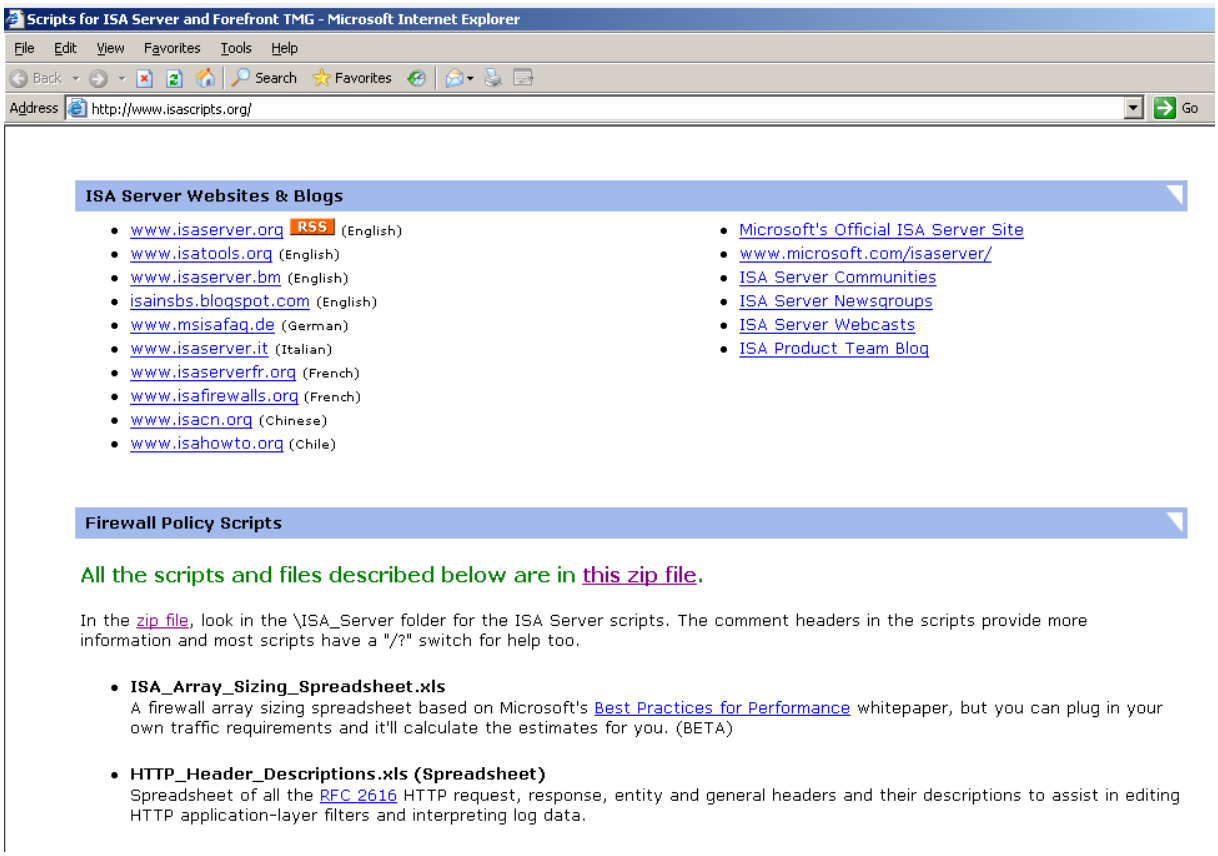
ISA Server Surfmaster – Beta 1

Das folgende Script von Jason Fossen (MVP ISA), schaltet per Script einzelne Firewallregeln an oder aus.

Diese Funktion kann dazu in Kombination mit einer simplen Webseite verwendet werden, um zum Beispiel Bildungseinrichtungen die Moeglichkeit zu geben, die InternetNutzung fuer einzelne Klassenraeume ein- und auszuschalten.

Der Lehrer hat somit die Moeglichkeit, selbst die InternetNutzung fuer seinen Klassenraum ein- oder auszuschalten.



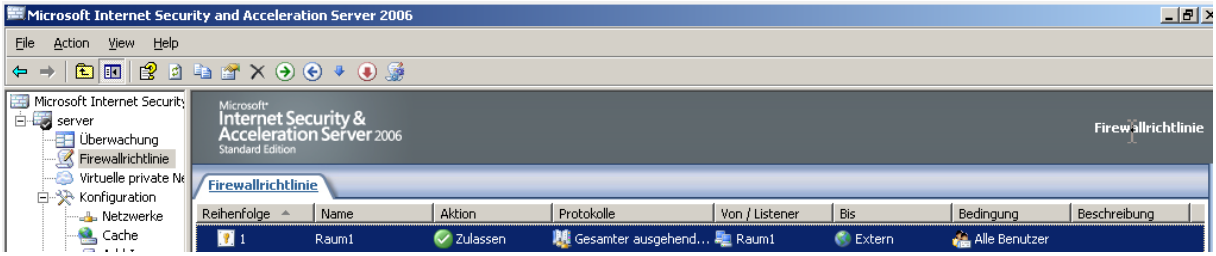
Das Script:

```
'*********************************************************************************
***
' Script Name: ISA_Enable-Disable_Rule.vbs
'     Version: 1.0
'      Author: Jason Fossen ( www.ISAscripts.org )
'Last Updated: 16.Oct.2005
'     Purpose: Enables or disables a rule in the Firewall Policy of an ISA Server array,
'              Standard or Enterprise edition.  But cannot manage System Policy rules or
'              Enterprise Policy rules, array or single-server rules only.
'       Legal: Public Domain.  Modify and redistribute freely.  No rights reserved.
'              SCRIPT PROVIDED "AS IS" WITHOUT WARRANTIES OR GUARANTEES OF ANY KIND.
'              USE AT YOUR OWN RISK.  Test on non-production servers first.
'*********************************************************************************
If WScript.Arguments.Count <> 2 Then Call ShowHelpAndQuit()
```

```
sRuleName = WScript.Arguments.Item(0)
sAction   = WScript.Arguments.Item(1)
If (LCase(sRuleName) = "/?") Or (LCase(sRuleName) = "/h") Or (LCase(sRuleName) = "-h") Then Call
ShowHelpAndQuit()
If EnableOrDisableRule(sRuleName, sAction) Then
   WScript.Echo vbCrLf & "Success! " & UCase(sRuleName) & " = " & UCase(sAction) & "D"
Else
   WScript.Echo vbCrLf & "ERROR: " & Err.Number & " " & Err.Description
End If
'*********************************************************************************
' Functions() & Procedures()
'*********************************************************************************
'
' sRuleName is the name of the rule, in doublequotes if it contains spaces.
' sAction is either "enable" or "disable" (or just "e" and "d").
'
' Function returns true if either it is successful or if sRuleName Is
' already set to sAction specified.
'
Function EnableOrDisableRule(sRuleName, sAction)
   On Error Resume Next
   If Not IsObject(oFPC) Then Set oFPC = CreateObject("FPC.Root")
   Set oPolicyRule = oFPC.GetContainingArray.ArrayPolicy.PolicyRules.Item(sRuleName)
   'If Err.Number = -2147024894 Then WScript.Echo "Cannot find the rule named " & sRuleName
   If Err.Number <> 0 Then EnableOrDisableRule = False : Exit Function
   If Left(LCase(sAction),1) = "e" Then bState = True Else bState = False
   If oPolicyRule.Enabled = bState Then EnableOrDisableRule = True : Exit Function
   oPolicyRule.Enabled = bState
   oPolicyRule.Save
   If Err.Number = 0 Then EnableOrDisableRule = True Else EnableOrDisableRule = False
   'If Err.Number <> 0 Then WScript.Echo "Problem changing rule state."
   On Error Goto 0
End Function
Sub ShowHelpAndQuit()
   Dim sUsage : sUsage = vbCrLf
   sUsage = sUsage & vbCrLf
   sUsage = sUsage & "ISA_Enable-Disable_Rule.vbs rulename action" & vbCrLf
   sUsage = sUsage & vbCrLf
   sUsage = sUsage & "Purpose: Enables or disables a rule, not including System Policy rules." &
vbCrLf
   sUsage = sUsage & vbCrLf
   sUsage = sUsage & "   Args: rulename = Name of the rule, placed in doublequotes if necessary." &
vbCrLf
   sUsage = sUsage & "         action   = The word ""Enable"" or ""Disable"" (not case sensitive)." &
vbCrLf
   sUsage = sUsage & vbCrLf
   sUsage = sUsage & "  Legal: SCRIPT PROVIDED ""AS IS"" WITHOUT WARRANTIES OR GUARANTEES
OF ANY" & vbCrLf
   sUsage = sUsage & "         KIND. USE AT YOUR OWN RISK. Public domain, no rights reserved." &
vbCrLf
   sUsage = sUsage & "         ( www.ISAscripts.org )" & vbCrLf
   sUsage = sUsage & vbCrLf
   WScript.Echo sUsage
```

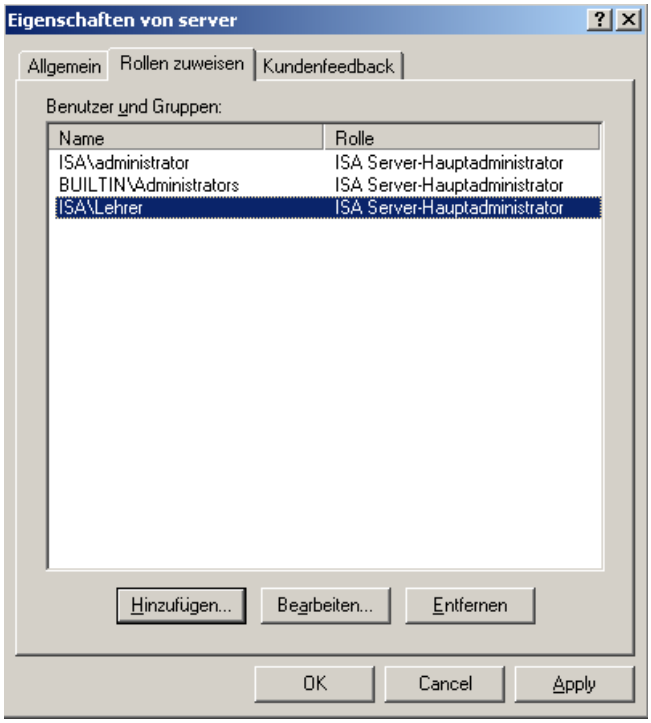
```
    WScript.Quit
End Sub
'EOF*************************************************************************
```

Anlegen einer Regel, welche per Skript aktiviert oder deaktiviert werden soll

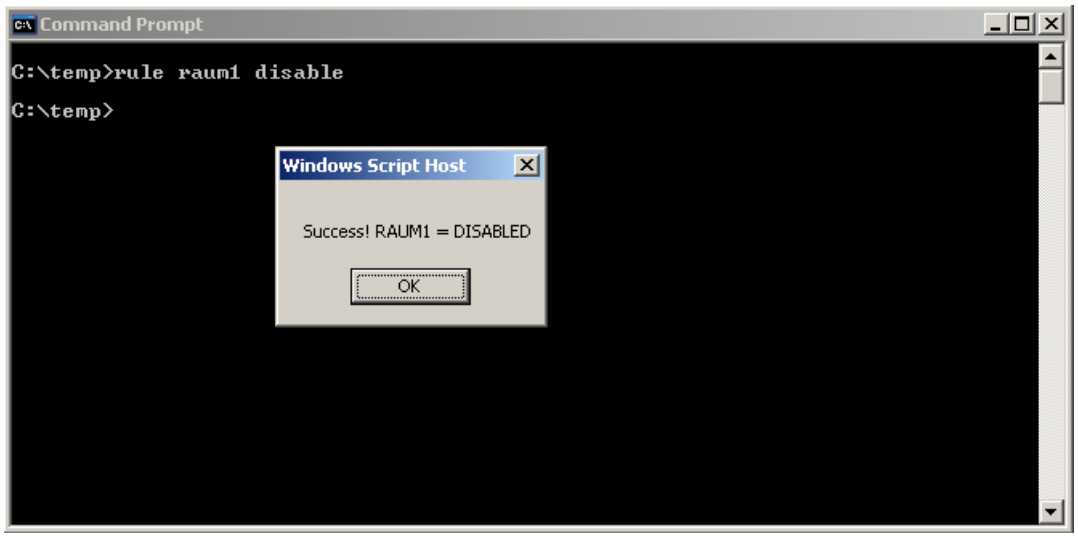


Berechtigungen zur Regel-Aktivierung/Deaktivierung fuer Benutzer erteilen

ISA MMC starten



Ausfuehren des Scripts



Meldung unterdruecken

Original:

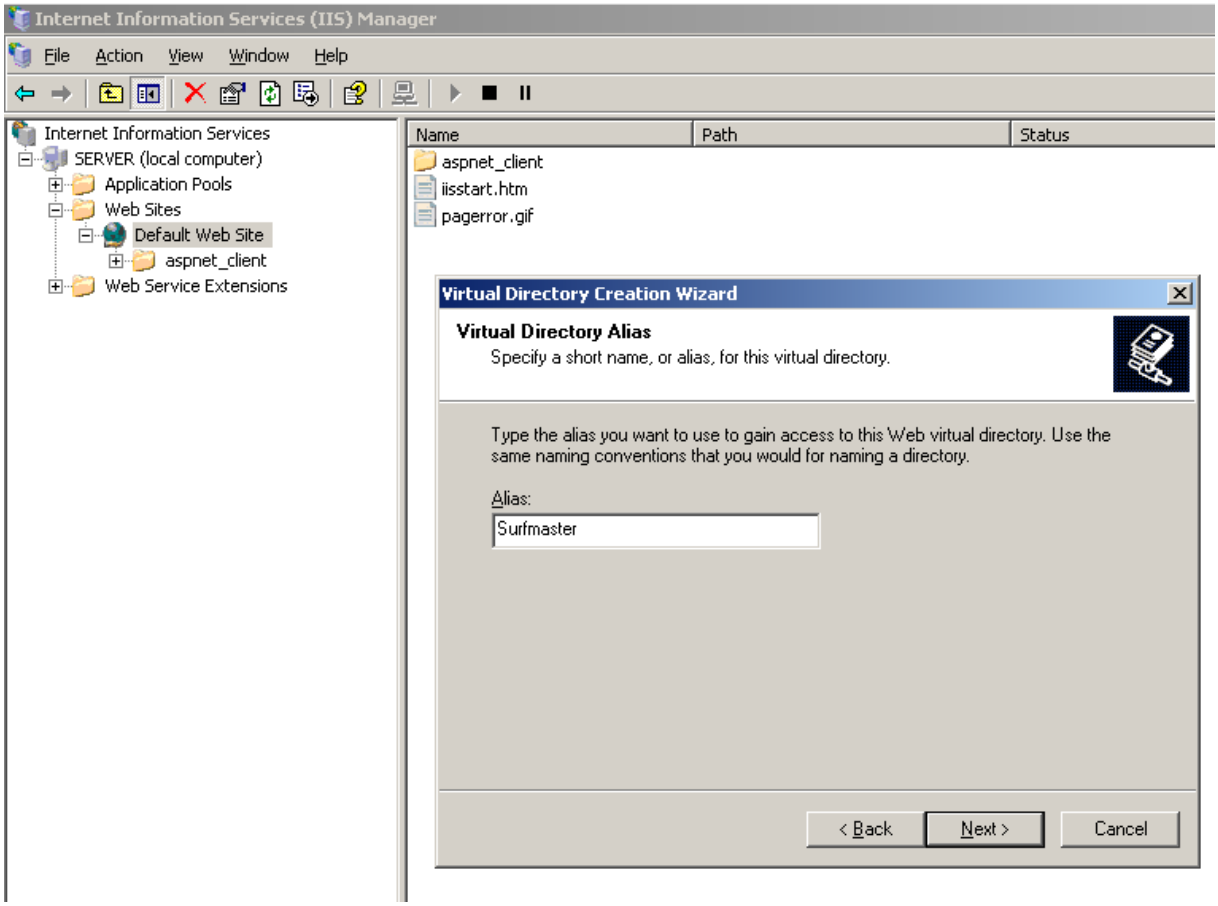
```
If EnableOrDisableRule(sRuleName, sAction) Then
   WScript.Echo vbCrLf & "Success! " & UCase(sRuleName) & " = " & UCase(sAction) & "D"
Else
   WScript.Echo vbCrLf & "ERROR: " & Err.Number & " " & Err.Description
End If
```

Angepasst:

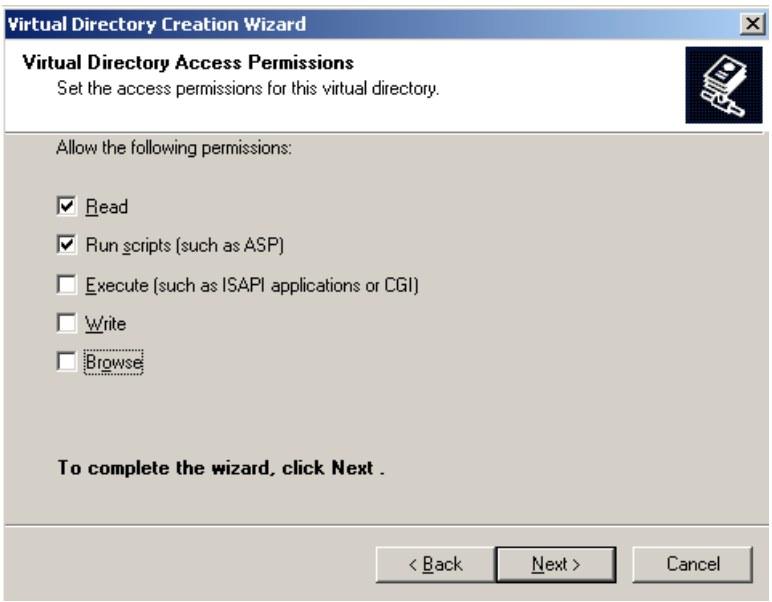
```
If EnableOrDisableRule(sRuleName, sAction) Then
   UCase(sRuleName) & " = " & UCase(sAction) & "D"
End If
```

Das ganze per Weboberflaeche

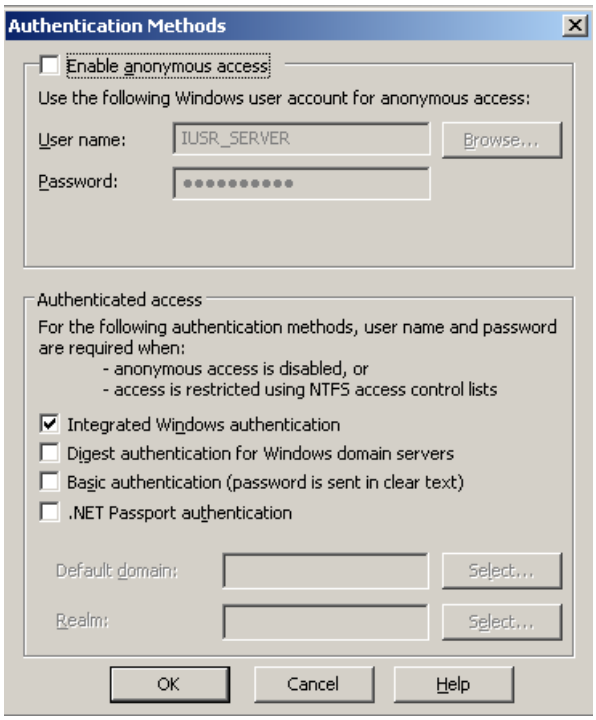
Neues virtuelles Verzeichnis anlegen



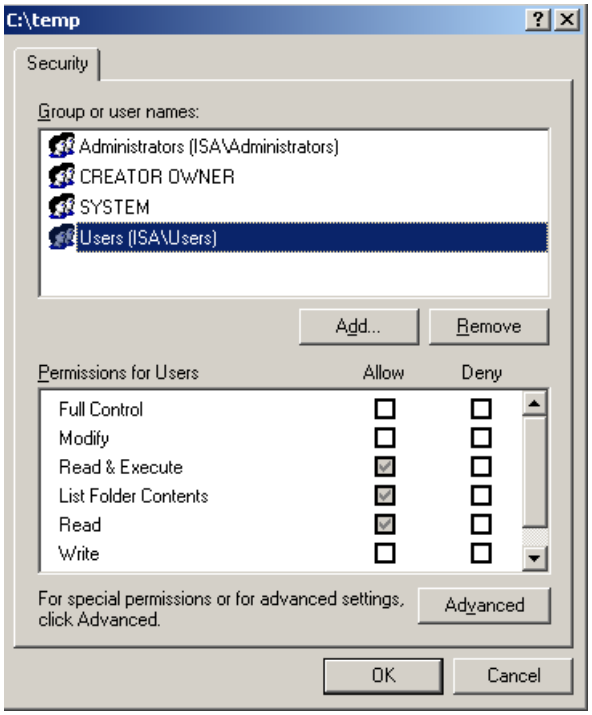
Ausfuehr-Berechtigungen vergeben



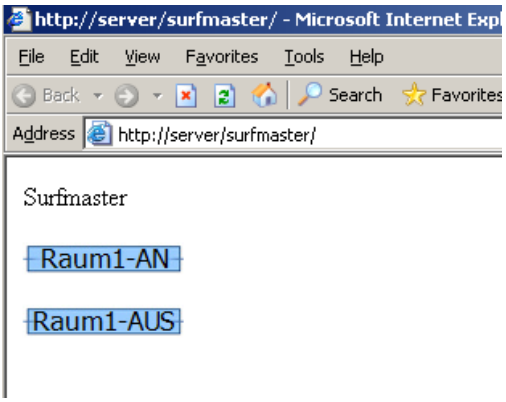
Authentifizierung auf Windows integriert setzen



Berechtigungen vergeben



Etwas webdesignen (und sicherlich besser als ich)



Und fertig ist die Loesung!

Sprachprobleme

Deutsches Windows + Deutscher ISA = Fehlermeldung
Englisches Windows+Deutscher ISA = OK
Englisches Windows+Englischer ISA = OK

Der andere – bessere Ansatz

VBS nicht verwenden, sondern stattdessen ASP.NET und das Script dahin „uebersetzen“. Wir Ihr wisst, keine Taetigkeit die ich machen kann.
Durch meine Zusammenarbeit mit Christmann Informationstechnik (www.christmann.info), hat sich einer der genialen Programmierer (Gunnar) von Christmann der Sache angenommen. Ich habe lediglich das Grundgeruest geliefert, sowie ein VM zum testen und stand fuer die ein oder andere administrative Frage zur Verfuegung.

Hier ein Auszug aus der Default.aspx

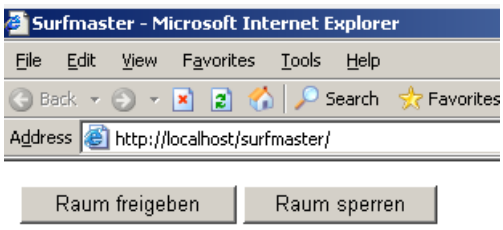
```
<%@ Page Language="VB" AutoEventWireup="false" CodeFile="Default.aspx.vb" Inherits="_Default"
Debug="true" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title>Surfmaster</title>
   <script runat="server">
      Sub Button_ToggleRoom_Click(ByVal Sender As Object, ByVal E As CommandEventArgs)
         ' Create the root object.
         Dim root As New FPCLib.FPC  ' The FPCLib.FPC root object
         root = CreateObject("FPC.Root")

         ' Declare the other object needed.
         Dim isaArray As FPCLib.FPCArray ' An FPCArray object

         ' Connect to the array of the remote ISA Server computer
         ' specified and retrieve the array object for it.
         isaArray = root.Arrays.Connect("server")
```

```
Dim rule As FPCLib.FPCPolicyRule
rule = isaArray.ArrayPolicy.PolicyRules.Item(E.CommandName)
Dim enable As Boolean
```

Angesprochen werden die COM-Objekte ueber die Datei MSFPCCOM.DLL.

Im IIS wie oben beschrieben, ein neues virtuelles Verzeichnis anlegen und die Default.aspx, sowie Web.CONFIG ablegen

Danach kann die Webseite geoeffnet werden:



In der finalen Version wird es eine automatische Installationsroutine geben und auf der Webseite der Status der Regel angezeigt werden, ob der entsprechende Raum freigegeben oder gesperrt ist.