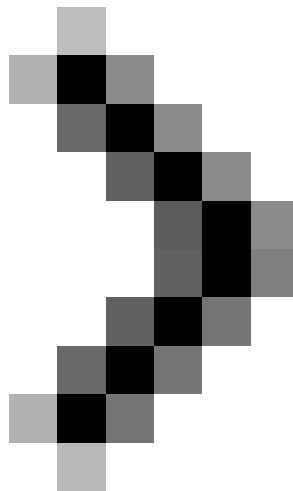
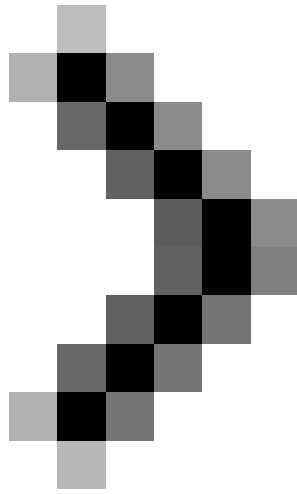


# Managing Windows Networks with Scripts - Part 10: Tricks of remote control scripts

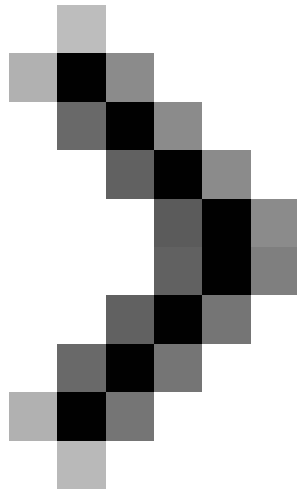
In the previous articles, we have been introduced to some concepts and issues around remote Windows scripting. In this section, I will continue the discussion by showing you two tricks related to this remote script using WMI script written in VBScript.



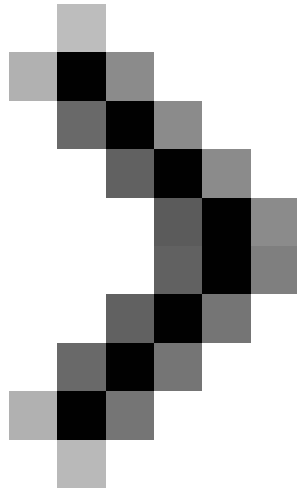
## Part 1: Basic concepts



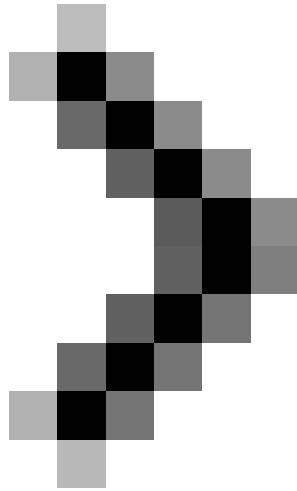
**Part 2: Complete the script**



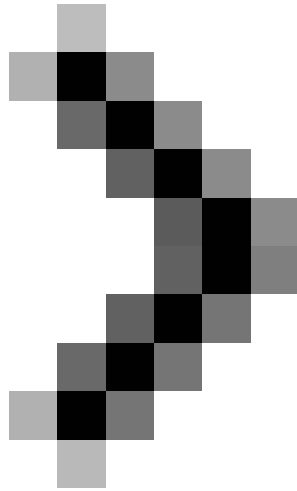
### **Part 3: Understanding WMI**



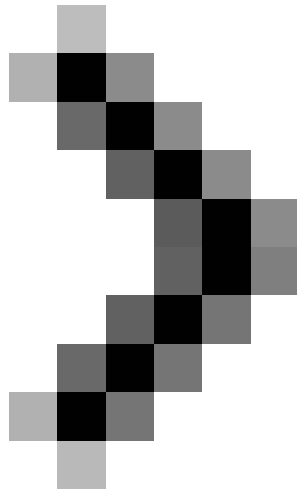
#### **Part 4: Use Win32\_NetworkAdapterConfiguration**



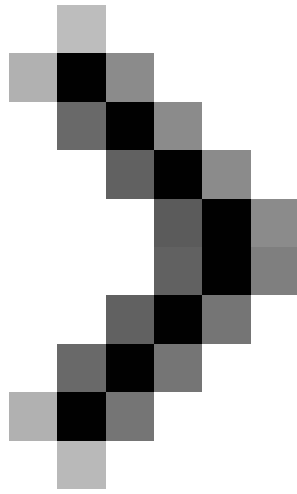
## **Part 5: Overcoming challenges**



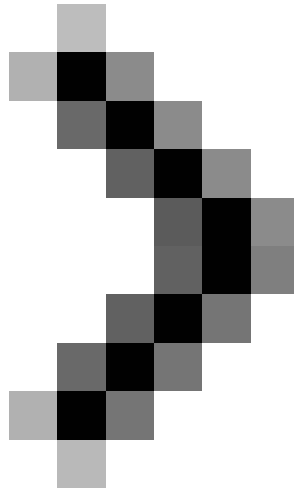
## **Part 6: The first steps for remote scripting**



## Part 7: Troubleshooting errors



## **Part 8: Remote script error handling with Network Monitor 3.0**



## **Part 9: Understanding remote control scenarios**

**Mitch Tulloch**

**In the previous articles, we have been introduced to some concepts and issues around remote Windows scripting. In this section, I will continue the discussion by showing you two tricks related to this remote script using WMI script written in VBScript.**

### **Tip 1: Create Cscript.exe, configure the default script on remote computers**

This is the first trick and it's really simple but also quite clever so we need to have some basic knowledge to easily access. Make sure you know some ways to launch scripts on Windows computers. For example, if you have the ChangeIPAddress.vbs script on a computer, you can launch it by:

- Double-click the .vbs file or shortcut to the file.
- Click Start, then click Run, enter ChangeIPAddress.vbs and click OK.
- Open a command window and navigate to the script folder, enter ChangeIPAddress.vbs and then press ENTER.

What if you do these things, the answer depends on which default settings are set for the Windows Script Host (WSH) on your computer. The Windows Script Host (WSH) is a standalone scripting language used for scripting engines, which means that WSH uses VBScript scripting engine to run VBScript scripts, so WSH acts as an

'environment' within Your script runs. However, WSH actually has two default script configurations:

- Wscript.exe, provides a Windows dialog box for setting script properties and displaying script output.
- Cscript.exe, allows you to configure script properties and display script output from the command prompt.

See the difference between them in case you don't know or forget. We will use the ChangeIPAddress.vbs script from Part 2 of this series to demonstrate. Open a command prompt on your Windows Vista computer and use this script to change the computer's IP address to 172.16.11.173. Now the first thing to note is that changing the network configuration settings that require local administrative standards on the machine, to do this we need to right-click the shortcut of the command window under Accessories and Select Run As Administrator. When you do, the User Account Control (UAC) window appears, click Continue to continue (if the first user account is a member of the local administrative group on the computer) or enter the relevant information. key of the local administrator account (if the account is only a member of the local user group on the machine).

Open the administrative and command level command window to change the address of the computer (Figure 1).

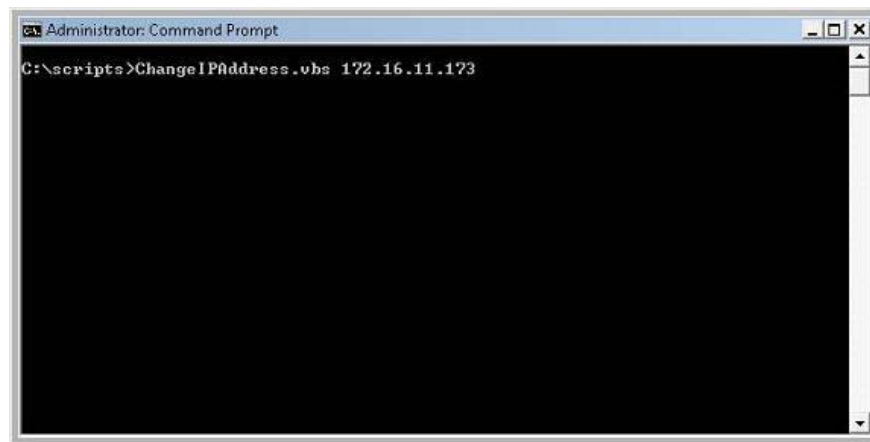


Figure 1: Change the IP address using the script

What happens when we press ENTER, when the button is pressed a few seconds later, the following dialog box appears (Figure 1).

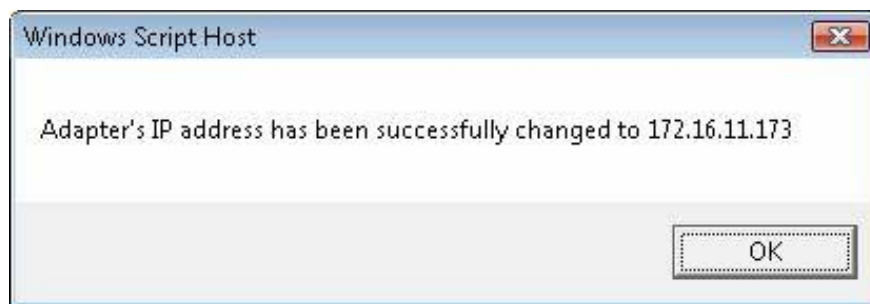


Figure 2: The output of the script appears as a dialog box

Where did this announcement come from? Recall that our script `Change IPAddress.vbs` includes the following lines at the end of the script:

```
'Display k?t qu? ho?c mã l?i

If errEnableStatic = 0 Then
Wscript.Echo "Adapter's IP address has been successfully changed to" & strAddress
Else
Wscript.Echo "Changing the adapter's address was not successful. Error code" & errEnableStatic
End If
```

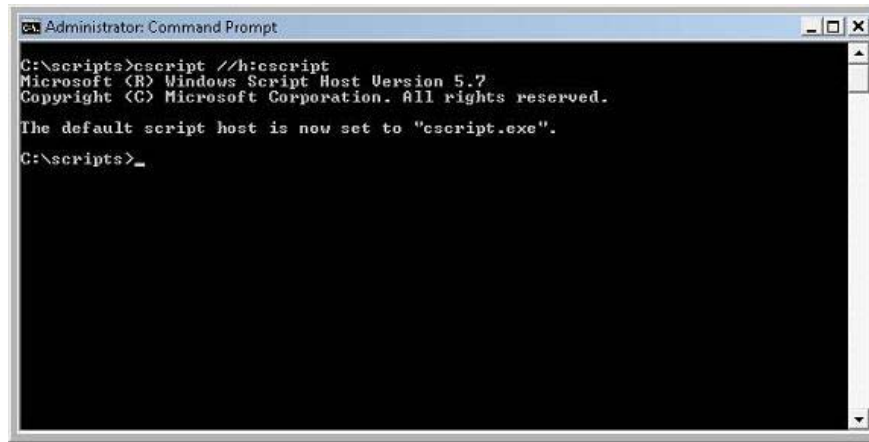
So what's happening is that the `Wscript.Echo` command is displaying the output (ie displaying a dialog box) instead of displaying the output inside the command window. The reason is because by default `Wscript.exe` has the default script configuration and this configuration is set for what it displays with the same window output.

Can we stop these behavior and get the script output to display inside the command window? One way to deal with this task is to call `Cscript.exe` configured with the command line script when running the script. You can do this like Figure 3 below:



Figure 3: Use `cscript.exe` to output the script output inside the command window

However, this may annoy you because you have to type `cscript` before you name the script the same way, so you can set `Cscript.exe` as a default script configuration for all WSH calls by executing. This (Figure 4):

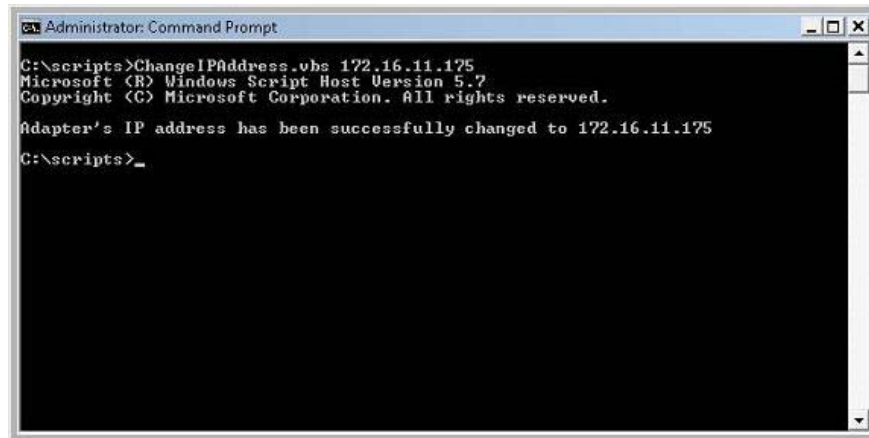


```
Administrator: Command Prompt
C:\scripts>cscript //h:cscript
Microsoft (R) Windows Script Host Version 5.7
Copyright (C) Microsoft Corporation. All rights reserved.

The default script host is now set to "cscript.exe".
C:\scripts>_
```

Figure 4: Create cscript.exe for the default configuration

Now we can run the script and display its output from within the command window without having to type cscript first (Figure 5):



```
Administrator: Command Prompt
C:\scripts>ChangeIPAddress.vbs 172.16.11.175
Microsoft (R) Windows Script Host Version 5.7
Copyright (C) Microsoft Corporation. All rights reserved.

Adapter's IP address has been successfully changed to 172.16.11.175
C:\scripts>_
```

Figure 5: When Cscript.exe is configuring the default script, output The script is displayed inside the command window

We have a scripting group like ChangeIPAddress.vbs that wants to run remotely by deploying them to destination machines such as logon scripts or starting scripts with Group Policy. In addition, some scripts have Wscript.Echo statements in them that create script output. What happens when one of those scripts is deployed to remote computers and running on the computer? A series of windows will appear on the user's desktop as the script runs on the computer, and users will have to click OK, OK and OK, . until these pop-up windows disappear and the script finish its work. So is there any way to overcome this phenomenon?

We can overcome this phenomenon in two ways. First, you can edit all scripts to delete or annotate the Wscript.Echo statement so that the script does not generate any output. That's a bit of a hindrance, especially if you have a large number of scripts. So what is the second method? Here is a little trick we want to share with you:

*In the Active Directory environment in Group Policy that is being used to manage desktop computers, you can*

*change the script configuration from Wscript.exe to Cscript.exe on computers in an OU using the steps below:*

*1. Use Notepad to create a text file called ChangeToCscript.bat with the following two lines in it:*

```
@echo off  
cscript //h: cscript //s
```

*2. Open the GPO linked to the OU and navigate to Computer Configuration>Windows Settings>Scripts>Startup.*

*3. Double-click the Startup startup policy setting to open its properties page.*

*4. Click the Show Files button and copy / paste the file ChangeToCscript.bat from Windows Explorer to the SYSVOL subdirectory, where startup scripts are located.*

*5. Click the Add button on the properties page for the boot policy setting.*

*6. Click the Browse button and select ChangeToCscript.bat.*

*7. Close all property pages.*

*8. The addition of this startup script will cause the default script on target computers to be transferred from Wscript.exe to Cscript.exe during the next reboot of those computers, and this will work without regard to the target user as a standard user or a local administrator.*

*Note : ChangeToCscript.bat must be run as a startup script and not a login script. If you run it as a login script, it will only work when the target user is an administrator on the computer.*

What can you do with only one file with two lines? Now you can deploy the scripts you want to target computers and not have to worry about users having a lot of pop-ups appearing on their screens.

### **Trick 2: Execute 'runAs' without the specified information**

The second trick was submitted to us by one of the readers after reading the previous sections of this series. We are allowed to quote directly from his explanatory email:

FYI is how I implement 'runAs' without having to enter other important information, is to use the local administrator account for WMI, then save the admin password in a text file on a network Sharing is protected by NTFS permissions. For example, if 'tech1', 'tech4' and 'tech5' are normal domain accounts (not domain administrators), but these users are authorized to run WMI scripts, then I allow access updating these NTFS accounts to a text file, this text file includes a local admin password. Then there is the script to enter password and connect using `Set objWMIService = GetObject ("winmgmts:" & strComputer & "rootcimv2", strComputer & 'Administrator', strImportedPassword) .`

The above is not 100% correct. I use.

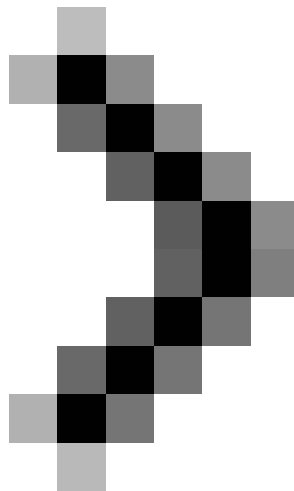
```
Set objSWbemLocator = CreateObject ("WbemScripting.SWbemLocator")  
objSWbemLocator.Security_.Privileges.AddAsString ("SeSecurityPrivilege")  
Set objWMIService = objSWbemLocator.ConnectServer (strComputer, "rootcimv2", strComputer &  
"administrator", strPWD, "", "", & H80)
```

```
Set objReg = objSWbemLocator.ConnectServer (strComputer, "rootdefault", strComputer & "administrator",  
strPWD, "", "", & H80)  
Set oReg = objReg.Get ("StdRegProv")
```

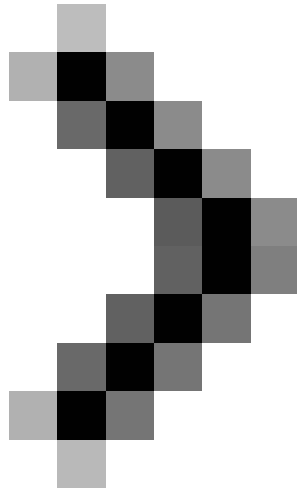
. to complete the same thing.

Connecting to a local admin account has another interesting effect; it does not leave a copy of your domain administrator account profile on the computer in Documents and Settings. When I used my domain account, I had some users ask 'Why are you connected to my computer' and wondering what kind of snooping is going on. By using a local admin account, they do not recognize anyone who is connected (they are blocked from viewing security records).

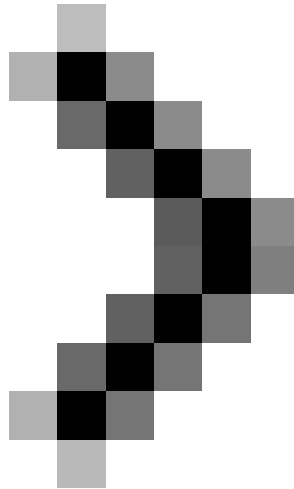
With the log, in fact we have a different local administrator password pair used here; so the script enters a list and tries them one by one until successful or out of options.



## Part 11: Other script tricks



## Part 12: Properties of the WMI class



### **Part 13: The script returns all values**

You finished reading the article "**Managing Windows Networks with Scripts - Part 10: Tricks of remote control scripts**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.