



Chapter 6

Server Management

In this chapter:

Enable Shadow Copy.....	66
Inventory Shadow Copy.....	70

Server management is obviously a broad category. For this chapter, I provide information specific to the Shadow Copy functionality of Microsoft® Windows Server™ 2003. Tasks automated in this chapter include:

- Enabling Volume Shadow Copy on a specified volume on one or more servers
- Inventorying Shadow Copy information on one or more servers

Enable Shadow Copy



On the CD The sample script can be found on the CD that accompanies this book at `\Chap6\MakeShadowCopy\MakeShadowCopy.wsf`.

Operating System	Supported?	Prerequisites
Microsoft Windows® 2000 family	No	<ul style="list-style-type: none"> ■ Windows Script Host (WSH) 5.6 or later
Windows XP Professional	No	<ul style="list-style-type: none"> ■ Microsoft Windows Management Instrumentation (WMI)
Windows Server 2003 family	Yes	<ul style="list-style-type: none"> ■ Administrative permission on targeted computers ■ Network connectivity to each remote computer

Description

The Windows Server 2003 Shadow Copy functionality is enabled on a per-volume basis (although Shadow Copies are made only of files residing in shared folders within that volume). This task creates a single Shadow Copy for a specified volume on one or more servers. Using Shadow Copies is recommended because it allows a user to retrieve an older version of a file on her own—she doesn't have to contact your organization's help desk and request that a backup tape be used to restore a previous version of a file.

Performing This Task Manually

Creating a Shadow Copy for a specific volume isn't difficult, but it becomes time-consuming when you must perform this task on multiple computers. To manually enable Shadow Copy, follow these steps:

1. Right-click the desired volume in My Computer, and select Properties from the shortcut menu.
2. Select the Shadow Copies tab. You can manage the Shadow Copy feature on *all* volumes, no matter which volume you right-clicked.
3. Select the desired volume from the list. The list indicates how many shared folders are available on the volume.
4. Click Enable to enable the Shadow Copies feature.
5. Read the information presented in the Enable Shadow Copies dialog box, and then click Yes.

After Shadow Copies is enabled, you can create a new Shadow Copy by clicking the Create Now button in the volume's Properties dialog box.

Example

You can use this tool in three ways. First, you can target a single remote computer named ServerA and create a Shadow Copy for the C:\ volume:

```
MakeShadowCopy.wsf /computer:ServerA /volume:C:\
```

Second, you can target a list of computers from a text file. The text file is expected to contain one computer name per line, with no other information in the file. Assuming the file is named C:\Computers.txt, you would use this syntax to create a Shadow Copy on the E volume of each server:

```
MakeShadowCopy.wsf /list:C:\Computers.txt /volume:E:\
```

Third, you can target an entire organizational unit (OU) of computer accounts. If your domain contains an OU named West, you would use the following syntax:

```
MakeShadowCopy.wsf /container:west /volume:E:\
```

Note that the */container* argument will work only against the default domain of the computer running the script. In other words, the OU specified must exist within the same domain that the computer running the script belongs to. If the specified OU has nested OUs, you can include their computer accounts as well by specifying one additional argument:

```
MakeShadowCopy.wsf /container:west /recurse /volume:E:\
```

Additional arguments provide extra functionality to the command, as described in the following section titled "Syntax."

Syntax

This script can be executed as a command-line utility. Set CScript.exe to be your default script processor, as described in Chapter 3, "Working with VBScript."

<code>/list:path</code>	One and only one of these is required by the script. Use <i>/list</i> to target a list of computers contained within a text file. Use <i>/computer</i> to target a single computer. Use <i>/container</i> to target an organizational unit within Active Directory®.
<code>/computer:name</code>	
<code>/container:name</code>	
<code>/volume:Letter:\</code>	Specifies the volume on which to create a Shadow Copy. The volume name should be specified as a drive letter, followed by a colon and a backslash.

<code>/recurse</code>	When used with <code>/container</code> , also targets computers contained within nested OUs.
<code>/ping</code>	Verifies the connectivity to all targeted computers prior to attempting a connection. Using this argument will reduce the timeout wait when one or more computers cannot be reached on the network.
<code>/log:path</code>	Logs unreachable computer names to the specified file. This file can then be used later, along with the <code>/list</code> argument, to retry these computers. This argument works only when used in conjunction with the <code>/ping</code> argument.
<code>/verbose</code>	Causes the script to display more detailed, step-by-step status messages.

You can run this script with the `/?` parameter to display the command's syntax.

Under the Hood

This script uses the *FileSystemObject* to read computer names when the `/list` argument is specified and uses Active Directory Services Interface (ADSI) to retrieve computer names when the `/container` argument is specified. The script contains a subroutine named *WorkWithOU*, which calls itself recursively (that is, it calls itself over and over) to process nested OUs when the `/recurse` argument is specified. The real work of the script is performed by a relatively small section of VBScript code:

```
Dim oStorage, errResult, oWMIService
On Error Resume Next
Verbose " Attempting to set Shadow Copy on " & sName
Set oWMIService = GetObject("winmgmts:{impersonationLevel=impersonate}!\\" & sName &
"\root\cimv2")
If Err <> 0 Then
    WScript.Echo " *** Error connecting to " & sName
    WScript.Echo "      " & Err.Description
    LogBadConnect(sName)
Else
    Set oStorage = oWMIService.Get("Win32_ShadowCopy")
    If Err <> 0 Then
        WScript.Echo " *** Couldn't get Shadow Copy services from " & sName
        WScript.Echo "      " & Err.Description
    Else
        ErrResult = oStorage.Create(WScript.Arguments.Named("volume"), "ClientAccessible")
        Verbose " " & sName & ": " & errResult
    End If
End If
```

The current computer name is contained in the variable `sName`. Note that the script connects to the computer's `\root\cimv2` namespace in Microsoft Windows Management Instrumentation (WMI) and then retrieves the *Win32_ShadowCopy* class from that namespace. This class provides a method named *Create()* that accepts the volume

name and a parameter; the parameter indicates that the created Shadow Copy should be accessible to clients. In Windows Server 2003, only “*ClientAccessible*” is used for the second parameter.

If you use this tool with the */verbose* argument, you’ll see an error code number for each targeted server. A 0 (zero) means that the command completed successfully; anything else means it didn’t. The following list provides some of the common error codes:

- 1 Access denied
- 2 Volume not found
- 4 Shadow Copies not supported (generally for removable or FAT volumes)
- 9 Maximum number of shadow copies reached

Other numbers indicate general failures on the part of the Shadow Copy provider within Windows itself.

Troubleshooting

This script has the potential for three major types of errors. First, a remote computer won’t be available. Second, you won’t have permission to enable Shadow Copies. In either case, you’ll see an error message that alerts you to the problem for that computer. Make sure you have the appropriate permissions and that the computer is reachable via the network from the computer running this script. Unreachable computers can optionally be logged to a text file for a later attempt, and you can speed up the script by specifying the */ping* argument.

Third, it’s possible that an error will occur during the Shadow Copy enabling. In that case, an error code will be reported by the script, as just described.

To Learn More

- To learn more about the Shadow Copies feature, consult the Help and Support Center within Windows Server 2003.
- You can also read more about Shadow Copies at <http://www.microsoft.com/windowsserver2003/techinfo/overview/scr.mspx>.
- Other script samples dealing with Shadow Copy can be found in the TechNet Script Center at <http://www.microsoft.com/technet/scriptcenter/scripts/shadow/default.mspx>.

Inventory Shadow Copy



On the CD The sample script can be found on the CD that accompanies this book at `\Chap6\ShadowCopyInventory\ShadowCopyInventory.wsf`.

Operating System	Supported?	Prerequisites
Windows 2000 family	No	■ WSH 5.6 or later
Windows XP Professional	No	■ WMI
Windows Server 2003 family	Yes	■ Administrative permission on targeted computers ■ Network connectivity to each remote computer

Description

This tool lists all Shadow Copy information for the specified server or servers. For each Shadow Copy on a server, the tool specifically lists the following items:

- **Volume name** The volume on which the Shadow Copy exists.
- **State** The Shadow Copy's current status. The script will display the status as a text string. Not all volumes provide state information; a blank will be displayed when this is the case.
- **Exposed name** The file system name of the Shadow Copy when it is exposed to clients.
- **Count** The number of shadow copies in the Shadow Copy set.

Performing This Task Manually

Windows doesn't provide a detailed user interface that exposes the Volume Name, State, Exposed Name, and Count. You can use the Shadow Copies user interface (reachable by right-clicking a volume, selecting Properties, and selecting the Shadow Copies tab) to view available shadow copies on a given server and a given volume, but the status information is not available.

Example

You can use this tool in three basic ways. First, you can target a single remote computer named ServerA:

```
ShadowCopyInventory.wsf /computer:ServerA
```

Second, you can target a list of computers from a text file. The text file is expected to contain one computer name per line, with no other information in the file. Assuming the file is named `C:\Computers.txt`, you would use this syntax:

```
ShadowCopyInventory.wsf /list:C:\Computers.txt
```

Third, you can target an entire organizational unit of computer accounts. If your domain contains an OU named `West`, you would use the following syntax:

```
ShadowCopyInventory.wsf /container:west
```

Note that the `/container` argument will work only against the default domain of the computer running the script. In other words, the OU specified must exist within the same domain that the computer running the script belongs to. If the specified OU has nested OUs, you can include their computer accounts as well by specifying one additional argument:

```
ShadowCopyInventory.wsf /container:west /recurse
```

Additional arguments provide extra functionality to the command. See the following section titled “Syntax.”

Syntax

This script can be executed as a command-line utility. Set `CScript.exe` to be your default script processor, as described in Chapter 3.

<code>/list:path</code>	One and only one of these is required by the script. Use <code>/list</code> to target a list of computers contained within a text file. Use <code>/computer</code> to target a single computer. Use <code>/container</code> to target an organizational unit within Active Directory.
<code>/computer:name</code>	
<code>/container:name</code>	
<code>/recurse</code>	When used with <code>/container</code> , also targets computers contained within nested OUs.
<code>/ping</code>	Verifies the connectivity to all targeted computers prior to attempting a connection. Using this argument will reduce the timeout wait when one or more computers cannot be reached on the network.
<code>/log:path</code>	Logs unreachable computer names to the specified file. This file can then be used later, along with the <code>/list</code> argument, to retry these computers. This argument works only when used in conjunction with the <code>/ping</code> argument.
<code>/verbose</code>	Causes the script to display more detailed, step-by-step status messages.

You can run this script with the `/?` parameter to display the command’s syntax.

Under the Hood

This script uses the *FileSystemObject* to read computer names when the */list* argument is specified and uses Active Directory Services Interface (ADSI) to retrieve computer names when the */container* argument is specified. The script contains a subroutine named *WorkWithOU*, which calls itself recursively (that is, it calls itself over and over) to process nested OUs when the */recurse* argument is specified. The real work of the script is performed by a relatively small section of VBScript code:

```
Dim oWMIService, cCopies, oCopy, sState
Verbose " Connecting to WMI on " & sName
Set cCopies = QueryWMI(sName,"root\cimv2","Select * From Win32_ShadowCopy","","")
If Not IsObject(cCopies)
  WScript.Echo " *** Could not retrieve shadow copies from " & sName
Else
  WScript.Echo ""
  WScript.Echo "Shadow copies on " & sName
  WScript.Echo "Volume,State,Name,Count"
  For Each oCopy In cCopies
    Select Case oCopy.State
      Case 0
        sState = "Unknown"
      Case 1
        sState = "Preparing"
      Case 2
        sState = "ProcessingPrepare"
      Case 3
        sState = "Prepared"
      Case 4
        sState = "ProcessingPrecommit"
      Case 5
        sState = "Precommitted"
      Case 6
        sState = "ProcessingCommit"
      Case 7
        sState = "Committed"
      Case 8
        sState = "ProcessingPostCommit"
      Case 9
        sState = "Created"
      Case 10
        sState = "Aborted"
      Case 11
        sState = "Deleted"
      Case 12
        sState = "Count"
      Case Else
        sState = "*****"
    End Select
    WScript.Echo oCopy.VolumeName & "," & sState & "," & oCopy.ExposedName & ","
  & oCopy.Count
  Next
End If
```

The current computer name is contained in the variable *sName*. Note that the script connects to the computer's `\root\cimv2` namespace in Windows Management Instrumentation (WMI) and then retrieves the *Win32_ShadowCopy* class from that namespace. Next, the script lists specific properties of that class for each instance retrieved. You can modify this script, if desired, to list other properties of each Shadow Copy.

Also note that this script will output most of its information (assuming you are not using the `/verbose` argument) in a comma-delimited format. This allows you to run the script and capture its output to a text file, if you want, for use within a report. You can open the resulting file in an application such as Microsoft Excel to view the report in a columnar format. Use standard command-line redirection to capture the script's output to a text file.

Troubleshooting

This script has the potential for two major types of errors. First, a remote computer won't be available. Second, you won't have permission to enable Shadow Copies. In either case, you'll see an error message that alerts you to the problem for that computer. Make sure you have the appropriate permissions and that the computer is reachable via the network from the computer running this script. Unreachable computers can optionally be logged to a text file for a later attempt, and you can speed up the script by specifying the `/ping` argument.

To Learn More

- To learn more about the Shadow Copies feature, consult the Help and Support Center within Windows Server 2003.
- You can also read more about Shadow Copies at <http://www.microsoft.com/windowsserver2003/techinfo/overview/scr.mspx>.
- Other script samples dealing with Shadow Copy can be found in the TechNet Script Center at <http://www.microsoft.com/technet/scriptcenter/scripts/shadow/default.mspx>.