font color="white"><b>Powered by SonicWALL LHM</b>&nbsp;<font></td>
</tr>
<tr class="heading">
<td colspan=3 align="center"><font color="white">&nbsp;</td>
</tr>
</table>

<table width="90%" border="0" cellpadding="2" cellspacing="0">
<tr>
<td>
<td>Welcome <%= ip%> to SonicWALL's LHM Guestbook. In exchange for providing us with your contact information,
along with your permission to occasionally contact you while you are in the middle of dinner, we will
provide you with <b>one complimentary hour of secure internet access.</b><br>
</td>
</tr>
</table>
<table width="100%" border="0" cellpadding="2" cellspacing="0">
<tr class="heading">
<td colspan=3><font color="white"><center><b>Thank you for your participation.</b></center></td>
</tr>
</table>
<table width="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td width="30%"><br>Enter your full name:</td>
<td width="30%"><asp:TextBox id="txtName" runat="server" /></td>
<td width="40%"><asp:RequiredFieldValidator id="valTxtName" ControlToValidate="txtName" ErrorMessage="Please enter your name."
runat="server" /></td>
</tr>
<tr>
<td width="30%"><br>Enter your address:</td>
<td width="30%"><asp:TextBox id="txtAddress" runat="server" /></td>
<td width="40%"><asp:RequiredFieldValidator id="valTxtAddress" ControlToValidate="txtAddress" ErrorMessage="Please enter your address."
runat="server" /></td>
</tr>
<tr>
<td width="30%"><br>Enter your city:</td>
<td width="30%"><asp:TextBox id="txtCity" runat="server" /></td>
<td width="40%"><asp:RequiredFieldValidator id="valTxtCity" ControlToValidate="txtCity" ErrorMessage="Please enter your city." runat="server"
/></td>
</tr>
<tr>
<td width="30%"><br>Enter your State:</td>
```



Tech Note

[illegible]

Tech Note

LHMQuiz Script

[myvars.aspx](#)

[default.aspx](#)

[quiz.aspx](#)

myvars.aspx

```
<script language="VB" runat="server">

'Set the logoutPopup window flag - 0 = no popup, 1 = popup
'The use of the logoutPopup in this script is discouraged because although the login event
'is non-exclusive, the login event produces data where redundancy is undesirable.
Dim logoutPopup as String = "0"

'Set the passing score
Dim passingScore as Integer = 80

'Set the filename of the quiz XML source
Dim quizFile as String = "quiz.xml"
'Dim quizFile as String = "shortquiz.xml"

'Set the name of the Quiz
Dim quizName as String = "SonicOS Quiz"

'Set the emailed quiz results "from" email address
Dim quizFrom as String = "joelevy@sonicwall.com"

'Set the email address to send failed test results to (the proctor/instructor)
Dim quizTo as String = "joelevy@sonicwall.com"

'Set the path for check and block embedded images - usually the same path as the quiz
Dim imagePath as String = "C:\inetpub\wwwroot\lhm\lhmquiz\"

'Set the IP or resolvable FQDN for the SMTP Server
'Make sure the server is configured to relay from the IP address of this server
'If setting to 127.0.0.1 (local IIS SMTP), you need to allow IIS SMTP to relay from 127.0.0.1
Dim smtpServer as String = "127.0.0.1"

'Set the LHM Session Timeout
Dim sessTimer as String = "86400"

'Set the LHM Idle Timeout
Dim idleTimer as String = "3600"

'Set the secret for use with optional HMAC auth, as configured in the Extern Guest Auth config on the SonicWALL
Dim strHmac as String = "password"

'Set the digest method for the HMAC, either MD5 or SHA1
Dim hmacType as String = "MD5"
'Dim hmacType as String = "SHA1"

'Set the logo image to use
Dim logo as String = "sonicwall.gif"

'-----End of Configurable Settings-----
</script>
```



Tech Note

default.aspx

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
```

```
<!-- #INCLUDE file="myvars.aspx" -->
```

```
<script runat="server">
```

'Sample LHM redirect querystring:

'http://10.50.165.231/xmlquiz/default.aspx?sessionId=0b712fd83b9f5313db5af1cea6b1004f&ip=10.50.165.231&mac=00:0e:35:bd:c9:37&ufi=0006b11184300&mgmtBaseUrl=https://10.50.165.193:4043/&clientRedirectUrl=https://10.50.165.193:444/&req=http%3A/www.google.com/ig

```
Dim ip as String
Dim sessionId as String
Dim mac as String
Dim ufi as String
Dim mgmtBaseUrl as String
Dim clientRedirectUrl as String
Dim req as String
Dim hmac as String
Dim emailAddr as String
Dim userName as String
Dim customCode as String
```

```
Sub Page_Load(src as Object, e as EventArgs)
    ip=Request.QueryString("ip")
    sessionId=Request.QueryString("sessionId")
    mac=Request.QueryString("mac")
    ufi=Request.QueryString("ufi")
    mgmtBaseUrl=Request.QueryString("mgmtBaseUrl")
    clientRedirectUrl=Request.QueryString("clientRedirectUrl")
    req=Request.QueryString("req")
    hmac=Request.QueryString("hmac")
    customCode=Request.QueryString("cc")
```

'customCode grabs the 'cc=' querystring value sent by the SonicWALL. This allows you to use the same 'page (e.g. this page) for the "Session Expiration" (?cc=2), "Idle Timeout" (?cc=3) and "Max Sessions" (?cc=4) page.

```
If customCode <> "" Then
    Select Case customCode
        Case "2"
            LHMResult.Text="<br><H3><font color=""red"">Your LHM session has expired. You may try to initiate a new session.</font></H3>"
        Case "3"
            LHMResult.Text="<br><H3><font color=""red"">You have exceeded your idle timeout. Please log back in.</font></H3>"
        Case "4"
            LHMResult.Text="<br><H3><font color=""red"">The maximum number of sessions has been reached. Please try again later.</font></H3>"
    End Select
End If
```

'Note - the routine below for handling the hmac requires the use of the SonicSSL.dll and libeay.dll libraries.

'The DLL must be copied to the IIS server, and the SonicSSL dll must be registered with "regsvr32 sonicssl.dll"

If hmac <> "" Then

'SonicWALL URL Encode routine is different from Microsoft - this is the SonicWALL method

```
req=Replace(req,"%","%25")
req=Replace(req,"%","%3A")
req=Replace(req,"%","%20")
req=Replace(req,"%?","%3F")
req=Replace(req,"%+","%2B")
req=Replace(req,"%&","%26")
req=Replace(req,"%=","%3D")
```

```
Dim strHmacText as String
Dim objCrypto as Object
Dim strHmacGenerated
Dim loginError as String
```

'Initialize the Crypto object

```
objCrypto = Server.CreateObject("SonicSSL.Crypto")
```

'The text to be encoded

```
strHmacText = sessionId & ip & mac & ufi & mgmtBaseUrl & clientRedirectUrl & req
```

'Calculate the hash with a key strHmac, the return value is a string converted from the output sha1 binary.

'The hash algorithm (MD5 or SHA1) is configured in myvars and in the Extern Guest Auth config on the SonicWALL

```
If hmacType = "MD5" Then
    strHmacGenerated = objCrypto.hmac_md5(strHmacText, strHmac)
```



Tech Note

```
Else
    strHmacGenerated = objCrypto.hmac_sha1(strHmacText, strHmac)
End If

If strHmacGenerated <> hmac Then
    Dim hmacFail as String
    hmacFail = "<font color=""red"">The HMAC failed validation. Please notify an attendant.</font><br><br>"
    hmacFail+="<font color=""9CBACE"">Received HMAC: " & hmac & "<br>Calculated HMAC: " & strHmacGenerated & "<br>"
    hmacFail+="Make sure the digest functions on the SonicWALL and LHM server match.<br>"
    hmacFail+="Also make sure the shared secret on the SonicWALL and myvars match.</font>"
    catchError.Text=hmacFail
End If

End If

End Sub

'When the submit button is clicked, pass the variables we need and load the quiz
Sub btnSubmit_Click(Sender As Object, E As EventArgs)

    Context.Items.Add("req", req)
    Context.Items.Add("sessionId", sessionId)
    Context.Items.Add("emailAddr", clientEmail.Text)
    Context.Items.Add("userName", clientName.Text)
    Context.Items.Add("mgmtBaseUrl",mgmtBaseUrl)
    Server.Transfer("quiz.aspx",true)

End Sub

</script>

<STYLE>
body {
    font-size: 10pt;
    font-family: verdana,helvetica,arial,sans-serif;
    color:#000000;
    background-color:#9CBACE;
}

tr.heading {
    background-color:#006699;
}

.button {
    border: 1px solid #000000;
    background-color: #ffffff;
}
</STYLE>

<HTML>
<HEAD>
<TITLE>LHM Quiz Script</TITLE>
</HEAD>

<BODY>
<form id="frmValidator" runat="server">

<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;  </td>
    </tr>
    <tr class="heading">
        <td width="50%" valign="center"><font color="white"><b>LHM Quiz Authorization</b></font></td>
        <td align="center"></center></td>
        <td width="50%" align="right" valign="center"><font color="white"><b>Powered by SonicWALL LHM</b>&nbsp;  </font></td>
    </tr>
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;  </td>
    </tr>
</table>
<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr>
        <td width="30%"><br>Enter your full name:</td>
        <td width="20%"><asp:TextBox id="clientName" runat="server" /></td>
        <td><asp:RequiredFieldValidator id="valTxtName" ControlToValidate="clientName" ErrorMessage="Please enter your name." Display="Dynamic"
runat="server" /></td>
    </tr>
</table>
```



Tech Note

```
<td width="30%"><br>Enter your real email address:</td>
<td width="20%"><asp:TextBox id="clientEmail" runat="server" /></td>
<td ><asp:RegularExpressionValidator id="fromEmail" runat="server" ControlToValidate="clientEmail" ValidationExpression=".*@.\..*"
ErrorMessage="Please enter a valid email address." Display="Dynamic" />
</asp:RegularExpressionValidator>
<asp:RequiredFieldValidator id="fromRequired" runat="server" ControlToValidate="clientEmail" ErrorMessage="Please enter your email address."
Display="Dynamic" />
</asp:RequiredFieldValidator>
</td>
</tr>
<tr>
<td></td>
<td><asp:button id="btnSubmit" class="button" text=" Submit " onClick="btnSubmit_Click" runat="server" /><br></td>
</tr>
<tr class="heading">
<td colspan=3 align="left"><font color="white"><b>Welcome Quiztaker <%= ip%></b></font></td>
</tr>
</table>
<table width="70%" border="0" cellpadding="2" cellspacing="0">
<tr>
<td>
<br>
<br>You have been redirected here by Lightweight Hotspot Messaging.
This environment has been setup to demonstrate the flexibility of LHM, including
support for both wired and wireless clients, and also the ability for LHM to use
more than just username and password authentication for providing access.<br><br>
The page that you are about to continue on to is a <%= quizName %> written in ASP.net.
A passing score of <%= passingScore%>% will serve as the authentication for LHM, and will grant
you network access. You must pass the test to continue, and will be prompted to retake
the entire quiz if you you do not pass. <br><br>
When you are done, the completed test will be emailed to you at the address you
specify above.<br><br>
So it's not just a good way to prove your understanding of some
key SonicOS concepts, but also a practical example of the versatility of LHM.
</td>
</tr>
<tr>
<td><asp:Label id=LHMResult runat="server" /></td>
</tr>
<tr>
<td colspan=2><asp:Label id=catchError runat="server" /></td>
</tr>
</table>
</form>
</BODY>
</HTML>
```

Tech Note

quiz.aspx

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Xml" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Security" %>
<%@ Import namespace="System.Security.Cryptography.X509Certificates"%>
<%@ Import Namespace="System.Web" %>
<%@ Import Namespace="System.Web.Mail" %>

<!-- Original quiz code from www.codeproject.com -->

<!-- #INCLUDE file="myvars.aspx" -->

<script language="VB" runat="server">

'This class allows SSL certs signed by unknown CAs to be accepted.
'This is necessary for the POST to the SonicWALL authorizing the LHM session.
Public Class acceptAllCerts
    Implements System.Net.ICertificatePolicy
    Public Function CheckValidationResult(ByVal srvPoint As ServicePoint, _
        ByVal cert As X509Certificate, ByVal request As WebRequest, ByVal problem As Integer) _
        As Boolean Implements ICertificatePolicy.CheckValidationResult
        Return True
    End Function
End Class

'Set the path to the XML quiz data
Dim strXmlFilePath as String = Server.MapPath(quizFile)

'Setup our variables
Dim emailAddr as String
Dim userName as String
Dim req as String
Dim sessionId as String
Dim mgmtBaseUrl as String
Dim xDoc as XmlDocument = New XmlDocument()
Dim intTotalQuestion as Integer
Dim intQuestionNo as Integer = 1
Dim intScore as Integer = 0
Dim arrAnswerHistory as new ArrayList()
Dim arrRightOrWrong as new ArrayList()
Dim arrCorrect as new ArrayList()

Sub Page_Load(src as Object, e as EventArgs)

    LHMResult.Text=""
    catchError.Text=""

    'Grab context items set in default.aspx
    emailAddr = Context.Items("emailAddr")
    userName = Context.Items("userName")
    req = Context.Items("req")
    sessionId = Context.Items("sessionId")
    mgmtBaseUrl = Context.Items("mgmtBaseUrl")

    'Use the override class in myvars.aspx to accept untrusted certificates from the SonicWALL
    'This is necessary for the POST to the SonicWALL authorizing the LHM session.
    System.Net.ServicePointManager.CertificatePolicy = New acceptAllCerts

    'Load xml data
    xDoc.Load(strXmlFilePath)

    'Start a new quiz?
    If Not Page.IsPostBack Then

        'Yes. Count total question
        intTotalQuestion = xDoc.SelectNodes("/quiz/mchoice").Count

        'Record start time
        ViewState("StartTime") = DateTime.Now

        ShowQuestion(intQuestionNo)

    End If

End Sub
```



Tech Note

End Sub

Sub btnSubmit_Click(src as Object, e as EventArgs)

```
'Retrieve variables from ViewState
intTotalQuestion = ViewState("TotalQuestion")
intQuestionNo = ViewState("QuestionNo")
intScore = ViewState("Score")
arrAnswerHistory = ViewState("AnswerHistory")
arrRightOrWrong = ViewState("RightOrWrong")
arrCorrect = ViewState("AnswerList")
req = ViewState("origReq")
userName = ViewState("origUserName")
emailAddr = ViewState("origEmailAddr")
mgmtBaseUrl = ViewState("mgmtUrl")
sessionId = ViewState("sessID")
```

```
'Correct answer?
If rblAnswer.SelectedItem.Value = ViewState("CorrectAnswer") Then
    intScore += 1
    arrRightOrWrong.Add(0)
Else
    arrRightOrWrong.Add(rblAnswer.SelectedItem.Value)
End If
```

```
'Remember all selected answers
arrAnswerHistory.Add(rblAnswer.SelectedItem.Value)
arrCorrect.Add(ViewState("CorrectAnswer"))
```

```
'End of quiz?
If intQuestionNo=intTotalQuestion Then
```

```
    'Yes. Show the result.
    QuizScreen.Visible = False
    ResultScreen.Visible = True
```

```
    'Render result screen
    ShowResult()
```

Else

```
    'Not yet. Show another question.
    QuizScreen.Visible = True
    ResultScreen.Visible = False
    intQuestionNo += 1
```

```
    'Render next question
    ShowQuestion(intQuestionNo)
```

End If

End Sub

Sub ShowQuestion(intQuestionNo as Integer)

```
Dim xNodeList as XmlNodeList
Dim xNodeAttr as Object
Dim strXPath as String
Dim i as Integer
Dim tsTimeSpent as TimeSpan
```

```
strXPath = "/quiz/mchoice[" & intQuestionNo.ToString() & "]"
```

```
'Extract question
lblQuestion.Text = intQuestionNo.ToString() & ". " & xDoc.SelectSingleNode(strXPath & "/question").InnerXml
```

```
'Extract answers
xNodeList = xDoc.SelectNodes(strXPath & "/answer")
```

```
'Clear previous listitems
rblAnswer.Items.Clear
```

```
For i = 0 to xNodeList.Count-1
```

```
    'Add item to radiobuttonlist
    rblAnswer.Items.Add(new ListItem(xNodeList.Item(i).InnerText, i+1))
```

```
    'Extract correct answer
    xNodeAttr = xNodeList.Item(i).Attributes.ItemOf("correct")
    If not xNodeAttr is Nothing Then
        If xNodeAttr.Value = "yes" Then
```



Tech Note

```
ViewState("CorrectAnswer") = i+1
End If
End If
Next

'Output Total Question and passing score
lblTotalQuestion.Text = intTotalQuestion
lblPassingScore.Text = passingScore

'Output Time Spent
tsTimeSpent = DateTime.Now.Subtract(ViewState("StartTime"))
lblTimeSpent.Text = tsTimeSpent.Minutes.ToString() & ":" & tsTimeSpent.Seconds.ToString()

'Store data to viewstate
ViewState("TotalQuestion") = intTotalQuestion
ViewState("Score") = intScore
ViewState("QuestionNo") = intQuestionNo
ViewState("AnswerHistory") = arrAnswerHistory
ViewState("RightOrWrong") = arrRightOrWrong
ViewState("AnswerList") = arrCorrect
ViewState("origReq")=req
ViewState("origUserName")=userName
ViewState("origEmailAddr")=emailAddr
ViewState("mgmtUrl")=mgmtBaseUrl
ViewState("sessID")=sessionID

End Sub

Sub ShowResult()
    Dim strResult as String
    Dim intCompetency as Integer
    Dim i as Integer
    Dim strXPath as String
    Dim tsTimeSpent as TimeSpan

    tsTimeSpent = DateTime.Now.Subtract(ViewState("StartTime"))

    strResult = "<center>"

    if passingScore <= Int((intScore/intTotalQuestion*100).ToString())
        strResult += "<h2><font color=""green"">You Passed!</h3></font>"
    else
        strResult += "<h2><font color=""red"">You Failed!</h3><b>Please review the answers and retake the test.</b><br></font>"
    End If

    strResult += "User Name: " & userName & "<br>"
    strResult += "Elapsed Time: " & tsTimeSpent.Minutes.ToString() & ":" & tsTimeSpent.Seconds.ToString() & "<br>"
    strResult += "Correct Answers: " & intScore.ToString() & " out of " & intTotalQuestion.ToString() & "<br>"
    strResult += "Your Percentage: " & Int((intScore/intTotalQuestion*100).ToString()) & "%<br>"
    strResult += "Required Percentage: " & passingScore.ToString() & "%<br>"
    strResult += "</center>"

    strResult += "<h3>Quiz Results</h3>"
    For i = 1 to intTotalQuestion
        strXPath = "/quiz/mchoice[" & i.ToString() & "]"
        strResult += "<b>" & i.ToString() & ". " & xDoc.SelectNodes(strXPath & "/question").Item(0).InnerXml & "</b><br>"
        If arrRightOrWrong.Item(i-1)=0 Then
            strResult += "<img src = ""checkMark.gif""><font color=""green"">&nbsp;"
            strResult += "<b>You answered:</b>" & xDoc.SelectNodes(strXPath & "/answer[" & arrAnswerHistory.Item(i-1).ToString() & "]"").Item(0).InnerXml & "</font><br><br>"
        Else
            strResult += "<img src = ""Block.gif""><font color=""red"">&nbsp;"
            strResult += "<b>You answered:</b>" & xDoc.SelectNodes(strXPath & "/answer[" & arrAnswerHistory.Item(i-1).ToString() & "]"").Item(0).InnerXml & "<br>"
            strResult += "The correct answer is: " & xDoc.SelectNodes(strXPath & "/answer[" & arrCorrect.Item(i-1).ToString() & "]"").Item(0).InnerXml & "</font><br><br>"
        End If
    Next

    'Setup the common Mail settings
    Dim objMail As MailMessage
    objMail = New MailMessage()
    objMail.From = quizFrom
    objMail.Body = strResult
    objMail.BodyFormat = MailFormat.Html

    'Path to the attachments for the Check and X images - update these in myvars.aspx
    objMail.Attachments.Add(New MailAttachment(imagePath & "block.gif"))
```



Tech Note

```
objMail.Attachments.Add(New MailAttachment(imagePath & "checkMark.gif"))
```

```
'Address of the SMTP server - can be localhost if SMTP is running on IIS - in myvars.aspx  
SmtpMail.SmtpServer = smtpServer
```

```
'Determine pass/fail  
If passingScore <= Int(intScore/intTotalQuestion*100).ToString()
```

```
    'Mail the passing test result to the test-taker  
    'Be sure to update the mail fields in myvars.aspx  
    objMail.To =emailAddr  
    objMail.Subject = quizName & " Results for " & emailAddr  
  
    'Send the mail  
    SmtpMail.Send(objMail)  
    strResult += "Your test is being emailed to you at " & emailAddr
```

```
    'Send the session Auth message to LHM  
    postLHM()
```

```
else  
    'Mail failing test results to the instructor  
    objMail.To =quizTo  
    objMail.Subject = "Failing " & quizName & " Test Results for " & emailAddr
```

```
    'Send the mail  
    SmtpMail.Send(objMail)  
    strResult += "<a href=""quiz.aspx"">Click here to retake the quiz</a>"
```

```
End If
```

```
'Write it  
lblResult.Text = strResult
```

```
End Sub
```

```
Sub postLHM()
```

```
    'The LHM cgi on the SonicWALL - this does not change  
    Dim loginCgi as String = "externalGuestLogin.cgi"
```

```
    'Assemble the data to post back to the SonicWALL to authorize the LHM session  
    Dim loginParams as String = "sessId=" & sessionId & "&userName=" & Server.URLEncode(userName) & "&sessionLifetime=" & sessTimer & "&idleTimeout=" &  
idleTimer
```

```
    'Combine mgmtBaseUrl from the original redirect with the login cgi  
    Dim postToSNWL as String = mgmtBaseUrl & loginCgi
```

```
    'Convert the loginParams to a well behaved byte array  
    Dim byteArray As Byte() = Encoding.UTF8.GetBytes(loginParams)
```

```
Try
```

```
    'Let the user know that we are setting up the session, just in case it takes more than a second  
    LHMRresult.Text = "Authorizing session. Please wait."
```

```
    'Create the webrequest to the SonicWALL  
    Dim toSNWL as WebRequest = WebRequest.Create(postToSNWL)
```

```
    'Calculate the length of the byte array  
    toSNWL.ContentLength = byteArray.Length
```

```
    'Set the method for the webrequest to POST  
    toSNWL.Method = "POST"
```

```
    'Set the content type  
    toSNWL.ContentType = "application/x-www-form-urlencoded"
```

```
    'Open the request stream  
    Dim dataStream As Stream = toSNWL.GetRequestStream()
```

```
    'Write the byte array to the request stream  
    dataStream.Write(byteArray, 0, byteArray.Length)
```

```
    'Close the Stream object  
    dataStream.Close()
```

```
    'Get the response  
    Dim snwlReply As WebResponse = toSNWL.GetResponse()
```



Tech Note

```
'Display the status - looking for 200 = OK.
'Response.Write(CType(snwIReply, HttpWebResponse).StatusCode)

'Grab the response and stuff it into an xml doc for possible review
Dim snwIResponse as XmlDocument = New XmlDocument()
snwIResponse.Load(snwIReply.GetResponseStream())

'Set the xPath to the SNWL reply, and get the response
Dim codePath as String = "SonicWALLAccessGatewayParam/AuthenticationReply/ResponseCode"

'Response code 50 - Login Succeeded

If snwIResponse.SelectSingleNode(codePath).InnerXml = "50"

    'Do we want to provide a logout popup window?
    If logoutPopup = "1" Then
        'Popup hack using Javascript for logout window
        Dim sb As New System.Text.StringBuilder()
        sb.Append("<script language='javascript'>")
        sb.Append("window.open('logout.aspx?sessId=")
        sb.Append(Server.URLEncode(CStr(sessionId)))
        sb.Append("&mgmtBaseUrl=")
        sb.Append(Server.URLEncode(CStr(mgmtBaseUrl)))
        sb.Append("&sessTimer=")
        sb.Append(Server.URLEncode(CStr(sessTimer)))
        sb.Append("'", 'logOut', 'toolbar=no,')
        sb.Append("addressbar=no,menubar=no,")
        sb.Append("width=400,height=250');")
        sb.Append("<")
        sb.Append("/")
        sb.Append("<script>")
        RegisterStartupScript("stp", sb.ToString)
    End If

    LHMResult.Text = "<br><b><font color=""green"">Session authorized:</font></b> You may now go to the URL you originally requested: <a
target=""_blank"" href=""& req & "">" & req & "</a>"

'Response code 51 - Session Limit Exceeded
Elseif snwIResponse.SelectSingleNode(codePath).InnerXml = "51"
    LHMResult.Text = "<br><b><font color=""red"">Session Limit Reached:</font></b> The maximum number of guest session has been
reached. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 100 - Login Failed.
Elseif snwIResponse.SelectSingleNode(codePath).InnerXml = "100"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> Your session cannot be created at this time. Sorry for
the inconvenience. Please close and relaunch your browser to try again."

'Response code 251 - Bad HMAC.
Elseif snwIResponse.SelectSingleNode(codePath).InnerXml = "251"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed message
authentication. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 253 - Invalid SessionID.
Elseif snwIResponse.SelectSingleNode(codePath).InnerXml = "253"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed to match a known
session identity. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 254 - Invalid CGI.
Elseif snwIResponse.SelectSingleNode(codePath).InnerXml = "254"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization was missing an essential
parameter. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 255 - Internal Error.
Elseif snwIResponse.SelectSingleNode(codePath).InnerXml = "255"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed due to an
unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again."

End If

'Close the streams
dataStream.Close()
snwIReply.Close()

'If there is some asp.net error trying to talk to the SonicWALL, print it
'in the same color as the background, but still show the quiz results.
Catch ex as Exception
    catchError.Text = "<font color=""9CBACE"">" & ex.ToString & "</font>"
```



Tech Note

LHMResult.Text = "
Session creation failed: The request for authorization failed due to an unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again. If the problem persists, please notify an attendant."
End Try

End Sub

```
</script>
<html>
<head>
<title><%= quizName %> </title>
</head>
<style>
body {
font-size: 10pt;
font-family: verdana,helvetica,arial,sans-serif;
color:#000000;
background-color:#9CBACE;
}

tr.heading {
background-color:#006699;
}

.button {
border: 1px solid #000000;
background-color: #ffffff;
}
</style>

<HTML>
<HEAD>
<TITLE>LHM Quiz Script</TITLE>
</HEAD>

<body>
<span id="QuizScreen" runat="server">
<form runat="server">
<table width="100%" border="0" cellpadding="2" cellspacing="0">
<tr class="heading">
<td colspan=3 align="center"><font color="white">&nbsp;</td>
</tr>
<tr class="heading">
<td width="50%" valign="center"><font color="white"><b><%= quizName %> - <%= userName%></b></font></td>
<td align="center"></center></td>
<td width="50%" align="right" valign="center"><font color="white"><b>This quiz has <asp:label id="lblTotalQuestion" runat="server" />
questions</b></font></td>
</tr>
<tr class="heading">
<td colspan=3 align="center"><font color="white">&nbsp;</td>
</tr>
</table>
<table width="100%" border="0" cellpadding="2" cellspacing="0">
<tr>
<td colspan="2">
<b><asp:label id="lblQuestion" runat="server" /></b><br>
<asp:radiobuttonlist id="rblAnswer" RepeatDirection="vertical" TextAlign="right" RepeatLayout="table" runat="server" /><br>
<asp:button id="btnSubmit" class="button" text=" Submit " onClick="btnSubmit_Click" runat="server" />
<asp:requiredfieldvalidator ControlToValidate="rblAnswer" ErrorMessage="Please select an answer" runat="server" />
</td>
</tr>
<tr class="heading">
<td width="70%"><font color="white"><b>Score required to pass <asp:label id="lblPassingScore" runat="server" />%</b></font></td>
<td width="30%" align="right"><font color="white"><b>Time spent <asp:label id="lblTimeSpent" runat="server" /></b></font></td>
</tr>
</table>
</form>
</span>

<span id="ResultScreen" runat="server"> <asp:label id="lblResult" runat="server" /> <br>
<asp:Label id=LHMResult runat="server" />
<asp:Label id=catchError runat="server" />
</span>

</body>
</html>
```



Tech Note

PayPal Script

[myvars.aspx](#)

[default.aspx](#)

[pdt.aspx](#)

myvars.aspx

```
<script language="VB" runat="server">

'Set the logoutPopup window flag - 0 = no popup, 1 = popup
'The use of the logoutPopup in this script is discouraged because the login event is exclusive.
Dim logoutPopup as String = "0"

'Set the debug flag (0 = off, 1 = on)
Dim debugFlag as String = "0"

'Set the path and file for the PDT responder script - this should be the same path as the LHM settings
'configured on the SonicWALL "External Web Server Settings" page, but pointing to the PDT handler script.
'Refer to http://www.paypal.com/cgi-bin/webscr?cmd=p/xcl/rec/pdt-techview-outside for information on PDT
Dim pdtPath as String = "http://10.50.165.2/lhm/paypal/pdt.aspx"

'Set the path the PayPal processing CGI. Use the sandbox (https://developer.paypal.com) and (https://www.sandbox.paypal.com) for testing
'Using the sandbox requires a developer network account and login.
Dim paypalCGI as String = "https://www.sandbox.paypal.com/cgi-bin/webscr"
'Dim paypalCGI as String = "https://www.paypal.com/cgi-bin/webscr"

'Set the email address of the paypal merchant account to which payment will be made
'The following is a valid sandbox account, but requires authentication by the parent (real) account.
'You must replace this with you own (real or sandbox account) for use.
Dim myBusiness as String = "lhmdemo@sonicwall.com"

'Set this to token from PayPal account. It must be your actual, valid token.
'Refer to http://paypaltech.com/PDTGen/PDTtokenhelp.htm for information on the identity token
'The following is a valid sandbox token, but requires authentication by the parent (real) account.
'You must replace this with you own (real or sandbox token) for use.
Dim token as String = "ucistq6vmKGWPxwJbrTJFDhFq889RxYt_6Mkz_3viraSzjiQJ5iPYCZ5Mdq"

'Set the names for the purchase item options (e.g. 1 hour Access, 3 hours access, etc.)
Dim itemName1 as String = "1 Hour Secure Internet Access"
Dim itemName2 as String = "24 Hours Secure Internet Access"

'Set the paypal querystring number for purchase item options (e.g. 1hour, 60mins, itemone, etc.)
Dim itemNumber1 as String = "1hour"
Dim itemNumber2 as String = "24hour"

'Set the purchase item options session and idle timers (timers use the same value since we do not want sessions idling out)
Dim itemTimer1 as String = "3600" 'One hour, in minutes
Dim itemTimer2 as String = "86400" '24 hours

'Set the costs in dollars for purchase item options (e.g. one penny = 0.01, one dollar = 1.00, etc.)
Dim itemAmount1 as String = "0.01"
Dim itemAmount2 as String = "0.02"

'Set the button names and descriptions for purchase item options
Dim itemButton1 as String = "1 Hour Access - $0.01"
Dim itemButton2 as String = "24 Hours Access - $0.02"

'Set the secret for use with optional HMAC auth, as configured in the Extern Guest Auth config on the SonicWALL
Dim strHmac as String = "password"

'Set the digest method for the HMAC, either MD5 or SHA1
Dim hmacType as String = "MD5"
'Dim hmacType as String = "SHA1"

'Set the logo image to use
Dim logo as String = "sonicwall.gif"
'-----End of Configurable Settings-----
</script>
```



Tech Note

default.aspx

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Xml" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Security" %>
<%@ Import namespace="System.Security.Cryptography.X509Certificates"%>
```

```
<!-- #INCLUDE file="myvars.aspx" -->
```

```
<script language="VB" runat="server">
```

'This class allows SSL certs signed by unknown CAs to be accepted.

'This is necessary for the POST to the SonicWALL authorizing the LHM session.

Public Class acceptAllCerts

Implements System.Net.ICertificatePolicy

Public Function CheckValidationResult(ByVal srvPoint As ServicePoint, _

ByVal cert As X509Certificate, ByVal request As WebRequest, ByVal problem As Integer) _

As Boolean Implements ICertificatePolicy.CheckValidationResult

Return True

End Function

End Class

'Note: For PayPal authorization to work, it is necessary to set up the PayPal sites (www.paypal.com, www.paypalobjects.com, and www.sandbox.paypal.com) as a bypass network on WGS. This is so that WGS/LHM users can access PayPal directly to complete the payment transactions. This list currently includes the following addresses: [64.4.241.32, 64.4.241.33, 216.113.188.32, 216.113.188.35, 216.113.188.66, 216.113.188.67], [216.113.188.25, 64.4.241.62, 216.113.188.9] and [66.135.197.160].

'Sample LHM redirect querystring:

'http://127.0.0.1/lhm/paypal/default.aspx?sessionId=0b712fd83b9f5313db5af1cea6b1004f&ip=10.50.165.231&mac=00:0e:35:bd:c9:37&ufi=0006b11184300&mgmtBaseUrl=ht
tps://10.50.165.193:4043/&clientRedirectUrl=https://10.50.165.193:444/&req=http%3A/www.google.com/ig

Dim ip as String

Dim sessionId as String

Dim mac as String

Dim ufi as String

Dim mgmtBaseUrl as String

Dim clientRedirectUrl as String

Dim req as String

Dim hmac as String

Dim customCode as String

Sub Page_Load(src as Object, e as EventArgs)

ip=Request.QueryString("ip")

sessionId=Request.QueryString("sessionId")

mac=Request.QueryString("mac")

ufi=Request.QueryString("ufi")

mgmtBaseUrl=Request.QueryString("mgmtBaseUrl")

clientRedirectUrl=Request.QueryString("clientRedirectUrl")

req=Request.QueryString("req")

hmac=Request.QueryString("hmac")

customCode=Request.QueryString("cc")

'customCode grabs the "cc=" querystring value sent by the SonicWALL. This allows you to use the same

'page (e.g. this page) for the "Session Expiration" (?cc=2), "Idle Timeout" (?cc=3) and "Max Sessions" (?cc=4) page.

If customCode <> "" Then

Select Case customCode

Case "2"

LHMResult.Text="
<H3>Your LHM session has expired. You may try to initiate a new session.</H3>"

Case "3"

LHMResult.Text="
<H3>You have exceeded your idle timeout. Please log back in.</H3>"

Case "4"

LHMResult.Text="
<H3>The maximum number of sessions has been reached. Please try again later.</H3>"

End Select

End If

'Set the button Text for the two buttons with the variable configured in myvars

btnBuyNow1.Text=itemButton1

btnBuyNow2.Text=itemButton2

'Use the override class in myvars.aspx to accept untrusted certificates from the SonicWALL

'This is necessary for the POST to the SonicWALL authorizing the LHM session.

System.Net.ServicePointManager.CertificatePolicy = New acceptAllCerts



Tech Note

'Note - the routine below for handling the hmac requires the use of the SonicSSL.dll and libeay.dll libraries.
'The DLL must be copied to the IIS server, and the SonicSSL dll must be registered with "regsvr32 sonicssl.dll"
If hmac <> "" Then

'SonicWALL URL Encode routine is different from Microsoft - this is the SonicWALL method

```
req=Replace(req,"%","%25")
req=Replace(req,"%","%3A")
req=Replace(req,"%","%20")
req=Replace(req,"%","%3F")
req=Replace(req,"%","%2B")
req=Replace(req,"%","%26")
req=Replace(req,"%","%3D")
```

```
Dim strHmacText as String
Dim objCrypto as Object
Dim strHmacGenerated
Dim loginError as String
```

```
'Initialize the Crypto object
objCrypto = Server.CreateObject("SonicSSL.Crypto")
```

```
'The text to be encoded
strHmacText = sessionId & ip & mac & ufi & mgmtBaseUrl & clientRedirectUrl & req
```

```
'Calculate the hash with a key strHmac, the return value is a string converted from the output sha1 binary.  
'The hash algorithm (MD5 or SHA1) is configured in myvars and in the Extern Guest Auth config on the SonicWALL
```

```
If hmacType = "MD5" Then
    strHmacGenerated = objCrypto.hmac_md5(strHmacText, strHmac)
Else
    strHmacGenerated = objCrypto.hmac_sha1(strHmacText, strHmac)
End If
```

```
If strHmacGenerated <> hmac Then
    Dim hmacFail as String
    hmacFail = "<font color=""red"">The HMAC failed validation. Please notify an attendant.</font><br><br>"
    hmacFail+="<font color=""9CBACE"">Received HMAC: " & hmac & "<br>Calculated HMAC: " & strHmacGenerated & "<br>"
    hmacFail+="Make sure the digest functions on the SonicWALL and LHM server match.<br>"
    hmacFail+="Also make sure the shared secret on the SonicWALL and myvars match</font>"
    catchError.Text=hmacFail
End If
```

End If

End Sub

Sub btnBuyNow_Click(Sender As Object, E As EventArgs)

'sample redirect generated by this routine:

'https://www.paypal.com/cgi-bin/webscr?cmd=_xclick&business=jelevy@sonicwall.com&item_name=24%20Hour%20Secure%20Internet%20Access&item_number=24hour&amount=0.02¤cy_code=USD&lc=US&bn=PP-BuyNowBF&no_note=1&no_shipping=1&cancel_return=http://127.0.0.1/lhm/paypal/default.aspx&return=http://www.moosifer.com/pdt.aspx

'sample redirect from the paypal server back the LHM server on transaction completion (modified).

'http://127.0.0.1/lhm/paypal/pdt.aspx?tx=4PG453F7LS133715V&st=Completed&amt=0.02&cc=USD&cm=&sig=EZhZtJygi7RTXuJt4SEhVBRI%2bJwLaC9z9kRLsrsXk4gQKnzvl5vjGy0vdhKPXAyVhbh%2bwBxWvn2cieEQDJ9P6R9qqjuKnzvl5vjGy0vdhKPXAyJ3GtOq5Jd3%2fvTY3s7FrRcKdKnzvl5vjGy0vdhKPXAyvyEKNxY3d

```
Dim str, itemName, itemNumber, itemAmount As String
Dim sb As New StringBuilder()
```

'Determine which button was pressed, and set item attributes appropriately

```
Select Case Sender.Text
    Case itemButton1
        itemName = itemName1
        itemNumber = itemNumber1
        itemAmount = itemAmount1
    Case itemButton2
        itemName = itemName2
        itemNumber = itemNumber2
        itemAmount = itemAmount2
End Select
```

'The paypal CGI URL - You can select either the real CGI or the sandbox CGI in myvars

sb.Append(paypalCGI & "?")

'The cmd passed to PayPal - do not change!

sb.Append("cmd=_xclick")

'The email address of the paypal merchant receiving payment. Replace in myvars with your paypal email address.

sb.Append("&business=" & myBusiness)

'The name of the item being purchased. This is the first item option (e.g. 1 hour). Set in myvars



Tech Note

```
sb.Append("&item_name=" & itemName)
'The optional item id
sb.Append("&item_number=" & itemNumber)
'The price being charged for the item (access)
sb.Append("&amount=" & itemAmount)
'The currency
sb.Append("&currency_code=USD")
'The country
sb.Append("&lc=US")
'The banana nullifier
sb.Append("&bn=PP-BuyNowBF")
'Disables the note option on the transaction
sb.Append("&no_note=1")
'Disables the shipping option on the transaction
sb.Append("&no_shipping=1")
'Build the path to return the client to (the LHM server address) on a cancelled transaction
sb.Append("&cancel_return=http://" & Request.ServerVariables("SERVER_NAME") & Request.ServerVariables("URL"))
'The return (success page) path to return the buyer to after the transaction. This is the PDT receiver/processor page.
sb.Append("&return=" & pdtPath)
'The LHM sessionID - append this so that it can be returned to us later by the PDT transaction - do not change!
sb.Append("&custom=" & sessionId & Server.URLEncode(mgmtBaseUrl))
'Optional notify_url that paypal will asynchronously send IPN confirmation to. Not used since it's not real-time.
'sb.Append("&notify_url=http://www.moosifer.com/ipn.aspx")
str = sb.ToString
Response.Redirect(str)
```

End Sub

</script>

<STYLE>

```
body {
    font-size: 10pt;
    font-family: verdana,helvetica,arial,sans-serif;
    color:#000000;
    background-color:#9CBACE;
}
```

```
tr.heading {
    background-color:#006699;
}
```

```
.button {
    border: 1px solid #000000;
    background-color: #ffffff;
}
```

</STYLE>

<HTML>

<HEAD>

<TITLE>LHM PayPal Script</TITLE>

</HEAD>

<BODY>

<form id="frmValidator" runat="server">

```
<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;</td>
    </tr>
    <tr class="heading">
        <td width="50%" valign="center"><font color="white"><b>LHM Access with PayPal Buy Now</b></font></td>
        <td align="center"></center></td>
        <td width="50%" align="right" valign="center"><font color="white"><b>Powered by SonicWALL LHM</b>&nbsp;</font></td>
    </tr>
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;</td>
    </tr>
</table>
```

```
<table width="100%" border="0" cellpadding="2" cellspacing="0">
```

```
    <tr>
        <td colspan=3><br></td>
    </tr>
    <tr>
```

<td colspan=3 align="left">Purchase Secure Internet Access through SonicWALL's LHM and PayPal's Buy Now feature.



Tech Note

The two Buy Now buttons below will send you to PayPal's website where you can use your PayPal account to pay \$<%= itemAmount1 %> for <%= itemName1 %>, or \$<%= itemAmount2 %> for <%= itemName2 %>.

PayPal will then redirect you to this site to initiate the Payment Data Transfer (PDT) exchange. The PDT exchange begins with the LHM server posting a paypal constructed querystring back to paypal. The response to the post will then be parsed by the LHM server to determine if the PayPal transaction was successful. Once all data are exchanged and verified, LHM will authorize access on the SonicWALL for the period of time purchased.

The clock for access will start immediately upon successful session authorization, and can be used on the local SonicWALL appliance by the client (as tracked by IP and MAC address) so long as session time remains. The idle timeout will effectively be disabled by setting the idle timer to the same value as the session timer.

Please select "<%= itemName1 %>" or "<%= itemName2 %>" below. You will be redirected to the PayPal site, and will be returned to this site on transaction completion.

</td>

</tr>

<tr class="heading">

<td colspan=3 align="center"> </td>

</tr>

<tr class="heading">

<td align="center"><asp:Button ID="btnBuyNow1" Class="button" OnClick="btnBuyNow_Click" runat="server" />
 <asp:Button ID="btnBuyNow2" Class="button" OnClick="btnBuyNow_Click" runat="server" /></td>

</tr>

<tr class="heading">

<td colspan=3 align="center"> </td>

</tr>

<tr>

<td colspan=3><asp:Label id=LHMResult runat="server" /></td>

</tr>

<tr>

<td colspan=3><asp:Label id=catchError runat="server"/></td>

</tr>

</table>

</form>

</BODY>

</HTML>



Tech Note

pdt.aspx

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Xml" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Security" %>
<%@ Import namespace="System.Security.Cryptography.X509Certificates"%>

<!-- #INCLUDE file="myvars.aspx" -->

<script language="VB" runat="server">

'This class allows SSL certs signed by unknown CAs to be accepted.
'This is necessary for the POST to the SonicWALL authorizing the LHM session.
Public Class acceptAllCerts
    Implements System.Net.ICertificatePolicy
    Public Function CheckValidationResult(ByVal srvPoint As ServicePoint, _
        ByVal cert As X509Certificate, ByVal request As WebRequest, ByVal problem As Integer) _
        As Boolean Implements ICertificatePolicy.CheckValidationResult
        Return True
    End Function
End Class

'Sample LHM redirect querystring:
'http://127.0.0.1/lhm/paypal/default.aspx?sessionId=0b712fd83b9f5313db5af1cea6b1004f&ip=10.50.165.231&mac=00:0e:35:bd:c9:37&ufi=0006b11184300&mgmtBaseUrl=ht
tps://10.50.165.193:4043/&clientRedirectUrl=https://10.50.165.193:444/&req=http%3A/www.google.com/ig

Dim ip as String
Dim sessionId as String
Dim mac as String
Dim ufi as String
Dim mgmtBaseUrl as String
Dim clientRedirectUrl as String
Dim req as String
Dim sessTimer as String
Dim idleTimer as String
Dim userName as String
Dim hmac as String
Dim firstname, lastName, itemName, mcGross, mcCurrency, itemNumber, business, txn, payStatus As String

Sub Page_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

    'Use the override class to accept untrusted certificates from the SonicWALL
    'This is necessary for the POST to the SonicWALL authorizing the LHM session.
    System.Net.ServicePointManager.CertificatePolicy = New acceptAllCerts

    Dim tx, PDTvalidateQuery As String
    Dim strResponse As HttpWebResponse
    Dim temp As String
    Dim PDTArray() As String
    Dim iParts, sResults(0, 0), aParts(), sParts(), sKey, sValue, snwlCustom As String
    Dim i As Integer

    'Set tx to value of tx passed in via Querystring from PayPal
    tx = Request.QueryString("tx")

    'Set string = to the cmd value, tx and at that needs to be
    'POSTed back to PayPal to validate the PDT
    PDTvalidateQuery = "cmd=_notify-synch&tx=" & tx & "&at=" & token

    'Now we need to POST this info back to PayPal for validation of the PDT
    'Create the request back
    Dim req As HttpWebRequest = CType(WebRequest.Create(paypalCGI), HttpWebRequest)

    'Set values for the request back
    'set method
    req.Method = "POST"
    'set content type
    req.ContentType = "application/x-www-form-urlencoded"
    'set length
    req.ContentLength = PDTvalidateQuery.Length

    'Write the request back to PayPal
```



Tech Note

```
Dim stOut As StreamWriter = New StreamWriter(req.GetRequestStream(), Encoding.ASCII)
stOut.Write(PDTvalidateQuery)
stOut.Close()
```

```
Try
    strResponse = CType(req.GetResponse(), HttpWebResponse)
Catch ex As SystemException
    catchError.Text = "<font color=""9CBACE"">" & ex.ToString & "</font>"
End Try
```

'Once we write the stream back to PayPal, we need to read the response.

```
Dim IPNResponseStream As Stream = strResponse.GetResponseStream
Dim encode As Encoding = System.Text.Encoding.GetEncoding("utf-8")
Dim readStream As New StreamReader(IPNResponseStream, encode)
```

```
'Read the response in String variable "temp"
temp = readStream.ReadToEnd
```

```
    'Debug flag, set in myvars - prints the whole output from the POST reply
    If debugFlag = "1" Then
        OutputEntirePDTString(temp)
    End If
```

```
'Check to see if the 1st line of the response was "SUCCESS"
If Mid(temp, 1, 7) = "SUCCESS" Then
```

```
    'if it is SUCCESS, the code below puts the response in a nice array
    temp = Mid(temp, 9)
    sParts = Split(temp, vbCrLf)
    iParts = UBound(sParts) - 1
    ReDim sResults(iParts, 1)
```

```
For i = 0 To iParts
```

```
    aParts = Split(sParts(i), "=")
    sKey = aParts(0)
    sValue = aParts(1)
    sResults(i, 0) = sKey
    sResults(i, 1) = sValue
```

'You can add more case statements here for other returned variables

```
Try
Select Case sKey
    Case "first_name"
        firstname = Server.URLDecode(sValue)
    Case "last_name"
        lastName = Server.URLDecode(sValue)
    Case "item_name"
        itemName = Server.URLDecode(sValue)
    Case "mc_gross"
        mcGross = sValue
    Case "mc_currency"
        mcCurrency = sValue
    Case "item_number"
        itemNumber = Server.URLDecode(sValue)
    Case "business"
        business = Server.URLDecode(sValue)
    Case "txn_id"
        txn = sValue
    Case "payment_status"
        payStatus = sValue
    Case "custom"
        snwlCustom = sValue
        sessionID = snwlCustom.SubString(0,32)
        mgmtBaseUrl=(Server.URLDecode(Mid(snwlCustom,33)))
End Select
Catch ex as Exception
    catchError.Text = "<font color=""9CBACE"">" & ex.ToString & "</font>"
End Try
```

```
Next
```

```
    If payStatus = "Completed" Then
        'Transaction Succeeded - Give the Guest a receipt
        Dim receipt As String
```



Tech Note

```
receipt = "<h3>Transaction Succeeded. Thank you for selecting SonicWALL LHM.</h3><br>"
receipt += "<b>Transaction Invoice:</b><br><br>"
receipt += "Name: " & firstName & " " & lastName & "<br>"
receipt += "Description: " & itemName & "<br>"
receipt += "Amount: " & mcCurrency & " " & mcGross & "<br>"
receipt += "Paid to: " & business & "<br>"
receipt += "Transaction ID: " & txn & "<br>"
receipt += "<br><br>"

paypalResult.Text = receipt

LHMResult.Text = "Authorizing your LHM session."

'Setup the LHM session variables and call LHM Routine
'Set the session and idle timers to match the variables set in myvars
If itemNumber = itemNumber1 Then
    sessTimer=itemTimer1
    idleTimer=itemTimer1
Else
    sessTimer=itemTimer2
    idleTimer=itemTimer2
End If

userName = firstName & " " & lastName

LHM()

Else
    'The transaction itself was a success, but the payment status was not Completed.
    paypalResult.Text = "The transaction succeeded, but the payment was not completed. The session cannot be authorized at this
time."

End If

Else
    'If PDT response is not "SUCCESS"
    paypalResult.Text = "The PayPal transaction did not succeed. The returned status is: <b>" & temp & "</b>"
End If

'Close the streams
readStream.Close()
strResponse.Close()

End Sub

'This is the parser for the debug function to print the entire response to the PDT POST
Private Function OutputEntirePDTString(ByVal myPDTString As String) As String
    Dim tempString() As String = Split(myPDTString, vbCrLf)
    Dim x As Integer
    For x = 0 To tempString.GetUpperBound(0)
        Response.Write(tempString(x) & "<br>")
    Next
End Function

Sub LHM()

    'Let the user know that we are setting up the session, just in case it takes more than a second
    LHMResult.Text = "Authorizing session. Please wait."

    'The LHM cgi on the SonicWALL - this does not change
    Dim loginCgi as String = "externalGuestLogin.cgi"

    'Assemble the data to post back to the SonicWALL to authorize the LHM session
    Dim loginParams as String = "sessId=" & sessionId & "&userName=" & Server.URLEncode(userName) & "&sessionLifetime=" & sessTimer & "&idleTimeout=" &
idleTimer

    'Combine mgmtBaseUrl from the original redirect with the login cgi
    Dim postToSNWL as String = mgmtBaseUrl & loginCgi

    'Convert the loginParams to a well behaved byte array
    Dim byteArray As Byte() = Encoding.UTF8.GetBytes(loginParams)

    Try
        'Create the webrequest to the SonicWALL
        Dim toSNWL as WebRequest = WebRequest.Create(postToSNWL)

        'Calculate the length of the byte array
        toSNWL.ContentLength = byteArray.Length

        'Set the method for the webrequest to POST
```



Tech Note

```
toSNWL.Method = "POST"

'Set the content type
toSNWL.ContentType = "application/x-www-form-urlencoded"

'Open the request stream
Dim dataStream As Stream = toSNWL.GetRequestStream()

'Write the byte array to the request stream
dataStream.Write(byteArray, 0, byteArray.Length)

'Close the Stream object
dataStream.Close()

'Get the response
Dim snwlReply As WebResponse = toSNWL.GetResponse()

'Display the status - looking for 200 = OK.
'Response.Write(CType(snwlReply, HttpWebResponse).StatusCode)

'Grab the response and stuff it into an xml doc for possible review
Dim snwlResponse As XmlDocument = New XmlDocument()
snwlResponse.Load(snwlReply.GetResponseStream())

'Set the xPath to the SNWL reply, and get the response
Dim codePath as String = "SonicWALLAccessGatewayParam/AuthenticationReply/ResponseCode"

'Response.Write(snwlResponse.SelectSingleNode(codePath).InnerXml)

'Response code 50 - Login Succeeded

If snwlResponse.SelectSingleNode(codePath).InnerXml = "50"

    'Do we want to provide a logout popup window?
    If logoutPopup = "1" Then
        'Popup hack using Javascript for logout window
        Dim sb As New System.Text.StringBuilder()
        sb.Append("<script language='javascript'>")
        sb.Append("window.open('logout.aspx?sessId=")
        sb.Append(Server.URLEncode(CStr(sessionId)))
        sb.Append("&mgmtBaseUrl=")
        sb.Append(Server.URLEncode(CStr(mgmtBaseUrl)))
        sb.Append("&sessTimer=")
        sb.Append(Server.URLEncode(CStr(sessTimer)))
        sb.Append("','logOut','toolbar=no,")
        sb.Append("addressbar=no,menubar=no,")
        sb.Append("width=400,height=250');")
        sb.Append("</")
        sb.Append("/")
        sb.Append(">")
        RegisterStartupScript("stp", sb.ToString)
    End If

    LHMResult.Text = "<br><b><font color=""green"">Session authorized:</font></b> You may now begin your secure Internet access
session."

'Response code 51 - Session Limit Exceeded
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "51"
    LHMResult.Text = "<br><b><font color=""red"">Session Limit Reached:</font></b> The maximum number of guest session has been
reached. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 100 - Login Failed.
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "100"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> Your session cannot be created at this time. Sorry for
the inconvenience. Please close and relaunch your browser to try again."

'Response code 251 - Bad HMAC.
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "251"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed message
authentication. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 253 - Invalid SessionID.
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "253"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed to match a known
session identity. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 254 - Invalid CGI.
Elseif snwlResponse.SelectSingleNode(codePath).InnerXml = "254"
```



Tech Note

LHMResult.Text = "
Session creation failed: The request for authorization was missing an essential parameter. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 255 - Internal Error.

Elseif snwResponse.SelectSingleNode(codePath).InnerText = "255"

LHMResult.Text = "
Session creation failed: The request for authorization failed due to an unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again."

End If

'Close the streams
dataStream.Close()
snwReply.Close()

'If there is some asp.net error trying to talk to the SonicWALL, print it in the same color as the background.

Catch ex as Exception

catchError.Text = "" & ex.ToString & ""

LHMResult.Text = "
Session creation failed: The request for authorization failed due to an unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again. If the problem persists, please notify an attendant."

End Try

End Sub

</script>

<STYLE>

body {
font-size: 10pt;
font-family: verdana,helvetica,arial,sans-serif;
color:#000000;
background-color:#9CBACE;
}

tr.heading {
background-color:#006699;
}

.button {
border: 1px solid #000000;
background-color: #ffffff;
}

</STYLE>

<HTML>

<HEAD>

<TITLE>LHM PayPal Script</TITLE>

</HEAD>

<BODY>

<table width="100%" border="0" cellpadding="2" cellspacing="0">
tr class="heading">
td colspan=3 align="center"> </td>
</tr>
tr class="heading">
td width="50%" valign="center">LHM Access with PayPal Buy Now</td>
td align="center"></center></td>
td width="50%" align="right" valign="center">Powered by SonicWALL LHM </td>
</tr>
tr class="heading">
td colspan=3 align="center"> </td>
</tr>
</table>

<table width="100%" border="0" cellpadding="2" cellspacing="0">
tr>
td>
</td>
</tr>
tr>
td><asp:Label id=paypalResult runat="server" /></td>
</tr>
tr>
td><asp:Label id=LHMResult runat="server" /></td>
</tr>
tr>
td><asp:Label id=catchError runat="server" /></td>
</tr>
</table>



Tech Note

</BODY>
</HTML>

Tech Note

Random Script

[myvars.aspx](#)

[defaultl.aspx](#)

[generator.aspx](#)

[print.aspx](#)

myvars.aspx

```
<script language="VB" runat="server">

'Set the logoutPopup window flag - 0 = no popup, 1 = popup
'The use of the logoutPopup in this script is discouraged because the login event is exclusive.
'The login event can be made non exclusive in this script by setting useDB to 0.
Dim logoutPopup as String = "0"

'Set the use of the database for storing and checking used passcodes. 0 = do not use DB, 1 = use DB.
Dim useDB as String = "1"

'The number of characters in the randomCode
Dim randChars as Integer = 6

'Set the salt the generation of the SHA1 hash
Dim salt as String = "moosifer"

'The LHM Session Timeout is set by the passcode in this script
Dim sessTimer as String

'Set the LHM Idle Timeout
Dim idleTimer as String = "300"

'Set the secret for use with optional HMAC auth, as configured in the Extern Guest Auth config on the SonicWALL
Dim strHmac as String = "password"

'Set the digest method for the HMAC, either MD5 or SHA1
Dim hmacType as String = "MD5"
'Dim hmacType as String = "SHA1"

'Set the logo image to use
Dim logo as String = "sonicwall.gif"

'-----End of Configurable Settings-----
</script>
```

default.aspx

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Xml" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Data" %>
<%@ Import Namespace="System.Data.OleDb" %>
<%@ Import Namespace="System.Security" %>
<%@ Import Namespace="System.Security.Cryptography" %>
<%@ Import namespace="System.Security.Cryptography.X509Certificates"%>

<!-- #INCLUDE file="myvars.aspx" -->

<script language="VB" runat="server">

'This class allows SSL certs signed by unknown CAs to be accepted.
'This is necessary for the POST to the SonicWALL authorizing the LHM session.
Public Class acceptAllCerts
    Implements System.Net.ICertificatePolicy
    Public Function CheckValidationResult(ByVal srvPoint As ServicePoint, _
        ByVal cert As X509Certificate, ByVal request As WebRequest, ByVal problem As Integer) _
        As Boolean Implements ICertificatePolicy.CheckValidationResult
        Return True
    End Function
End Class
```



Tech Note

End Class

'Sample LHM redirect querystring:

'http://127.0.0.1/lhm/random/default.aspx?sessionId=0b712fd83b9f5313db5af1cea6b1004f&ip=10.50.165.231&mac=00:0e:35:bd:c9:37&ufi=0006b11184300&mgmtBaseUrl=https://10.50.165.193:4043/&clientRedirectUrl=https://10.50.165.193:444/&req=http%3A/www.google.com/ig

```
Dim ip as String
Dim sessionId as String
Dim mac as String
Dim ufi as String
Dim mgmtBaseUrl as String
Dim clientRedirectUrl as String
Dim req as String
Dim hmac as String
Dim customCode as String
```

```
Dim passCode as String
Dim grabCode as String
```

Sub Page_Load(Source as Object, E as EventArgs)

```
LHMResult.Text=""
catchError.Text=""
authResult.Text=""
```

```
ip=Request.QueryString("ip")
sessionId=Request.QueryString("sessionId")
mac=Request.QueryString("mac")
ufi=Request.QueryString("ufi")
mgmtBaseUrl=Request.QueryString("mgmtBaseUrl")
clientRedirectUrl=Request.QueryString("clientRedirectUrl")
req=Request.QueryString("req")
hmac=Request.QueryString("hmac")
customCode=Request.QueryString("cc")
```

'customCode grabs the "cc=" querystring value sent by the SonicWALL. This allows you to use the same page (e.g. this page) for the "Session Expiration" (?cc=2), "Idle Timeout" (?cc=3) and "Max Sessions" (?cc=4) page.

```
If customCode <> "" Then
    Select Case customCode
        Case "2"
            LHMResult.Text="<br><H3><font color=""red"">Your LHM session has expired. You may try to initiate a new session.</font></H3>"
        Case "3"
            LHMResult.Text="<br><H3><font color=""red"">You have exceeded your idle timeout. Please log back in.</font></H3>"
        Case "4"
            LHMResult.Text="<br><H3><font color=""red"">The maximum number of sessions has been reached. Please try again later.</font></H3>"
    End Select
End If
```

'Use the override class in myvars.aspx to accept untrusted certificates from the SonicWALL

'This is necessary for the POST to the SonicWALL authorizing the LHM session.

System.Net.ServicePointManager.CertificatePolicy = New acceptAllCerts

'Note - the routine below for handling the hmac requires the use of the SonicSSL.dll and libeay.dll libraries.

'The DLL must be copied to the IIS server, and the SonicSSL dll must be registered with "regsvr32 sonicssl.dll"

If hmac <> "" Then

'SonicWALL URL Encode routine is different from Microsoft - this is the SonicWALL method

```
req=Replace(req,"%","%25")
req=Replace(req,":","%3A")
req=Replace(req," ","%20")
req=Replace(req,"?", "%3F")
req=Replace(req,"+", "%2B")
req=Replace(req,"&","%26")
req=Replace(req,"=", "%3D")
```

```
Dim strHmacText as String
Dim objCrypto as Object
Dim strHmacGenerated
Dim loginError as String
```

```
'Initialize the Crypto object
objCrypto = Server.CreateObject("SonicSSL.Crypto")
```

```
'The text to be encoded
strHmacText = sessionId & ip & mac & ufi & mgmtBaseUrl & clientRedirectUrl & req
```

'Calculate the hash with a key strHmac, the return value is a string converted form the output

sha1 binary.



Tech Note

```
'The hash algorithm (MD5 or SHA1) is configured in myvars and in the Extern Guest Auth config on the SonicWALL
If hmacType = "MD5" Then
    strHmacGenerated = objCrypto.hmac_md5(strHmacText, strHmac)
Else
    strHmacGenerated = objCrypto.hmac_sha1(strHmacText, strHmac)
End If

If strHmacGenerated <> hmac Then
    Dim hmacFail as String
    hmacFail = "<font color=""red"">The HMAC failed validation. Please notify an attendant.</font><br><br>"
    hmacFail+="<font color=""9CBACE"">Received HMAC: " & hmac & "<br>Calculated HMAC: " & strHmacGenerated & "<br>"
    hmacFail+="Make sure the digest functions on the SonicWALL and LHM server match.<br>"
    hmacFail+="Also make sure the shared secret on the SonicWALL and myvars match</font>"
    catchError.Text=hmacFail
End If

End If

End Sub

sub OnBtnClearClicked (Sender As Object, e As EventArgs)
    enteredCode.Text = ""
    LHMResult.Text=""
    catchError.Text=""
end sub

Sub btnSubmit_Click(Sender As Object, E As EventArgs)

    'The following subroutine validates client provided passcodes.
    'The first 6 characters (definable in myvars) are grabbed.
    'These characters are then run though a SHA1 hash with a salt that is defined in myvars.

    '3 pairs of substrings are then retrieved from the hash.
    'The code is validated if the 3 pairs concatenated to the randChars (defined in myvars) characters consist of the following:

    'Validating the 4 0 8 pairs (4,5+0,1+8,9 characters) will provide 1 hour of guest access.
    'Validating the 7 5 2 pairs (7,8+5,6+2,3 characters) will provide 24 hours of guest access.

    grabCode = enteredCode.Text.SubString(0,randChars)

    'Manually compute SHA1 on salt+randomCode, and convert result to base64 - gives stranger output
    Dim sha1 As sha1 = sha1.Create()
    Dim manualHash As Byte() = sha1.ComputeHash(Encoding.UTF8.GetBytes(salt & grabCode))
    Dim hashResult as String = Convert.ToBase64String(manualHash)

    'Alternatively, use forms hash routine - only provides upper case A-Z + 0-9 output.
    'Dim hashResult as String = FormsAuthentication.HashPasswordForStoringInConfigFile(salt & randomCode,"SHA1")

    'First try to match on 1 hour code
    passCode = ""
    passCode = grabCode & hashResult.SubString(4, 2)
    passCode = passCode & hashResult.SubString(0, 2)
    passCode = passCode & hashResult.SubString(8, 2)
    If enteredCode.Text = passCode Then
        sessTimer = "3600"
        authResult.Text="<font color=""green""><b>1 hour code validated.</b></font>"

        'Check the used passcode DB if useDB is enabled in myvars.
        If useDB = "1" Then
            wasItUsed()
        End If
    Else
        'Now try to match on 24 hour code
        passCode = ""
        passCode = grabCode & hashResult.SubString(7, 2)
        passCode = passCode & hashResult.SubString(5, 2)
        passCode = passCode & hashResult.SubString(2, 2)
        If enteredCode.Text = passCode Then
            sessTimer = "86400"
            authResult.Text="<font color=""green""><b>24 hour code validated.</b></font>"

            'Check the used passcode DB if useDB is enabled in myvars.
            If useDB = "1" Then
                wasItUsed()
            End If
        End If
    Else
```



Tech Note

```
again.</font>"
    authResult.Text="<font color=""Red""><b>Passcode cannot be validated.</b><br>The passcode is case-sensitive.<br>Please try
    End If
End If

End Sub

Sub wasItUsed ()
    'Check to see if the root (randChars) of the passcode is already in the used database.
    Dim strConn as string = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & server.mappath("usedcodes.mdb") & ";"
    Dim MySQL as string = "SELECT * From passCodes Where passCode = " & grabCode & ""
    Dim MyConn as New OleDbConnection (strConn)
    Dim cmd as New OleDbCommand (MySQL, MyConn)
    Dim objDR As OleDbDataReader
    Dim isUsed As Boolean

    MyConn.Open()
    objDR = cmd.ExecuteReader()
    isUsed = objDR.Read()
    objDR.Close()
    MyConn.Close()

    'If the passcode is not found in the database
    if isUsed = False
        LHM()
    Else
        authResult.Text="<font color=""Red""><b>Passcode has already been used.</b><br>Please see an attendant for assistance.</font>"
    End If
End Sub

Sub writeToDB ()
    'Try to write the submitted (only randChars characters instead of the whole passcode) info to the database file
    Try
        Dim strConn as string = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & server.mappath("usedcodes.mdb") & ";"

        Dim MySQL as string = "INSERT INTO passCodes (passCode) VALUES (" & grabCode & ")"
        Dim MyConn as New OleDbConnection (strConn)
        Dim cmd as New OleDbCommand (MySQL, MyConn)
        MyConn.Open ()
        cmd.ExecuteNonQuery ()
        MyConn.Close ()

        Catch ex as Exception
            catchError.Text = "<font color=""9CBACE"">" & ex.ToString & "</font>"
        End Try
    End Try

End Sub

Sub LHM()
    'The writeToDB sub is in the Response code 50 - Login Succeeded routine, after the LHM exchange succeeds. You may move it to the top to write the passcode
    to the DB before the LHM transaction for testing purposes.
    'writeToDB ()

    enteredCode.Text = "Code Accepted."

    'Let the user know that we are setting up the session, just in case it takes more than a second
    LHMResult.Text = "Authorizing session. Please wait."

    'The LHM cgi on the SonicWALL - this does not change
    Dim loginCgi as String = "externalGuestLogin.cgi"

    'Assemble the data to post back to the SonicWALL to authorize the LHM session
    Dim loginParams as String = "sessId=" & sessionId & "&userName=" & mac & "&sessionLifetime=" & sessTimer & "&idleTimeout=" & idleTimer

    'Combine mgmtBaseUrl from the original redirect with the login cgi
    Dim postToSNWL as String = mgmtBaseUrl & loginCgi

    'Convert the loginParams to a well behaved byte array
    Dim byteArray As Byte() = Encoding.UTF8.GetBytes(loginParams)

    Try
        'Create the webrequest to the SonicWALL
        Dim toSNWL as WebRequest = WebRequest.Create(postToSNWL)
```



Tech Note

```
'Calculate the length of the byte array
toSNWL.ContentLength = byteArray.Length

'Set the method for the webrequest to POST
toSNWL.Method = "POST"

'Set the content type
toSNWL.ContentType = "application/x-www-form-urlencoded"

'Open the request stream
Dim dataStream As Stream = toSNWL.GetRequestStream()

'Write the byte array to the request stream
dataStream.Write(byteArray, 0, byteArray.Length)

'Close the Stream object
dataStream.Close()

'Get the response
Dim snwlReply As WebResponse = toSNWL.GetResponse()

'Display the status - looking for 200 = OK.
'Response.Write(CType(snwlReply, HttpWebResponse).StatusCode)

'Grab the response and stuff it into an xml doc for possible review
Dim snwlResponse as XmlDocument = New XmlDocument()
snwlResponse.Load(snwlReply.GetResponseStream())

'Set the xPath to the SNWL reply, and get the response
Dim codePath as String = "SonicWALLAccessGatewayParam/AuthenticationReply/ResponseCode"

'Response.Write(snwlResponse.SelectSingleNode(codePath).InnerXml)

'Response code 50 - Login Succeeded

If snwlResponse.SelectSingleNode(codePath).InnerXml = "50"

    'Do we want to provide a logout popup window?
    If logoutPopup = "1" Then
        'Popup hack using Javascript for logout window
        Dim sb As New System.Text.StringBuilder()
        sb.Append("<script language='javascript'>")
        sb.Append("window.open('logout.aspx?sessId=")
        sb.Append(Server.URLEncode(CStr(sessionId)))
        sb.Append("&mgmtBaseUrl=")
        sb.Append(Server.URLEncode(CStr(mgmtBaseUrl)))
        sb.Append("&sessTimer=")
        sb.Append(Server.URLEncode(CStr(sessTimer)))
        sb.Append("'", 'logOut', 'toolbar=no,')
        sb.Append("addressbar=no,menubar=no,")
        sb.Append("width=400,height=250');")
        sb.Append("<")
        sb.Append("/")
        sb.Append("<script>")
        RegisterStartupScript("stp", sb.ToString)
    End If

    LHMResult.Text = "<br><b><font color=""green"">Session authorized:</font></b> You may now go to the URL you originally requested: <a
target=""_blank"" href="" & req & """" & req & ""</a>"

    'Write the passcode the DB if the LHM session succeeds and if useDB = 1.
    If useDB = "1" Then
        writeToDB ()
    End If

'Response code 51 - Session Limit Exceeded
ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "51"
    LHMResult.Text = "<br><b><font color=""red"">Session Limit Reached:</font></b> The maximum number of guest session has been
reached. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 100 - Login Failed.
ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "100"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> Your session cannot be created at this time. Sorry for
the inconvenience. Please close and relaunch your browser to try again."

'Response code 251 - Bad HMAC.
ElseIf snwlResponse.SelectSingleNode(codePath).InnerXml = "251"
```



Tech Note

```
LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed message authentication. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 253 - Invalid SessionID.
ElseIf snwIResponse.SelectSingleNode(codePath).InnerText = "253"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed to match a known session identity. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 254 - Invalid CGI.
ElseIf snwIResponse.SelectSingleNode(codePath).InnerText = "254"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization was missing an essential parameter. Sorry for the inconvenience. Please close and relaunch your browser to try again."

'Response code 255 - Internal Error.
ElseIf snwIResponse.SelectSingleNode(codePath).InnerText = "255"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed due to an unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again."

End If

'Close the streams
dataStream.Close()
snwIReply.Close()

'If there is some asp.net error trying to talk to the SonicWALL, print it in the same color as the background.
Catch ex as Exception
    catchError.Text = "<font color=""9CBACE"">" & ex.ToString & "</font>"
    LHMResult.Text = "<br><b><font color=""red"">Session creation failed:</font></b> The request for authorization failed due to an unspecified error. Sorry for the inconvenience. Please close and relaunch your browser to try again. If the problem persists, please notify an attendant."
End Try

End Sub

</script>

<STYLE>
body {
    font-size: 10pt;
    font-family: verdana,helvetica,arial,sans-serif;
    color:#000000;
    background-color:#9CBACE;
}

tr.heading {
    background-color:#006699;
}

.button {
    border: 1px solid #000000;
    background-color: #ffffff;
}
</STYLE>

<HTML>
<HEAD>
<TITLE>LHM Random Script</TITLE>
</HEAD>

<BODY>
<form id="frmValidator" onKeyPress="if(event.keyCode==13) {document.getElementById('btnSubmit').click(); return false}" runat="server">

<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;</td>
    </tr>
    <tr class="heading">
        <td width="50%" valign="center"><font color="white"><b>Algorithmic Authentication</b></font></td>
        <td><center></center></td>
        <td width="50%" align="right" valign="center"><font color="white"><b>Powered by SonicWALL LHM</b>&nbsp;</font></td>
    </tr>
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;</td>
    </tr>
</table>

<table width="90%" border="0" cellpadding="2" cellspacing="0">
    <tr>
```



Tech Note

Welcome **<%= ip%>** to SonicWALL's LHM Algorithmic Authenticator.

Enter your unique randomly generated passcode to obtain secure guest internet access.

Valid passcodes are not stored anywhere, so validation is not performed against any kind of database. Instead, when a passcode is entered, it is algorithmically validated. Once a passcode is successfully used, it is written to a "used passcode" database so that it cannot be reused. The validator will recognize 1 hour and 24 hour passcodes - these characteristics were encoded within the passcodes themselves during generation.

```
</td>
</tr>
</table>
<table width="100%" border="0" cellpadding="2" cellspacing="0">
  <tr class="heading">
    <td colspan=3 align="center"><font color="white">&nbsp;</td>
  </tr>
</table>
<table width="100%" border="0" cellpadding="2" cellspacing="0">
  <tr><br>
    <td width="30%">Enter your passcode:</td>
    <td width="30%"><asp:TextBox id="enteredCode" runat="server" /></td>
    <td width="40%"><asp:RequiredFieldValidator id="valEnteredCode" ControlToValidate="enteredCode" ErrorMessage="Please enter your passcode."
runat="server" /></td>
  </tr>
  <tr>
    <td colspan=2><asp:Label id=authResult runat="server" />&nbsp;</td>
  </tr>
  <tr>
    <td colspan=2>
      <asp:button id="btnSubmit" class="button" text=" Submit " onClick="btnSubmit_Click" runat="server" />
      &nbsp;&nbsp;&nbsp;
      <asp:button id="btnClear" class="button" text=" Clear " CausesValidation="False" onClick="OnBtnClearClicked" runat="server" />
    </td>
  </tr>
  <tr>
    <td colspan=2><asp:Label id=LHMResult runat="server" /></td>
  </tr>
  <tr>
    <td colspan=2><asp:Label id=catchError runat="server" /></td>
  </tr>
</table>
</form>
</BODY>
</HTML>
```



Tech Note

generator.aspx

```
<%@ Page Language="VB" Debug="true" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Xml" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Data" %>
<%@ Import Namespace="System.Data.OleDb" %>
<%@ Import Namespace="System.Security" %>
<%@ Import Namespace="System.Security.Cryptography" %>
<%@ Import namespace="System.Security.Cryptography.X509Certificates"%>

<!-- #INCLUDE file="myvars.aspx" -->

<script language="VB" runat="server">

Dim genCodes As New ArrayList()
Dim codeType As String

Sub Page_Load(Source as Object, E as EventArgs)
    If Not isPostBack Then
        Heading.Text="&nbsp;"
        btnExport.Visible = False
    End If
End Sub

Sub btnSubmit_Click(Sender As Object, E As EventArgs)
    'The following generates passcodes beginning with a random character generator.
    'The number of characters in randomCode is configurable in myvars.
    'The randomCode output is then run though a SHA1 hash with a salt that is defined in myvars.
    'Note: If you are using this in a live environment, it is important to change the salt to prevent algorithm compromise.

    '3 pairs of substrings are then retrieved from the hash, and concatenated to the randomCode to form the passcode.

    'In the current sample implementation:
    'The 4 0 8 pairs (4,5+0,1+8,9 characters) from the hash will provide 1 hour of guest access.
    'The 7 5 2 pairs (7,8+5,6+2,3 characters) from the hash will provide 24 hours of guest access.

    Dim myLooper As Integer
    Dim passCode as String

    For myLooper = 1 to Convert.ToInt32(codeCount.Text)

        Dim x As Integer = 0
        Dim isItRand as boolean = False
        Dim intRand as Integer = 0
        Dim randomCode as String = ""

        For x = 1 to randChars
            Do Until isItRand = True
                '48 to 57 for numbers, 65 to 90 for uppercase, 97 to 122 for lowercase
                intRand = Int((122 - 48 + 1) * Rnd + 48)
                'Select the legal character set for randomCode by including legal characters below.
                If InStr(1, "abcdefghijklmnopqrstuvwxyzABCDEFGHJKLMNPQRSTUVWXYZ23456789",Chr(intRand), 1) Then
                    isItRand = True
                End If
            Loop
            randomCode = randomCode & Chr(intRand)
            isItRand = False
        Next

        'Manually compute SHA1 on salt+randomCode, and convert result to base64 - gives stranger output
        Dim sha1 As sha1 = sha1.Create()
        Dim manualHash As Byte() = sha1.ComputeHash(Encoding.UTF8.GetBytes(salt & randomCode))
        Dim hashResult as String = Convert.ToBase64String(manualHash)

        'Alternatively, use forms hash routine - only provides upper case A-Z + 0-9 output.
        'Dim hashResult as String = FormsAuthentication.HashPasswordForStoringInConfigFile(salt & randomCode,"SHA1")

        If dropDownList1.SelectedItem.Value = "1 Hour" Then
            passCode = randomCode & hashResult.SubString(4, 2)
            passCode = passCode & hashResult.SubString(0, 2)
            passCode = passCode & hashResult.SubString(8, 2)
            genCodes.Add(passCode)
        Else
```



Tech Note

```
passCode = randomCode & hashResult.SubString(7, 2)
passCode = passCode & hashResult.SubString(5, 2)
passCode = passCode & hashResult.SubString(2, 2)
genCodes.Add(passCode)

End If

Next

btnExport.Visible = True
heading.Text = "Your " & codeCount.Text & " <b>" & dropDownList1.SelectedItem.Value & "</b> Passcodes:"
genOutput.DataSource = genCodes
genOutput.DataBind()
codeCount.Text=""

'Store the genCodes array in session state for retrieval for printing and exporting
Session("myGenCodes") = genCodes
Session("codeType") = dropDownList1.SelectedItem.Value

End Sub

Sub printIt(Src As Object, e As DataListCommandEventArgs)
    If not Session.Item("myGenCodes") is Nothing Then
        genCodes=Session.Item("myGenCodes")
        codeType=Session.Item("codeType")
        'response.write(CStr(genCodes.Item(e.Item.ItemIndex)))

        'Popup hack using Javascript so that individual entries can be printed from the DataList
        Dim sb As New System.Text.StringBuilder()
        sb.Append("<script language='javascript'>")
        sb.Append("window.open('print.aspx?genCode=")
        sb.Append(Server.URLEncode(CStr(genCodes.Item(e.Item.ItemIndex))))
        sb.Append("&sessLife=")
        sb.Append(Server.URLEncode(codeType))
        sb.Append("'", 'printCode', 'toolbar=no,')
        sb.Append("addressbar=no,menubar=no,")
        sb.Append("width=400,height=250');")
        sb.Append("<")
        sb.Append("/")
        sb.Append("script>")
        RegisterStartupScript("stp", sb.ToString)
    End If

End Sub

Sub exporter(Sender As Object, E As EventArgs)
    If not Session.Item("myGenCodes") is Nothing Then
        genCodes=Session.Item("myGenCodes")

        'Convert the genCodes array to a string with CRs for later conversion to a byte array
        Dim i as Integer
        Dim genCodeString as String
        for i = 0 To genCodes.Count - 1
            genCodeString += CStr(genCodes.Item(i)) & Chr(13)
        Next

        'response.write(genCodeString)

        'Create the byte array and send it to the browser as genCodes.csv
        Dim data() As Byte = System.Text.ASCIIEncoding.ASCII.GetBytes(genCodeString)
        Response.Clear()
        Response.AddHeader("Content-Type", "application/Excel")
        Response.AddHeader("Content-Disposition", "inline;filename=genCodes.csv")
        Response.BinaryWrite(data)
        Response.End()
    End If

End Sub

</script>

<STYLE>
body {
    font-size: 10pt;
    font-family: verdana,helvetica,arial,sans-serif;
    color:#000000;
    background-color:#9CBACE;
}
```

Tech Note

```
tr.heading {
    background-color:#006699;
}

.button {
    border: 1px solid #000000;
    background-color: #ffffff;
}
</STYLE>

<HTML>
<HEAD>
<TITLE>LHM Random Script</TITLE>
</HEAD>

<BODY>
<form id="frmValidator" runat="server">

<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;  </td>
    </tr>
    <tr class="heading">
        <td width="50%" valign="center"><font color="white"><b>Algorithmic Authentication</b></font></td>
        <td><center></center></td>
        <td width="50%" align="right" valign="center"><font color="white"><b>Passcode Generator</b>&nbsp;</font></td>
    </tr>
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;</td>
    </tr>
</table>

<table width="90%" border="0" cellpadding="2" cellspacing="0">
    <tr>
        <td><b>Welcome to SonicWALL's LHM Algorithmic Generator.</b><br><br>This will allow you to create randomly generated passcodes for secure
        guest internet access.<br><br>Valid passcodes are not stored anywhere, so validation is not performed against any kind of database. Instead, when a passcode is entered, it
        is algorithmically validated. Once a passcode is successfully used, it is written to a "used passcode" database so that it cannot be reused.<br><br>The validator will recognize
        1 hour and 24 hour passcodes - these characteristics were encoded within the passcodes themselves during generation.<br><br>
    </td>
    </tr>
</table>

<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr class="heading">
        <td colspan=3 align="center"><font color="white">&nbsp;</td>
    </tr>
</table>

<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr><br>
        <td width="15%">Passcode type:</td>
        <td width="10%"><asp:DropDownList id="DropDownList1" runat="server">
            <asp:ListItem>1 Hour</asp:ListItem>
            <asp:ListItem>24 Hours</asp:ListItem>
        </asp:DropDownList></td>
        <td width="20%">Number to generate:</td>
        <td width="20%"><asp:TextBox id="codeCount" runat="server" /></td>
        <td width="50%"><asp:RequiredFieldValidator id="valcodeCount" ControlToValidate="codeCount" ErrorMessage="Enter a value." Font-Size="10"
        Display="Dynamic" runat="server" />
        <asp:RangeValidator id="Range1" ControlToValidate="codeCount" MinimumValue="1" MaximumValue="999" Type="Integer" Font-Size="10"
        ErrorMessage="Values from 1 to 999." runat="server" /></td>
    </tr>
    <tr>
        <td colspan=3></td>
        <td><asp:button id="btnSubmit" class="button" text=" Submit " onClick="btnSubmit_Click" runat="server" />&nbsp;&nbsp;&nbsp;<asp:button id="btnExport"
        class="button" text=" Export " CausesValidation="False" onClick="exporter" runat="server" /><br></td>
    </tr>
    <tr>
        <td colspan=3></td>
        <td><br></td>
    </tr>
    <tr><tr class="heading">
        <td colspan=5><font color="white"><asp:Label id=heading runat="server" /></td>
    </tr>
    <tr>
        <td colspan=5><br></td>
    </tr>
</table>

<asp:DataList id="genOutput" Runat="Server" RepeatColumns="4" RepeatDirection="Horizontal" CellPadding="0" Cellspacing="0" GridLines="Both" align="center"
OnItemCommand="printIt">
    <ItemTemplate>
        <td>
```



Tech Note

```
<asp:Label Text='<%= # Container.DataItem %>' Runat="Server"/>
</td>
<td>
<asp:ImageButton id="print" runat="server" ImageUrl="print.gif" EnableViewState="False" CausesValidation="False" CommandName='<%= # Container.DataItem
%>' />
</td>
</ItemTemplate>
</asp:DataList>

</form>
</BODY>
</HTML>
```

print.aspx

```
<!-- #INCLUDE file="myvars.aspx" -->

<script language="VB" runat="server">

Dim genCode as String
Dim sessLife as String

'Grab the code and the session lifetime from the generator page
Sub Page_Load(src as Object, e as EventArgs)
    genCode=Request.QueryString("genCode")
    sessLife=Request.QueryString("sessLife")
End Sub

</script>
<STYLE>
body {
    font-size: 10pt;
    font-family: verdana,helvetica,arial,sans-serif;
    color:#000000;
    background-color:#9CBACE;
}
tr.heading {
    background-color:#006699;
}
</STYLE>
<BODY>
<table width="100%" border="0" cellpadding="2" cellspacing="0">
    <tr class="heading">
        <td colspan=2 align="center"><font color="white">&nbsp;</td>
    </tr>
    <tr class="heading">
        <td colspan=2 align="center"></td>
    </tr>
    <tr class="heading">
        <td colspan=2 align="center"><font color="white">&nbsp;</td>
    </tr>
    <tr><td><br><br></td></tr>
    <tr>
        <td>Your Pass Code is:</td>
        <td><b><%= genCode%></b></td>
    </tr>
    <tr><td><br></td></tr>
    <tr>
        <td>Session Lifetime is:</td>
        <td><b><%= sessLife%></b></td>
    </tr>
</table>

<script language='javascript'>window.print();</script>

</BODY>
</HTML>
```