Realtime publishers
"Leading the Conversation"

August 2008
Volume 1, Number 7

Windows Administration

in Realtime

QUEST SOFTWARE®

Read an excerpt from the new Quest Software white paper *"Be the Master of Your Domain – Understanding Windows Server 2008 Active Directory Domain Services."*

—by Tony Murray.
Directory Services MVP

Microsoft recently announced the release of Windows Server 2008 RTM. Codenamed "Longhorn," this latest version of the server operating system from Microsoft marks a significant departure from its predecessors. This white paper introduces the changes made to Active Directory in Windows Server 2008 and addresses the impact for organizations with Active Directory already installed.

New Forest and Domain Functional Level

In Windows Server 2003, Microsoft allowed administrators to set the functional level of the domain or forest to a specific value, assuming certain conditions were met. The available AD functionality was determined by the functional level. For example, a Domain Functional Level of 2 (Windows Server 2003) permitted domain controller renames, updated and replicated last logon time stamp attribute and certain other features not available with other levels. Similarly, a Forest Functional Level of 2 allowed for cross-forest trusts, domain renames, and so on. Windows Server 2008 provides a new level: 3 (also known as Windows Server 2008).

Domain Functional Level 3

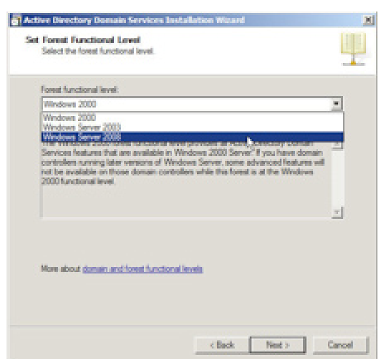
Domain Functional Level 3 provides the following features:

- All features from the Windows Server 2003 domain functional level
- Distributed File System Replication support for SYSVOL, which provides more robust and detailed replication of SYSVOL contents
- Advanced Encryption Services (AES 128 and 256) support for the Kerberos protocol
- Last Interactive Logon Information, which displays the time of the last successful interactive logon for a user, the workstation used, and the number of failed logon attempts since the last logon
- Fine-grained password policies, which make it possible for password and account lockout policies to be specified for users and global security groups in a domain

Forest Functional Level 3

Forest Functional Level 3 provides all the features available at the Windows Server 2003 forest functional level, but no additional features. The sole purpose of raising the forest functional level to 3 is to prevent any new downstream domains or domain controllers from being joined to the forest.

One other item to note is that Windows Server 2008 domain controllers can be added to domains at Functional Level 2 (i.e., Windows Server 2003). If your organization has an urgent need for read-only domain controllers (RODCs), you can deploy them into your existing Windows Server 2003 forest without having to first upgrade your existing domain controllers.



Read-only Domain Controller (RODC)

The introduction of the RODC in Windows Server 2008 may represent the biggest change to AD since its Windows 2000 inception. The RODC is intended to reduce the risk of security compromises in locations where the threat is highest (such as a perimeter network) or where the physical security of the domain controller is not optimal (for example, a branch office). Unlike standard domain controllers (called writable domain controllers), the RODC does not replicate any changes to other domain controllers. This means that an attacker cannot use a compromised RODC to gain control of the forest by replicating permissions or schema changes. An attacker would be limited to using a compromised RODC to gain access to data held within the local credential cache. To reduce this risk, administrators have the ability to configure the RODC to cache only the password hashes of the accounts that will actually use the RODC for authentication.

The risk of compromises can be reduced even further by installing the RODC in combination with the Server Core version of Windows Server 2008. This effectively lowers the surface area for attack and reduces the patching requirements.

The RODC also supports the filtered attribute set, a new feature that enables administrators to define a set of attributes with values that do not replicate to RODCs in the forest. An example would be an application that uses certain attributes to store credential information for authentication to the application. If these attributes are added to the filtered attribute set, their values are replicated between writable domain controllers as normal, but they are not replicated to your RODCs.

About Quest Software, Inc.

Quest Software, Inc. delivers innovative products that help organizations get more performance and productivity from their applications, databases, Windows infrastructure and virtual environments. Through a deep expertise in IT operations and a continued focus on what works best, Quest helps more than 90,000 customers worldwide meet higher expectations for enterprise IT. Quest Software can be found in offices around the globe and at **www.quest.com**.

Want more? Read the entire white paper at **www.quest.com/GetActive**

Letter from the Editor

Saying Thanks for a BSOD at 35,000 Feet

by Greg Shields

Last February, the wife and I took a trip over the pond to do some skiing and visit some friends in Chamonix, France. Having saved up a boatload of frequent flyer miles over the past few years, we'd squeaked out just enough to get us two tickets in Business Class on Lufthansa. Let me first tell you that doing a 10-hour flight like that one in B-Class is really the only way to fly. They give you this little menu when you first sit down with all your food options, and I swear there's a wine cellar in the belly of the plane. Best of all, you get this nifty seat with a remote control that enables it to move in all sorts of directions, all the way down to a sort of quasi-bed for sleeping after the flight attendant brings by the last liquor cart of the night.

Another bonus is this little entertainment center in the seatback in front of you. This little device plays movies, video games, and will even give you some lessons on the German language if you want (Lufthansa being a German airline, this makes sense). But while we had a relatively event-free trip on our way over, the return flight was another story.

About halfway through the return flight, just as we were crossing over what I think was Iceland, the entire airplane…well…blue screened. Black screened to be precise. Turns out that much of the functionality of the plane's cabin systems, as well as those nifty chairs, actually runs off of a version of Windows CE. Something about our flight, the phases of the moon, sunspot activity, and Jupiter rising in the house of Venus convinced every single one of our Windows-based airplane systems to spontaneously bug check, all at the same time. Lights go out, movies stop, people get freaked out, and everyone at once looks out the window with a shudder.

Now I've been doing Windows administration for nearly 15 years, and in that time I've seen my fair share of BSODs, both black and blue. But this one struck a chord within me. In our standard office environments, we've pretty much got it made with all the resources we need to fix problems and update computers. We've got the Internet for seeking out answers. We've got WSUS for quickly installing patches. We've got plenty of tools at our disposal for resolving issues as they arrive.

But when it comes to airplanes over the Atlantic, or deep space probes, or people working in Antarctica or the bottom of the ocean, there is no such infrastructure. Windows is still Windows, which means it decides to go down at inopportune times, no matter whether it's convenient or not. We may do a lot of complaining that Vista isn't XP, or that Microsoft doesn't understand us, or that the network is slow today. But we IT administrators in the typical office environments of the world still have a lot to be thankful for.

That plane stayed in the air, thankfully. Eventually it rebooted itself and I finished the rest of my movie. But that thought of thankfulness for all the infrastructure that supports our chosen career continues in my mind today.

While you're experiencing your own Zen moment in the world of IT, may I suggest a look at some of the great features we've got on deck for this month. Richard Siddaway's feature on snapshot backups in Server 2008 will help ease those nasty restores. Jeff Hicks discusses Event Log management using PowerShell. Don does a review on Camtasia, while Peter talks about hosted Exchange services. ◆

Answers from the Experts

Approval of AD User Account Creation

by Don Jones

Q: Dave, who's having a tough time making his Active Directory (AD) management line up to his company's ITIL-based business processes, sent in this month's question: Our flowcharts require a senior administrator to approve all new user account creation before they actually go into AD. What's the best way to actually enforce that?

A: You're talking about workflow and change control here, and it's a tough topic. Microsoft certainly doesn't have any kind of built-in functionality for this, although if you move to their Identity Lifecycle Manager (ILM) server product, you do gain the capability to add business rules and workflow to identity management. It's possible to add this kind of functionality in a VBScript or Windows PowerShell script, but doing so comes with problems. First, you'd have to agree to only do AD management via those scripts—which I guarantee won't be universally popular—and there's no real way of enforcing those scripts as the only means of managing the directory. In other words, anybody could simply use Active Directory Users & Computers (ADUC) whenever they wanted to, and the scripts wouldn't be able to do a thing about it.

Commercial tools from third parties offer the functionality you're after and then some. NetIQ's Directory & Resource Administrator (DRA), Quest's ActiveRoles Server (ARS), and NetPro's Access Manager (AM) all have various built-in capabilities for improving directory management, including workflow capabilities. DRA and ARS achieve these through a proprietary UI that's basically intended to replace ADUC; AM extends ADUC so that you still work inside that tool but pick up workflow and change control capabilities. These products are licensed on a per-directory-user basis, and run from $12 to $25 per user plus yearly maintenance. It's important to note that these tools all provide capabilities for centrally managing permissions on files, folders, and other resources across the enterprise, and those permission management features drive a lot more of the cost than the AD workflow features.

If you're just after workflow in AD, there's a free tool that might interest you: the Active Directory Management Console (ADMC) available from www.turbochargead.com. It provides customizable workflow and business rule enforcement for AD, extends ADUC (so you're working in the same tool you probably use today), and does pretty much what Dave asked for—provides a way to prevent new objects from being created without approval. As of this writing, the product is in "Community Preview" status but should be launching by the time you read this and is slated to remain free after its launch. One of its biggest benefits over a script (aside from the time it saves you, since you don't have to actually write a script) is that it does live right within ADUC, and it uses the native AD data to perform its tasks. There's no separate database or anything, so there's really no easy way to bypass the tool and the workflow it's enforcing.

Do you have an IT question you'd like Don to answer? Send it to answers@realtimepublishers.com for consideration!

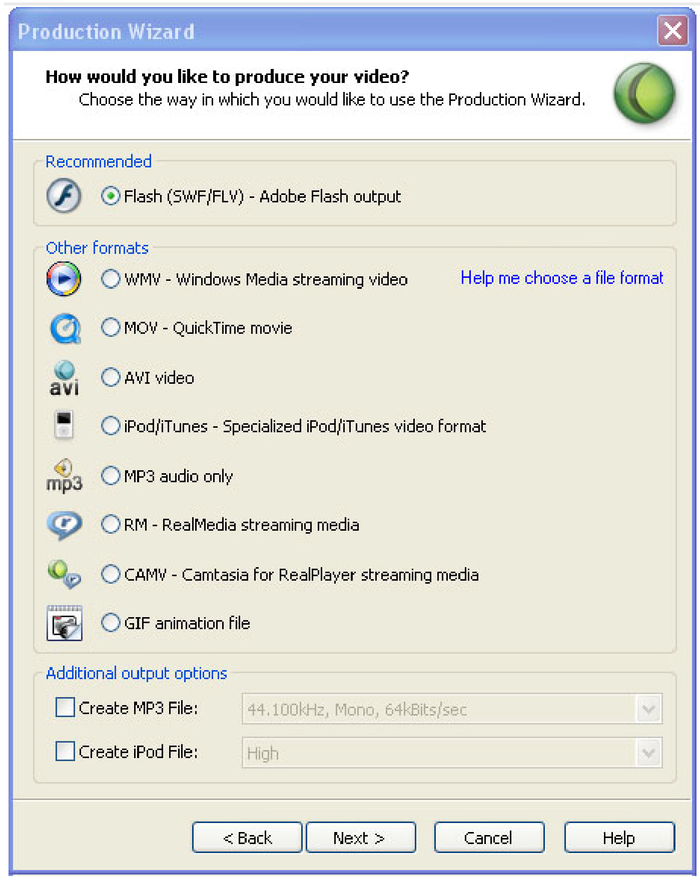
Don Jones is a series editor for Realtime Publishers. He's the author of more than 30 IT books, including Windows PowerShell: TFM; VBScript, WMI, and ADSI Unleashed; Managing Windows with VBScript and WMI; *and many more. He is a multiple-year recipient of Microsoft's "Most Valuable Professional" (MVP) Award with a specialization in Windows PowerShell.*

Product Review

Camtasia Studio 5

by Don Jones

Unless you're a trainer or conference presenter, you may have never thought of using TechSmith's Camtasia software, which allows you to capture the action from your computer screen, edit and annotate it, and then produce high-quality videos for distribution in various formats. I'm here to tell you, though, that Camtasia is something nearly every administrator should be using. Why?



Have you ever had to demonstrate how to accomplish some administrative task to a junior admin? Have you ever needed to walk a user through a particular troubleshooting or repair task? Have you ever rolled out a new product that was completely unfamiliar to your users—Office 2007 pops into my mind—and wished you could quickly show them how to accomplish once-familiar tasks? These are all scenarios where Camtasia shines. Simply hit "Record," walk through the task, and your demonstration is saved for all time as a convenient video file.

Camtasia Recorder is a small program that actually performs the screen recordings, by default using an encoding package called the TechSmith Screen Capture Codec (TSSC), which is specifically designed to capture high-quality recordings of onscreen action. You can elect to narrate your recordings as you go, so it's just like you're sitting next to the viewer, walking them through whatever process you've recorded.

Camtasia Studio itself is an editing environment, where you can edit out mistakes, re-record narration if needed, add callouts and annotations to reinforce onscreen action, and so forth. Simple controls allow you to work in a timeline view, add transitions and annotations, zoom in to particular parts of the video (which is great when you want to focus the viewer on a particular piece of action), and so on. You can even embed closed captioning to make your final video more accessible to a wider audience.

When you're finished, Camtasia can produce your video in a variety of formats, including QuickTime, AVI, Windows Media, Flash, and more, as Figure 1 shows. A wizard suggests an appropriate format for whatever means of distribution you plan to use, such as Web, CD, email, and so forth. One of the best packaging options is the standalone executable, which creates a single EXE file that includes your movie and a custom player. The advantage of this package is that it also includes the TSSC software needed to view your recordings; if you export to some other formats (QuickTime, WMV, etc.), your users will first have to install the TSSC from TechSmith's Web site (a version called Ensharpen is available for Mac users; both codecs are free to install).

A downside of Camtasia is that it isn't supported on server operating systems (OSs) and flat-out won't run on Windows Server 2008 (as of this writing). That restricts your ability to create short instructional videos on those OSs; the only workaround is to run your server OS in a virtual machine (such as Microsoft Virtual PC or VMware Workstation), run Camtasia on a supported client OS (WinXP or Vista), and capture the contents of the virtual machine window. It's a bit awkward, and prevents you from capturing tasks performed on a production server.

That limitation aside, Camtasia is a good tool for quickly producing short instructional videos that are easily shared. Its rich toolset allows you to go much further, producing complicated Flash animations that include interactivity and even short quizzes, although the learning curve for those additional features is a bit steeper than the basic record-edit-produce workflow.

See www.techsmith.com for product information and trial software.

Snapshot Backup in Windows 2008

by Richard Siddaway

Active Directory (AD) has become a fundamental infrastructure component for environments based on Microsoft technologies. The technology was introduced with Windows 2000. Upgrades and new functionality have been released with Windows 2003 and Windows 2003 R2. The latest version of the Windows operating system (OS), Windows Server 2008, introduces further enhancements that enable administrators to manage the AD environment in a better and more granular fashion.

In my article in the April 2008 issue, I described Granular Password Policies, which were introduced to AD with Windows Server 2008, and showed how they could be administered. In this article, I will describe another new feature in Windows Server 2008 AD that aids managing and protecting AD: Snapshot Backup. I will put this feature into context by discussing how it relates to backing up AD in general and the changes that have happened to AD backup in Windows Server 2008.

Snapshot Backup is managed from the command line rather than a GUI. I will show the commands required to create and manage Snapshot Backups. Backup should never be regarded as an activity in its own right. As administrators, we only perform backups so that we can restore our systems if something goes wrong, so I will also show how to use Snapshot Backup to recover objects that have been deleted from AD.

Some organizations have implemented lag sites in previous versions of Windows. These are sites that are only replicated to on a very limited basis, often only once a week. With two lag sites, the replication to the sites can be staggered by half a week. This gives you the capability of performing an authoritative restore from the lag site in the event that objects are deleted that you need to recover as long as you do it within the lag period. Creating lag sites requires additional hardware purchase and software licenses together with extra work for the administrators. Using Snapshot Backups instead of lag sites has merit as a method of reducing the expenditure on hardware, software, and administration. It will also optimize AD management.

AD Tombstones

One issue that complicates backup and restore of AD is the tombstone interval. When an object is deleted, it does not instantly vanish from the AD data store. Most of the attributes are stripped from the object, and it is moved to the deleted objects container (CN=Deleted Objects, DC=domain_name). It will remain in the deleted objects container for a number of days depending on the version of Windows used to create the forest.

Tombstone Period	Windows Version
60 days	Windows 2000 Server Windows Server 2003 Windows Server 2003 R2
180 days	Windows Server 2003 SP1 Windows Server 2003 SP2 Windows Server 2008

Note that the tombstone period is set when the forest is created. It is not altered by applying service packs or upgrading to a later version. It is possible to manually modify the tombstone period. Once the tombstone period has expired, that object is finally deleted from AD.

Backups are valid for the length of the tombstone period. If you were to restore an object from a backup that was older than the tombstone period, you would restore objects that had been completely deleted. This would create an object that other Domain Controllers knew nothing about, as they had deleted it. This creates an inconsistency in AD that will have to be manually addressed.

In Windows 2000, the only way to recover a deleted object was to perform an authoritative restore. Windows Server 2003 introduced the ability to restore or re-animate a tombstone object. Unfortunately, the re-animation process does not restore the attributes that were lost when the object was originally deleted. Those attributes have to be recreated from another source, which often means typing them in manually.

To restore an object, especially in Windows 2000, you would need to perform an authoritative restore; otherwise, when the object was restored, AD replication will result in the object(s) being deleted again. This is still the optimum way to restore a set of objects simultaneously (for example, if a careless or rogue administrator deletes an OU full of users). Performing an authoritative restore requires the Domain Controller to be rebooted into Directory Services Restore Mode. The AD database is restored, and the objects to be recovered are marked as authoritative using ntdsutil. The Domain Controller is then rebooted into normal operating mode and AD will replicate the restored objects to other Domain Controllers.

NTBackup

In Windows 2000 and Windows 2003, AD backups were taken as part of a System State backup. It was not possible to take a backup consisting of only AD. NTBackup, or a third-party backup utility, was used to create the System State backup. System State, on a Domain Controller, consists of:

- AD database
- SYSVOL
- Registry
- Boot files and protected Windows files
- COM+ database

Other data is included if the Domain Controller has multiple roles installed.

The AD database could be restored to another location to be used in Domain Controller creation via the Install From Media option. This enables the AD database to be pre-populated, from the restored file, which drastically reduces the time needed for AD replication to bring the database up to date. For this purpose, the more recent the backup, the better.

Windows Server 2008 Backup

In Windows Server 2008, the rules change. NTBackup does not exist either as a base part of the OS or as an installable feature. It has been replaced by Windows Server Backup, which is an installable feature. If you also install the command-line and scripting tools, you get a command-line tool called wbadmin.exe and a PowerShell snap-in that can be used to administer backups. Windows Server Backup and its associated tools are not the focus of this article, so I will not discuss its operation in detail.

Using Windows Server Backup does have a number of implications for AD backup and restore. The utility performs Volume Shadow Copy Service (VSS) backups rather than the file-based backup of NTBackup. Thus, only full complete volumes can be backed up. You cannot use it to back up only System State, individual folders, or files. Backups are only performed to disk or DVD. Direct backups to tape are not supported. However, the restore operation is more granular, enabling

- Full server restore
- Full volume restore
- System State only restore
- Folder or file restore

The ability to generate a copy of the AD database to use during Domain Controller creation has not gone away. It has been moved into ntdsutil. Using the IFM subcommand, a file is created that can be used to create either a full or read-only Domain Controller. Both of those options can be created with or without the SYSVOL data.

Snapshot Backup

Having examined some of the issues around backing up and restoring AD, let's turn our attention to Snapshot Backups, which are a new feature in Windows Server 2008. Snapshot Backups take a point-in-time copy of the AD database. The first backup is a full backup, but subsequent backups are incremental. The backups are stored in the System Volume Information Folder, so you will need to ensure that there is sufficient room on the volume for the backups. It is possible to remove snapshots from the volume, and it should be remembered that the tombstone period of 180 days still applies.

The great advantage of Snapshot Backups is that you can mount them as an instance of AD Lightweight Directory Services (AD LDS) side by side with the currently running AD. This means you can use the normal administration tools (for example, Active Directory Users and Computers) to view the mounted snapshot and another instance of Active Directory Users and Computers can be used simultaneously to view and work with the live AD.

There are a number of scenarios where Snapshot Backups are invaluable:

- Find the old value of an attribute to back out a change made to an AD object. If the new auditing option of capturing Directory Service Changes is enabled, the event logs will capture the before and after values. However, depending on the log size, retention periods, archiving policy, and audit settings, the information may not be available through the event logs.
- Repopulate the attributes on an object that has been reanimated after deletion stripped off all the data attributes associated with the object.
- Compare the current values of particular attributes with those stored in the snapshot to determine a past value. The intention may not be to restore or reset the data but just to view the settings at a particular point in time. One use for this is to check whether a user was in a particular group at that time and therefore had access to a specific file or folder.

Snapshot Creation

Snapshot Backups are created using ntdsutil. Given this additional functionality as well as the creation of AD backups for use when creating a Domain Controller via the Install From Media option plus all the previously available functionality, ntdsutil really is a utility that administrators must understand how to use. The full syntax of ntdsutil is explained in the help system available with Windows Server 2008.

The syntax to create a Snapshot Backup is relatively straightforward. I run most of the commands from a PowerShell console rather than a command prompt, but they work equally as well from a command prompt. In the following examples, I show the ntdsutil syntax as a single command. It is possible to work with ntdsutil in an interactive manner, but I have chosen to present it in this way for brevity. In addition, the commands presented in the article can be used directly in a command file, which makes automation of the activity easier. This helps to optimize the management of your systems.

I tend to run the PowerShell or command prompt with elevated privileges (that is, Run as Administrator) when working with Snapshot Backups. If you try running without prior elevation, you will be prompted for the required level of privileges each time the commands are run.

To create a Snapshot Backup, use this command:

```
ntdsutil "activate instance ntds" snapshot create quit quit
```

When the snapshot has completed, you will see a message similar to

```
Snapshot set {9bd853c8-d95b-4308-8ea0-ac93638aab7d} generated successfully.
```

Snapshots are identified and managed by their GUIDs—so be prepared to be working with GUIDs quite a bit.

The first time a Snapshot Backup is taken on an individual Domain Controller, it results in a full backup of the AD database being generated. Note that it is only the contents of the AD database that are backed up. SYSVOL and other data that would be backed up under a System State backup are totally ignored by Snapshot Backups. Subsequent backups are incremental.

Viewing Snapshots

The backups produced by running the previous command are stored in the System Volume Information folder. This is a protected folder. In order to see what snapshots are available, you need to go back to ntdsutil and use this command:

```
ntdsutil snapshot "list all" quit quit
```

The results from listing the snapshots are as shown below.

```
1: 2008/02/05:19:23 {4f1e07fd-629c-4aae-ba9a-fff20c654d4b}
 2:  C: {3ec8430e-9b56-45df-b361-19cff4a33493}

 3: 2008/02/09:12:03 {1818597f-d8c4-4224-812c-e7f963ecdb02}
 4:  C: {c2001e07-0853-4461-acd4-f4bd0be44927}

 5: 2008/06/08:17:14 {1fbd5b9e-f407-4ff9-ad81-0eac47c129c4}
 6:  C: {7dcb5a49-9737-4a0c-8499-193942667539}

 7: 2008/06/08:17:15 {9bd853c8-d95b-4308-8ea0-ac93638aab7d}
 8:  C: {a6e87a73-8386-4e6c-8b7f-a241cea13776}

 9: 2008/06/08:17:17 {5f18cd9e-8517-445a-99ed-f195ef08f39d}
10:  C: {0b5fa6c4-6a6c-4546-9a4c-1f51d901ba3d}
```

Note that there are two lines for each backup. The first line gives the date and time the backup was produced as YYYY/MM/DD:HH:MM. The second line contains the GUID of the backup.

You can view the System Volume Information folder by performing the following at a command prompt:

```
dir /AS
cd “system volume information”
dir /AS
```

This gives the following results:

```
05/02/2008 20:23      65,536 {3808876b-c176-4e48-b7ae-04046e6cc752}
05/02/2008 22:03    38,141,952 {81d5c001-d41d-11dc-9a36-0003ff5912e9}{3808876b-c176-4e48-
b7ae-04046e6cc752}
08/02/2008 22:10    99,368,960 {81d5c00b-d41d-11dc-9a36-0003ff5912e9}{3808876b-c176-4e48-
b7ae-04046e6cc752}
09/02/2008 13:03    60,473,344 {361dbb71-d67f-11dc-ac3e-0003ff5912e9}{3808876b-c176-4e48-
b7ae-04046e6cc752}
12/02/2008 22:03          0 MountPointManagerRemoteDatabase
08/06/2008 17:14    945,717,248 {43c04911-d6f8-11dc-94e0-0003ff5912e9}{3808876b-c176-
4e48-b7ae-04046e6cc752}
08/06/2008 17:15    26,755,072 {7b36b0b7-3548-11dd-bf00-0003ff5912e9}{3808876b-c176-4e48-
b7ae-04046e6cc752}
08/06/2008 17:16   2,831,155,200 {7b36b0bf-3548-11dd-bf00-0003ff5912e9}{3808876b-c176-
4e48-b7ae-04046e6cc752}
08/06/2008 17:17    24,838,144 {7b36b0bb-3548-11dd-bf00-0003ff5912e9}{3808876b-c176-4e48-
b7ae-04046e6cc752}
```

Mounting a Snapshot

Having identified the snapshot you need to work with, the next task is to derive the database path so that you can mount the snapshot. The snapshot is mounted side by side with the live AD database. You do this by taking the date the snapshot was created and coming up with a database path similar to the following, which is taken from line 3 in the listing of available snapshots:

```
C:\$SNAP_200802091203_VOLUMEC$\Windows\NTDS\ntds.dit
```

The only variable part of the database path is derived from the date the snapshot was created. Having derived the path to the database, you can then mount the database to make it usable. The GUID is contained on the line following the date—line 4 in the list of snapshots given earlier:

```
ntdsutil snapshot “mount {c2001e07-0853-4461-acd4-f4bd0be44927}“ quit quit
```

You now have to make this database accessible via LDAP. To do so, use the dsamain (directory services maintenance) utility:

```
dsamain -dbpath: C:\$SNAP_200802091203_VOLUMEC$\Windows\NTDS\ntds.dit
-ldapPort: 60000 -sslPort: 60001 -gcPort: 6002 -gcSslPort: 60003
```

This would be input on one line but is split here for readability.

In this example, the database path is derived as explained earlier. The other parameters refer to the port used for LDAP connectivity, the port used for encrypted (SSL) LDAP connectivity, the port used for Global Catalog connectivity, and the port used for encrypted (SSL) Global Catalog connectivity, respectively. The port numbers are arbitrary. Chose ports that you know are not in use and not likely to be used with your organization. High ports such as 60000 and above are usually good candidates, but check that they are not being used by other applications.

I normally run this command in a command prompt with elevated privileges. I use a command prompt as it is easy, on my systems, to distinguish between a command prompt and a PowerShell prompt due to the background colour. Also, once you have mounted the database using this command, you cannot perform any other commands at that particular prompt until you dismount the database. You can see the mounted database in Windows Explorer.

Using a Snapshot

Having mounted the snapshot, how do you work with it? The snapshot is mounted as an instance of Active Directory Lightweight Services (ADAM). This means you can access it with normal LDAP tools or scripts. You can point Active Directory Users and Computers at the snapshot by right-clicking Active Directory Users and Computers [FQDN of Domain Controller] and selecting Change Domain Controller. Over type the <Type a Directory Server name[:port] here> entry with the FQDN of the Domain Controller on which the snapshot is mounted and the port number, as the following example illustrates:

```
DC02.Manticore.org:60000
```

Figure 1 shows the live and snapshot versions of the AD database being accessed via Active Directory Users and Computers.

In the case of an object that still exists in the live AD, you can access it and the version in the snapshot. Settings can be compared using Active Directory Users and Computers and manually modified in the live database as required.

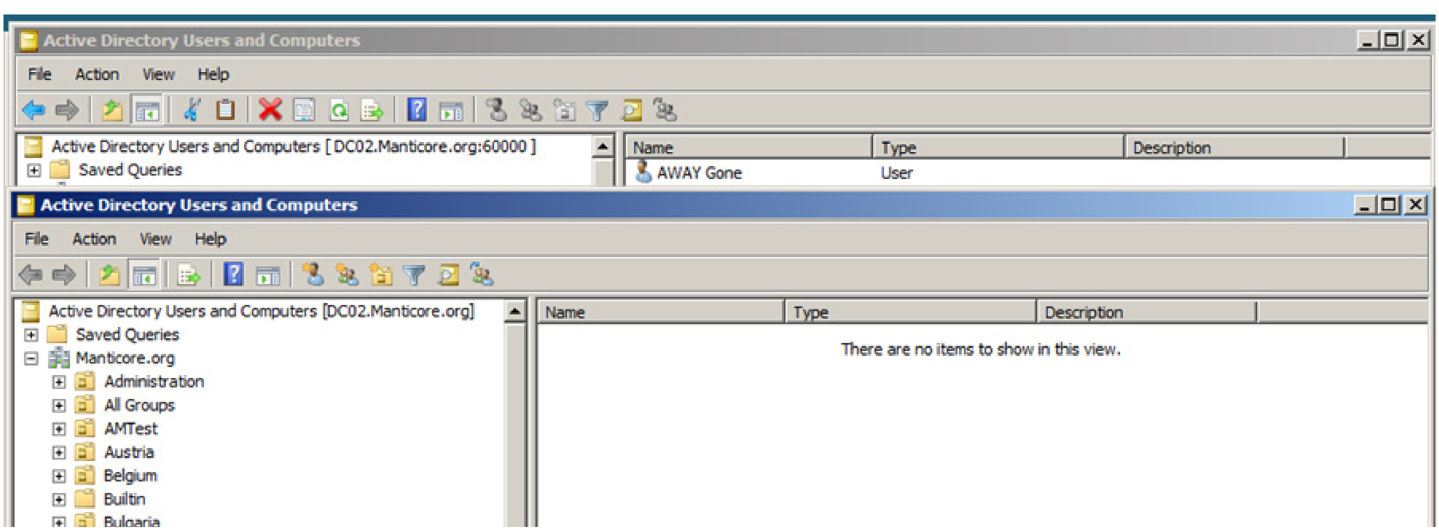


Figure 1: Two instances of Active Directory Users and Computers. The front one shows the live AD with the user from the Test OU having been deleted. The instance at the rear shows the snapshot with the user account available.

Recovering an Object

If you want to restore a deleted object, you need to do a little more work. With the introduction of PowerShell and access to .NET System.DirectoryServices.DirectorySearcher, you can view the tombstone records directly (this was not possible using VBScript). However, reanimating a tombstoned object is more difficult. You have to use the System. DirectoryServices.Protocols namespace, which is difficult to work with, not very well documented, and has a limited set of examples. This situation has changed for the better recently with the release of the SDM AD Tombstone Reanimation Cmdlets, which give us the capability to view the tombstone record and to restore a tombstoned object. The cmdlets can be [downloaded](). If you have any difficulty in installing or accessing the cmdlets, see my [blog]() for more details.

To view the tombstoned objects you use

```
Get-SDMADTombstone
```

Be aware that this will show ALL tombstoned records. In a large and busy AD, this could be a very significant number of records. To reduce the number of records returned, you can utilize the –Filter parameter, as the following example shows:

```
PS> Get-SDMADTombstone -Filter Gone


CN          : AWAY Gone
           DEL:6d8667b3-7019-48e7-81f7-9575952be33c
DistinguishedName : CN=AWAY Gone\0ADEL:6d8667b3-7019-48e7-81f7-9575952be33c,CN=Deleted
Objects,DC=Manticore,DC=org
Name         : AWAY Gone
           DEL:6d8667b3-7019-48e7-81f7-9575952be33c
WhenCreated     : 05/02/2008 21:02:34
WhenChanged     : 09/06/2008 08:43:47
Description     :
LastKnownParent   : OU=Test,DC=Manticore,DC=org
```

If more than one record is returned, keep refining the filter until you get just the required record returned.

To restore the deleted record, you use the other cmdlet supplied in the snapin:

```
PS> Get-SDMADTombstone -Filter Gone | Restore-SDMADTombstone -WhatIf
What if: Performing operation " To be Undeleted: " on Target "CN=AWAY Gone,OU=Test,DC=Man
ticore,DC=org".
```

In this case, I have used PowerShell's -WhatIf parameter to double check the restore operation. When you are sure that you have identified the correct object to restore, simply remove the –WhatIf parameter:

```
PS> Get-SDMADTombstone -Filter Gone | Restore-SDMADTombstone

Confirm
Are you sure you want to perform this action?
Performing operation " To be Undeleted: " on Target "CN=AWAY Gone,OU=Test,DC=Manticore,DC
=org".
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "Y"): Y


RequestId   :
MatchedDN   :
Controls    : {}
ResultCode  : Success
ErrorMessage :
Referral    : {}
```

Notice that you are asked to confirm the action as a final check. The object is restored to the container referenced in the tombstoned object's LastKnownParent property. There is no way to change the restoration target container. It is possible to restore multiple objects simultaneously as long as the filter setting is correct. If the OU and the leaf objects have been deleted, the OU has to be restored before any of the leaf objects.

The restored object has minimal attributes, as the majority of the data was removed when the object was deleted. To make the object usable, you need to repopulate the attributes from your Snapshot Backup.

Repopulating an Object

I have chosen to script the repopulation of the object attributes. The following code is based on an original script by Guido Grillenmeir.

```
## set DC and snapshot
$newdn = "CN=AWAY Gone,OU=Test,DC=Manticore,DC=org"
$ADSnapShot = "dc02.manticore.org:60000"
$ProductionAD = "dc02.manticore.org"

# get directory entry for user in production AD
$adsPath = "LDAP://$ProductionAD/" + $newdn
$user_prod = [ADSI]($adsPath)

# get directory entry for user in AD snapshot
$adsPath = "LDAP://$ADSnapShot/" + $newdn
$user_snap = [ADSI]($adsPath)
```

```
# write data to user in production AD
$user_prod.sn         = $user_snap.sn
$user_prod.givenName     = $user_snap.givenName
$user_prod.SamAccountName  = $user_snap.samAccountName
$user_prod.UserPrincipalName = $user_snap.UserPrincipalName
$user_prod.DisplayName    = $user_snap.DisplayName
$user_prod.streetAddress   = $user_snap.streetAddress
$user_prod.l          = $user_snap.l
$user_prod.c          = $user_snap.c
$user_prod.CommitChanges()

## repopulate group membership
foreach ($groupDN in $user_snap.memberOf)
{
   $group = [ADSI]("LDAP://$ProductionAD/" + $groupDN)
   $group.Add("LDAP://"+$newdn)
   $group.CommitChanges()
}

## reset password
$user_prod.SetPassword("4rEA11yC0mpl3xPwd")
$user_prod.CommitChanges()

## set user must change password at next logon
$user_prod.pwdLastSet = 0
$user_prod.CommitChanges()

## enable account
## reset account control
$user = [ADSI]("LDAP://$newdn")
$user.useraccountcontrol = 512
$user.commitchanges()
```

You start by defining the distinguished name of the object to be repopulated and the LDAP connection strings for the live AD and the snapshot. You then create a directory entry for the restored object and the object in the snapshot. It is a simple matter to then copy the required attributes into the restored object and use CommitChanges() to write the data back into the live database.

The group information shown in the memberof property is created as back links from the group objects themselves. This means you have to access each of the groups that the user is a member of and add the user back into the group. This can be achieved using the simple loop shown in the code where you loop through the groups of which the user is a member and create a directory entry pointing to the group. You then add the user into the group and commit the changes.

When the user object is restored, it does not have a password set and it is restored in a disabled state. To make the object usable (that is, the user can log on), you need a few final steps. You need to set a password as shown. In a production environment, you could use Read-Host and get the password entered as a secure string for additional security. This would mean you would not have to put the password explicitly into the script. Having set a password, you need to force the user to change the password at next logon, and you finally enable the user account so that the user can log on.

It is possible to change the list of attributes that are preserved when an object is deleted to include the password. Doing so would enable us to restore the user with the same password as prior to deletion. It is not usually recommended to make this change for security issues. However, if EFS is used within the organization, forcing a change on the user's password in the manner shown in this article will break the link between the user and the encrypted files. It will be necessary to recover access to the EFS encrypted files using the standard techniques.

Once you have finished working with the snapshot, you need to dismount it. You can close LDAP access to the snapshot by CRTL-C in the command prompt in which you ran dsamain. To complete the dismounting of the snapshot, you use ntdsutil again:

```
ntdsutil snapshot "unmount *" quit quit
```

You now have only your live AD running on the Domain Controller.

Managing and Deleting Snapshots

As with all backup systems, you need to spend a little time managing the system. The commands I have shown were all used interactively. In a production environment, you would want to automate the production of snapshots. You can use the Task Scheduler in the Computer Management MMC to create a task that will run a snapshot at a pre-determined time. This task can be run daily or weekly depending on your requirements. It is still possible to then run a manual task if required (for example, just before making a lot of changes to AD).

When you want to trim the number of snapshots stored on disk, use the list command to display the available snapshots:

```
ntdsutil snapshot "list all" quit quit
```

Select the snapshot to delete, and use this syntax to perform the deletion:

```
ntdsutil snapshot "delete {c2001e07-0853-4461-acd4-f4bd0be44927} " quit quit
```

Again, you are working with the GUID to identify the particular snapshot to delete. Note that when you use this command, there is no confirmation or warning issued. The deletion will just happen.

Summary

Snapshot Backups are a new AD feature introduced with Windows Server 2008. They enable recovery of AD objects to a point in time without performing an authorative restore. Snapshot Backups should not be regarded as a replacement for complete backups that you would use for Disaster Recovery or Business Continuity purposes. They are another option that adds further flexibility to your backup and recovery regime and enables a more granular restore capability than previous versions of AD. By combining them with the new Protection from Accidental Deletion feature of Windows Server 2008 AD, you have the capability to significantly increase the level of protection given to your AD installations and therefore to the organization's reliance on those systems.

Note: All scripts were created and run using PowerShell v2 CTP 2.

Richard Siddaway is Microsoft Practice Leader for Centiq Ltd, a Microsoft Gold partner specialising in optimisation, measurement, management and migration involving Microsoft technologies. With over 20 years experience in various aspects of IT Richard is currently concentrating on the Microsoft environment at an architectural level especially around Active Directory, Exchange, SQL Server and Infrastructure Optimisation. Richard is a PowerShell MVP who founded and currently leads the UK PowerShell User Group. Richard has presented to the Directory Experts Conference 2007, at various events at Microsoft UK, and for other UK User Groups. Richard can be contacted through email at RSiddaway@centiq.co.uk or via his blog at http://richardsiddaway.spaces.live.com/.

The Deep Dive

Small Business Server 2008: A Fresh Perspective

by Greg Shields

So, they sent an entire computer…

And not just any old computer you might find lying around the back of someone's cubicle, but a brand new, 50-pound, beast of a Supermicro server-class machine.

But wait. A little background first. Not long ago, I was approached by Microsoft to do a review of the new Small Business Server (SBS) 2008 to be released relatively soon. This update to SBS first and foremost switches the core operating system (OS) from Windows Server 2003 to Windows Server 2008, but also includes some new features and upgrades goodies. Having relatively little experience with SBS 2003, I figured this was a great opportunity for a fresh take on this new solution from the perspective of a relative SBS newbie.

We first sat down for a 2-hour presentation between me and a number of teams at Microsoft. The presentation was interesting, discussing the changes to their interface between the two versions and what to expect out of the Server 2008 version of this SMB-friendly solution. But then, a couple of days later, I get a knock at the door. On the other side is a weary and overloaded delivery person from Southwest Airlines dragging in an impossibly large box filled with this humongous server. Turns out that for this review, Microsoft somewhat astutely wanted the review to happen from the perspective of the quasi-technical small business IT person as they unwrap their newly purchased, pre-built, and pre-installed SBS server. Nice touch.

Taking the time to unpack this behemoth, I plugged it in and booted it up. The server starts with some initial installation and configurations before prompting with the standard Windows Server 2008 WinPE setup screen. There is the opportunity to select the typical settings such as country, time and currency, keyboard, product key, and license terms. After a bit more waiting, you're taken to the wizard for setting up SBS 2008.

> One interesting component of the installation at this point is that if you already have an Active Directory (AD) domain in your network, the wizard mentions that, "…you must use an answer file to start the migration." For what has so far been a relatively easy install, perfect for that quasi-technical IT person, answer files are a much more difficult topic. If you're just this person in this situation, consider checking out http://go.microsoft.com/fwlink/?LinkID=102937 for more information about this potentially complex process.

The wizard then asks for common startup elements such as clock and time zone, downloading any required updates, company information, server name, domain name, and administrator account information. Once completed here, the SBS installation begins. The wizard and accompanying paperwork mentions that the process that can take upwards of 30 minutes to complete. However, on this four-processor server with 8GB of RAM, it actually took around 80 minutes to return control.

Although you may feel disinterested in filling out the company information component of the initial installation, this information is not actually sent to Microsoft. Instead, this information is used to pre-populate a number of fields on the server such as fax cover pages and the pre-built Web site.

SBS…with Optional Plus One

SBS 2008 is designed to be a holistic solution for the data and messaging needs for a small business of 75 users or less. SBS 2008 arrives in two versions, with the Standard Edition including a license for Windows Server 2008, Exchange Server 2007, pre-installed instances of WSUS and Microsoft Office Live Small business, and 120-day trial licenses of Forefront Security for Exchange Server and Microsoft Live OneCare. Businesses that find themselves needing a second server for their processing needs can optionally purchase the Premium version, which adds an extra Windows Server 2008 license to be used on a second server. Also included with Premium Edition is a license for SQL Server 2008 Standard Edition with downgrade rights to use SQL Server 2005 if necessary. Five CALs are provided natively with the purchase, with license packs being available in as few as 1-packs, as organizations find the need to add users or devices.

Interestingly enough, SBS 2008 can only be installed onto 64-bit hardware, with x64 being a full requirement along with a minimum of 4GB of RAM. These requirements, though appearing stiff at first blush, are necessary for SBS to run with good performance considering the multitude of services being provided by this single server.

Also, Exchange Server 2007 has a requirement for x64, which is a primary driver for the entire system requirement.

Managing Your New SBS Server

As with previous versions, once the installation completes, Microsoft has wrapped much of the administrative responsibilities of managing your new SBS server into a centralized console. This Windows SBS console is available through the Start menu and starts with a list of common Getting Started tasks that assist the administrator with correctly completing the initial configuration. These tasks include setting up Internet addresses, configuring Smart Hosts for starting the receipt of inbound mail, adding trusted certificates if necessary, and configuring server backup.

SBS is designed for the part-time IT professional, so this SBS console makes easy the typical processes necessary for getting up and running. From the console, seven major elements are presented:

1. *Home.* Here resides the list of initial tasks as well as high-level alerting for security, updates, backups, and other alerts. Drilling down into each alert category takes you to the specific console page where the problem can be resolved.

2. *Users and Groups.* Somewhat different than the Active Directory Users and Computers console used elsewhere, this console page enables a set of tasks associated with the creation and management of users and groups. Adding to the traditional users and groups, however, is the ability to add User Roles. This capability associates a "role" with a set of group memberships, email settings like quota and access to Outlook Web Access, remote access capabilities, as well as shared folders quota and redirection. With this capability being integrated with the new user creation process, the ability to do roles-based administration will be a boon for SBS administrators.

3. *Network.* The network tab provides a heads-up display for all the computers within the domain, showing stoplight charts with information on their backup, alert, update, and security status information. From this node, new computers can be added to the network, and as a very exciting add-on, power management settings for connected Windows Vista machines can be managed.

4. *Shared Folders and Web Sites.* Here, new shared folders can be created through the same new Provision a Shared Folder Wizard now available in Windows Server 2008. This new wizard, in both the SBS and "regular" versions of Server 2008, is a great new addition that integrates all the minute steps for creating a shared folder that were otherwise found in multiple places across the system. For Web sites, three are created by default, one for Outlook Web Access, another to be used as an internal Web site, and a third Remote Web Workplace (RWW) site used for connecting users from outside the company LAN into internal resources. This RWW site provides users a single location from which to check email, connect to a computer, or view the internal Web site, and if configured properly with the right security, can be of great utility for small businesses that might not otherwise be able to afford other remote access solutions.

5. *Backup and Server Storage.* Server backups for computers all across the domain are consolidated into this page of the console. Here, server backups can be configured and scheduled through the interface, while the status of backups can be monitored. With backups—and the lack of good ones—being a critical risk for any business, this tool gives small businesses the "warm fuzzy" that they're successfully getting backups stored. Note that although previous backup utilities could use tape storage, backups with Windows Server 2008 and SBS 2008 require disk-based backup. Although this changes the paradigm for backups quite a bit, Microsoft realizes that USB-based disk drives are inexpensive solutions (especially in comparison with tape-based alternatives) that can store huge amounts of data. SBS backups can work with multiple USB-based disks to store backups and manually rotate backups off-site. I myself was a little hesitant at first to hear about the procedures recommended by Microsoft for completing these new types of backups, but as I spent some time thinking about carrying around and supporting external drives rather than cranky and cantankerous tapes, I grew more impressed with this new solution.

6. *Reports.* Within this node, any number of reports based on any of the functionality discussed here can be created and subsequently emailed to individuals within the organization. One of the biggest problems with many small environments is in getting the necessary alerts to know when problems have occurred before they become critical. Integrating these reporting and emailing capabilities into SBS ensures that the IT administrators in these environments get the information they need.

7. *Security.* Lastly, is a node that aggregates many of the security-based configurations required in today's server-based systems. The security node monitors for virus protection, malware protection, client firewalls, and spam and virus protection for email as well as the status associated with each. As with each of the other nodes, keeping this information in a single location helps the small business administrator gain a situational awareness of the network's security posture from a single location.

What is particularly useful about the configuration of SBS is that none of the traditional management consoles have been restricted or otherwise made unavailable. Thus, for more skilled administrators used to the traditional management tools, it remains possible to administer the system and the domain using familiar tools.

So, for a relative newbie's first glimpse into SBS 2008, I personally am impressed with what I see. Microsoft's realization that some small businesses need a second server to offload database or other processing is a smart move, as is keeping the SQL components encapsulated into the SBS purchase line-item and management toolsets. Although businesses that are approaching the upper limit of SBS's 75 users will likely consider upgrading to Microsoft's higher-level solution, Windows Essential Business Server, those that remain true small businesses will enjoy the simplicity that comes with the SBS solution.

Greg Shields, MCSE: Security, CCEA, is an independent author, speaker, and consultant, based in Denver, Colorado. With more than 10 years of experience in information technology, Greg has developed extensive experience in systems administration, engineering, and architecture. Greg is a contributing editor for both Redmond magazine and MCPmag.com, authoring two regular columns along with numerous feature articles, webcasts, and white papers. He is also the resident editor for Realtime Publishers' Windows Server Community at www.realtime-windowsserver.com. Greg is currently finishing his new book Windows 2008: What's New, What's Changed through SAPIEN Press.

Practical PowerShell

"I've Got Your Back[Up]"

by Jeffery Hicks

When it comes to managing Windows event logs in PowerShell, Windows Management Instrumentation (WMI) should be an obvious choice—especially when it comes to managing event logs on remote computers. This month, I have a rather lengthy and complex function for backing up event logs and optionally clearing them. You should be able to download the function from the Realtime site. Don't try to copy and paste from here, as code sometimes gets misformatted in the production process. Remember, to use this function, you either have to copy it to your profile or dot source the script file.

You can download the following code from http://www.realtime-windowsserver.com/code/v1n7_Practical_PowerShell.zip

```
Function Backup-EventLog {
    Param([string]$computername=$env:computername,
    [string]$path="C:\Backup",
    [switch]$prompt,
    [switch]$clear,
    [switch]$verbose,
    [switch]$force,
    [switch]$help
    )

    Function Show-Help {
        Param([string]$script,[string]$parameters,[string]$synopsis,
        [string]$full,[string]$examples)

        $pad="  "

        $syntax="{0} {1}" -f $script,$parameters
        write "NAME"
        write ($pad + $script)
        write "SYNOPSIS"
        write ($pad + $synopsis)
        write "SYNTAX"
        write ($pad + $syntax)
        write "PARAMETERS"
        write $full
        write " "
        write $examples

    }
```

```
    if ($help) {
       #show help for the current script
       $script=$myinvocation.invocationname
       $synopsis="Backup and clear Windows Event logs to a specified location."
       $parameters="-path <string> [-computername <string>] [-prompt <switch>] [-clear
<switch>] [-verbose <switch>] [-force <switch>] [-help <switch>]"
       $full="  -path <string> `
    The backup folder. The location will be verified.`
  -computername <string> `
    The name of the computer to backup. The default is the local computer.`
    You cannot use alternate credentials with this script.
  -prompt <switch> `
    If specified, prompt the user to backup and clear each event log file.
  -clear <switch>
    Specify whether to clear the event log if it is successfully backed up.`
    The default is to NOT clear the log.
  -verbose <switch>
    Turn on verbose messages
  -force <switch>
    Windows will not overwrite an existing backup with the same name. `
    Specify -force to overwrite existing files.
  -help <switch>
    Display this help message"

       $examples="
  ------------------Example 1------------------ `
  Backup-EventLog -computername 'SERVER01' -prompt -clear `


  ------------------Example 2------------------ `
  Backup-EventLog -path '\\server01\backups' -force -prompt | where {$_.backup} | select
Computername,Logfile,Size,Events,BackupLog,Cleared
       "

       Show-Help -script $script -parameters $parameters -synopsis $synopsis -full $full
-examples $examples
       Return

    }
```

```
    if (!(Test-Path $path)) {
        Write-Warning “Failed to verify $path”
        Return
    }
    else {
        if ($verbose) { Write-Host “Verifed $path” -ForegroundColor CYAN}

    }

    #verify computer can be contacted
    $logs=Get-WmiObject win32_nteventlogfile -ComputerName $computername -ea
“SilentlyContinue”

    if ($logs.count -lt 1) {
        Write-Warning “There are either no logs to backup or failed to connect to
$computername!”
        Return
    }

    else {
        if ($verbose) {
            $msg=”Found {0} logs to backup on {1}” -f $logs.count,$computername
            Write-Host $msg -ForegroundColor CYAN
        }

    }

    foreach ($log in $logs) {
        #build a backup file name from the computer name, date and logfile
        #replace spaces in the log file name with _

        $filename=”{0}_{1}_{2}.evt” -f (Get-Date -format yyyyMMdd),$log.CSName,($log.
logfilename).Replace(“ “,”_”)
        $backup=Join-Path $path $filename
        #enable backup and security privileges
        $log.psbase.scope.options.enablePrivileges=$TRUE

        #this variable will determine if the log should be backed up
        $go=$TRUE

        #create custom object
```

```
 $obj=New-Object PSObject
        $obj | Add-Member -MemberType NoteProperty -Name “Computername” -Value $log.
CSName
        $obj | Add-Member -MemberType NoteProperty -Name “Logfile” -Value $log.
logfilename
        $obj | Add-Member -MemberType NoteProperty -Name “Size” -Value $log.filesize
        $obj | Add-Member -MemberType NoteProperty -Name “Events” -Value $log.
NumberofRecords

        if ($prompt) {
            $msg=”Do you want to backup the {0} event log [{1} KB {2} events] [YN]” -f
$log.logfilename,[int](($log.filesize)/1KB),$log.NumberofRecords
            $reply=Read-Host $msg
            if ($reply -eq “y”) {
                $go=$TRUE
            }
        else {
            $go=$False
            }
        }

        if ($go) {

            #backup can’t overwrite an existing file so delete it if
            #$force is specified

            if ((Get-ChildItem $backup -ErrorAction “SilentlyContinue”) -AND ($force)) {
                Remove-Item $backup
            }
   $obj | Add-Member -MemberType NoteProperty -Name “BackupLog” -Value $backup

            if ($verbose) {
                $msg=”Backing up {0} to {1}” -f $log.LogFileName,$backup
                Write-Host $msg -ForegroundColor CYAN
            }
           $rc=$log.backupeventlog($backup)
            if ($rc.ReturnValue -eq 0) {
                if ($verbose) {Write-Host -foreground GREEN “Backup successful” }
                $obj | Add-Member -MemberType NoteProperty -Name “BackUp” -Value $TRUE
                $obj | Add-Member -MemberType NoteProperty -Name “BackupReturn” -Value 0
```

```
                #clear log file since backup was successful
                if ($clear) {
                    if ($verbose) {Write-Host "Clearing eventlog" -foregroundcolor CYAN}
                    $rc2=$log.ClearEventLog()
                    if ($rc2.ReturnValue -eq 0) {
                        $obj | Add-Member -MemberType NoteProperty -Name "Cleared" -Value
$TRUE

                        if ($verbose) {
                            $msg="Successfully cleared {0} on {1}" -F $log.
logfilename,$log.CSName
                            Write-Host $msg -ForegroundColor GREEN
                            }
                        }
                        else {
                            $obj | Add-Member -MemberType NoteProperty -Name "Cleared"
-Value $False
                            if ($verbose) {
                                $msg="Failed to clear {0} on {1}. Return code {2}" -F
$log.logfilename,$log.CSName,$rc2.returnvalue
                                Write-Host $msg -ForegroundColor RED
                            }
                         }
                } # end if $clear
                else {
                    $obj | Add-Member -MemberType NoteProperty -Name "Cleared" -Value
$False
                }
              }
            else {
                if ($verbose) {
                    $msg="Backup failed with a return value of {0}" -f $rc.ReturnValue
                    Write-Host -foreground RED $msg
                }
                $obj | Add-Member -MemberType NoteProperty -Name "BackUp" -Value $False
                $obj | Add-Member -MemberType NoteProperty -Name "BackupReturn" -Value
$null
                $obj | Add-Member -MemberType NoteProperty -Name "Cleared" -Value $False

           }
```

```
        } #end if $go loop

          Else {
              $obj | Add-Member -MemberType NoteProperty -Name "BackUp" -Value $False
              $obj | Add-Member -MemberType NoteProperty -Name "BackupReturn" -Value $null
              $obj | Add-Member -MemberType NoteProperty -Name "Cleared" -Value $False
          }

        write $obj

       } #end ForEach loop
}
```

Here's how this all works. The function takes a number of parameters, many of them switches. I've hard coded a default location for the backups, which, of course, you can change or specify a location when you execute the function. You'll also need to specify a computer name:

```
PS C:\Backup-eventlog -path \\server01\backup\logs -computername SERVER01
```

The function gets all Windows event logs with this WMI expression:

```
    $logs=Get-WmiObject win32_nteventlogfile -ComputerName $computername -ea
"SilentlyContinue"
```

If $logs doesn't exist, there was some sort of error and the function will exit:

```
    if ($logs.count -lt 1) {
        Write-Warning "There are either no logs to backup or failed to connect to
$computername!"
        Return
    }
```

Assuming event logs were found, each Windows event log will be backed up to the specified location. I've defined a default file name in the function:

```
$filename="{0}_{1}_{2}.evt" -f (Get-Date -format yyyyMMdd),$log.CSName,($log.
logfilename).Replace(" ","_")
$backup=Join-Path $path $filename
```

The filename will be in the format YYYYMMDD_computername_logname.evt. For example, from Server01, you should see a backup log like 20080812_SERVER01_Application.evt. Any log file names with spaces in the name will have the space replaced with an underscore. This name is joined with Join-Path to create a valid and complete filename.

Because you might need to back up the security event log, you also need appropriate WMI privileges:

```
        $log.psbase.scope.options.enablePrivileges=$TRUE
```

I like my functions to write objects to the pipeline, so I create a custom object for each log file that will provide information about each log and the backup process.

```
$obj=New-Object PSObject
        $obj | Add-Member -MemberType NoteProperty -Name "Computername" -Value $log.
CSName
        $obj | Add-Member -MemberType NoteProperty -Name "Logfile" -Value $log.
logfilename
        $obj | Add-Member -MemberType NoteProperty -Name "Size" -Value $log.filesize
        $obj | Add-Member -MemberType NoteProperty -Name "Events" -Value $log.
NumberofRecords
```

I'll add properties throughout the rest of the function, but now we're ready to back up each log:

```
            $rc=$log.backupeventlog($backup)
```

I've added code to check the return value of the BackupEventLog() method, which will indicate whether there was an error:

```
            if ($rc.ReturnValue -eq 0) {
                if ($verbose) {Write-Host -foreground GREEN "Backup successful" }
                $obj | Add-Member -MemberType NoteProperty -Name "BackUp" -Value $TRUE
                $obj | Add-Member -MemberType NoteProperty -Name "BackupReturn" -Value 0
```

Notice that I've added a few more properties to the custom object indicating that the log was backed up and the backup method's return value.

By default, the function does not clear any event logs, but if you specify –Clear, it will (assuming the backup was successful):

```
                if ($clear) {
                    if ($verbose) {Write-Host "Clearing eventlog" -foregroundcolor CYAN}
                    $rc2=$log.ClearEventLog()
                    if ($rc2.ReturnValue -eq 0) {
                        $obj | Add-Member -MemberType NoteProperty -Name "Cleared" -Value
$TRUE
```

If the ClearEventLog method was successful, another property is added to the custom object. This process repeats for every event log writing an object like this for each log to the pipeline:

```
Computername : PUCK
Logfile      : System
Size         : 69632
Events       : 17
BackupLog    : c:\backup\20080618_PUCK_System.evt
Cleared      : False
```

This means you could use a PowerShell expression like this:

```
PS C:\ get-content servers.txt | foreach {Backup-EventLog -computername $_ -path "\\
server2\backup" } | where {$_.backup} | select Computername,Logfile,Size,Events,BackupLog
,Cleared
```

This should go through every computer in servers.txt, back up event logs on each one, and then display information for every log that was successfully backed up.

The function has a few other bells and whistles I'd like to point out. The –verbose switch will display additional messages written directly to the PowerShell host. You can identify these messages because they are in color.

There might also be situations in which you don't want to back up and clear every event log. In these cases, use –prompt, which will use the Read-Host cmdlet to ask whether you want to back up the file:

```
        if ($prompt) {
            $msg="Do you want to backup the {0} event log [{1} KB {2} events] [YN]" -f
$log.logfilename,[int](($log.filesize)/1KB),$log.NumberofRecords
            $reply=Read-Host $msg
            if ($reply -eq "y") {
                $go=$TRUE
            }
```

You have to answer "Y" or "y" in order to back up the log.

One thing I want to point out is my use of the –f operator to create the $msg variable. The –f operator replaces placeholders like {0} with corresponding values from a comma separated list. Thus {0} will be replaced with the value of $log.logfilename, {1} will be replaced with the log file size, and {2} will be replaced with the number of event log records:

$msg="Do you want to backup the {0} event log [{1} KB {2} events] [YN]" -f $log.logfilename,[int](($log.filesize)/1KB),$log.NumberofRecords

The BackupEventLog() method will not overwrite an existing file with the same name, so if you were to run this function repeatedly on the same day, no other backups would occur. Use the –force parameter to instruct the function to delete the file if it already exists:

```
            if ((Get-ChildItem $backup -ErrorAction "SilentlyContinue") -AND ($force)) {
                Remove-Item $backup
            }
```

Finally, given how complex this function is, I thought it would be nice to include some help. If you use the –help switch, you'll get a help summary like this:

```
PS C:\backup-eventlog -help
NAME
  Backup-EventLog
SYNOPSIS
  Backup and clear Windows Event logs to a specified location.
SYNTAX
  Backup-EventLog -path <string> [-computername <string>] [-prompt <switch>] [-
clear <switch>] [-verbose <switch>] [-force <switch>] [-help <switch>]
PARAMETERS
  -path <string>
    The backup folder. The location will be verified.
  -computername <string>
    The name of the computer to backup. The default is the local computer.
    You cannot use alternate credentials with this script.
  -prompt <switch>
```

```
 If specified, prompt the user to backup and clear each event log file.
  -clear <switch>
    Specify whether to clear the event log if it is successfully backed up.
    The default is to NOT clear the log.
  -verbose <switch>
    Turn on verbose messages
  -force <switch>
    Windows will not overwrite an existing backup with the same name.
    Specify -force to overwrite existing files.
  -help <switch>
    Display this help message



  ------------------Example 1------------------
  Backup-EventLog -computername 'SERVER01' -prompt -clear




  ------------------Example 2------------------
  Backup-EventLog -path '\\server01\backups' -force -prompt | where {.backup} |
 select Computername,Logfile,Size,Events,BackupLog,Cleared
```

As with the other switch parameters, if the parameter is passed, it will have a value of TRUE:

```
    if ($help) {
        $script=$myinvocation.invocationname
        $synopsis="Backup and clear Windows Event logs to a specified location."
        $parameters="-path <string> [-computername <string>] [-prompt <switch>] [-clear
<switch>] [-verbose <switch>] [-force <switch>] [-help <switch>]"
        $full="  -path <string> `
```

As you look through this function, I'm sure you'll discover many opportunities for enhancements or modifications to meet your needs. For example, you might want a simpler function to only back up a single log file. Or you might want to add support for alternate credentials. I'll leave these for you to work out on your own. Feel free to post any questions in the PowerShell forum at ScriptingAnswers.com.

Jeffery Hicks, MCSE, MCSA, MCT, and Microsoft PowerShell MVP, is a Scripting Guru for SAPIEN Technologies. Jeff is a 16-year IT veteran. He has co-authored and authored several books, courseware, and training videos on administrative scripting and automation. His latest book is WSH and VBScript Core: TFM (SAPIEN Press 2007). You can contact him at jhicks@sapien.com.

Microsoft Exchange Hosted Services

by J. Peter Bruzzese

Microsoft often leaves certain aspects of the server world to others to develop and sell. But every once in a while, the company decides to enter a new arena. At times they develop a new product. Other times they simply purchase or partner with a company that has the product they need. Well, with that in mind, when I first heard the idea of *Microsoft Exchange Hosted Services*, I immediately thought this was going to be an attempt on Microsoft's part to go up against hosted solutions such as Rackspace. In actuality, Microsoft doesn't want to invade that space (yet). Instead, they have decided to focus on four distinct services:

- Provide protection from the ever-increasing array of threats in the form of viruses and malware
- Assist with email retention so that companies can abide by legal requirements
- Encrypt messaging data
- Assist with access to messaging in the most extreme disaster scenarios

Check out the Microsoft Exchange Hosted Services home page.

Although the title Exchange Hosted Services may lead one toward the idea that your messaging will be hosted by Microsoft, that isn't the case at all. However, these other services may be of real value to companies, so let's investigate what Microsoft is offering to Exchange organizations and how the company is providing these solutions to customers.

The Four Hosted Services Overview

With the idea that you already have an in-house or hosted Exchange environment, the concept is for you to take advantage of one of the following solutions:

- **Microsoft Exchange Hosted Filtering:** Utilizing multiple filter types, inbound and outbound traffic is filtered before harm can be done to your network or to others. The service provides multiple antivirus engines that offer 'massive virus signature libraries' and powerful heuristic detection. Spam filtering is also included with several methods to either remove or quarantine spam coming into your organization, as well as policy settings that allow you to comply with government security restrictions. Disaster recovery services are offered so that in the event the destination email server is unavailable, the hosted filtering server will hold that mail for as long as 5 days and try to deliver it every 20 minutes. There are also real-time message trace and reporting tools included.
- **Microsoft Exchange Hosted Archive:** With a greater emphasis than ever before on messaging being archived for legal purposes, it can be a full-time job worrying about archiving. Thus, there is an ever-increasing need for admins to offload that worry to someone else. This service promises to handle those concerns. Providing the filtering features mentioned earlier, clean messages are sent to your company server with a copy being sent to a storage repository either in or outside of the US (it's your choice). To keep the size of the repository in check, a de-duplication process and single instance storage is used. One of the nice features is that the system will automatically be aware of the life cycle of your messages and delete messages that fall outside of the legal scope, unless you specify otherwise.

- **Microsoft Exchange Hosted Continuity:** Learning from recent disasters that have struck all around the world, the increasing concern that a messaging solution could be down for an extended period of time is what drives this service. With this service in place, users will be able, either due to a planned outage or immediately following a disaster, to access the previous 30 days of mail and be able to send and receive email. Work can continue unhindered. The data storage location can be located in or outside of the US (again, it's up to the customer). Once the outage is fixed, all the messages that have been received will be automatically added to the primary email service, while sent messages can be optionally restored as well. (One other interesting feature here is that the service, being that it is always on, can restore messages that may be lost or deleted from the primary messaging service, provided it falls within the scope of retention).
- **Microsoft Exchange Hosted Encryption:** This is a solution designed by Voltage Security that employs Identity-Based Encryption (IBE), where the user's email address becomes the public key (eliminating the need for certificates, which are used in traditional PKI encryption solutions). Essentially, the sender doesn't have to do anything at all. When an email is sent, it uses a Transport Layer Security (TLS)-encrypted tunnel and is automatically encrypted. The recipient receives a private key based upon their identity through their email address as well, and the message can be decrypted by the recipient without either the sender or recipient performing any special actions. This is actually a great solution for those who are working to be in compliance with government regulations such as the Health Insurance Portability and Accountability Act (HIPAA). All messages handled by Exchange Hosted Services will use TLS encryption and will attempt to use TLS with other servers that mail is sent to, unless the destination server is not prepared to handle it, in which case SMTP will be used.

How These Solutions Are Provided

These four services are broken down separately, but they can be deployed collectively as "Software as a Service" where all you, the admin, need to do is change your simple mail exchange (MX) record on your external DNS servers. No additional hardware nor any other software configuration changes are required on the part of the company. And with the MX record pointing to the Microsoft network (with all of those services in place), your company becomes invisible to the outside world of 'malicious mailers.'

Your next obvious concern is cost. Everything mentioned must cost something. Well, Microsoft offers these options through licensed service providers and reseller partners. Microsoft offers a simple break down of estimated pricing with a per user, per month model that allows you to pick and choose the services you might need from a small business perspective. For Volume Licensing, you have to go through the Microsoft Volume Licensing Services through one of the many programs Microsoft has in place.

Summary

Obviously, there are other solutions on the market to provide each and every one of the services just discussed. Microsoft isn't billing itself out as the only solution—or even the ultimate solution—but as a reliable, quality solution backed by guaranteed service from which your company may be able to benefit. If you have alternative solutions you prefer, drop me a line and let me know what they are and why you love them and I'll put them in a future article with your name and comments. If you make the article, I'll send you a copy of my book Tricks of the Vista Masters!

J. Peter Bruzzese is an MCSE (NT,2K,2K3)/MCT and MCITP: Enterprise Messaging Administrator. His expertise is in messaging through Exchange and Outlook. J.P.B. is the Series Instructor for Exchange 2007 for CBT Nuggets. In harmony with the joy of writing Exclusively Exchange for Realtime Publishers, he has created a free Exchange training site at www.exclusivelyexchange.com. He is co-founder of ClipTraining.com, a provider of short, educational screencasts on Exchange, Windows Server, Vista, Office 2007, and more. You can reach Peter at jpb@cliptraining.com.