Realtime publishers
"Leading the Conversation"

March 2008
Volume 1, Number 2

Windows Administration
in Realtime

Letter from the Editor

Getting the Kinks Out

by Greg Shields

Not long ago I built a brick patio in my backyard. On a lazy spring morning I got the hair brained idea that I could do it all by myself, without even a lick of instructions. I grabbed a pile of bricks, a shovel, and took to digging the right-sized hole and lying them all into place.

It wasn't until after I laid the last brick and called the wife out to the back yard did I discover the need for ***sand*** and ***gravel***. You see, when you build a brick patio, sand and gravel are critically necessary to make sure the bricks lie straight and even. My first attempt ignored that need completely, so my brand new patio looked a lot like a freeze-frame of the ocean in a bad storm.

Building and distributing a newsletter like this one can be a lot like building a brick patio, though there aren't really any instructions. Sometimes you hit it on the head on your first attempt. Sometimes it takes a time or two to get everything down just right. Unlike my brick patio, we here are Realtime are pretty happy with our first eJournal, and we believe that in this our second we've really hit it out of the park.

In this issue we'll be helping you out with all the new and exciting parts of Server 2008's Group Policy. You'll ponder a discussion on whether you should upgrade to Vista or not. You'll get valuable knowledge on managed folders in Exchange. And we'll even leave you with a neat trick on how use PowerShell to determine who is logged into a particular server.

Please do enjoy Issue #2, and let us know how we're doing at feedback@realtimepublishers.com.

After all, myself included, we think we're much better writers than bricklayers.

The Industry Outlook

Hot Topics in the IT Arena

by Don Jones

Automation Trends Continue

Microsoft release of Windows PowerShell continues to gain momentum as more third-party software vendors enable automated management of their products using the new shell. Recent additions to the lineup include VMWare's ESX Server, Citrix' MetaFrame, and now IBM's WebSphere. Download Windows PowerShell from www.microsoft.com/powershell, and connect to community resources at www.powershellcommunity.org.

Bill Gates Doesn't Get the "Wow"

In a recent series of video interviews with blog Web site Gizmodo.com, Bill Gates was asked what Microsoft products from the past five years he personally wishes the company had put more polish on. "Ask me after we ship the next version of Windows," he replied, "and I'll give you a more honest answer." Other sources suggest discontent within Microsoft's leadership over Vista's reception by corporate customers: While the company points to surging sales, most of those are attributable to OEM licenses and aren't necessarily a reflection of actual in-use copies of the new OS.

Remember Your Password, Check Your Psyche

Network World reports that Microsoft Research is testing Rorschach-style inkblots as a means of remembering complex passhrases. Any given person will usually "see" something different in a given inkblot: A woman, a butterfly, a pack of dogs, and so forth. The research project suggests that inkblots could be an easy way to remind users of complex passwords such as "My Dog Fred," without necessarily giving the password away to others - who would interpret the inkblot differently. www.InkblotPassword.com is the public Web site for the project, which currently features more than 1,000 inkblots.

Windows Server 2008: Closer than you Think

As of this writing, Windows Server 2008 is in Release Candidate status, the last step before final Release To Manufacturing (RTM). Businesses are showing marked pre-release interest in the new server OS, particularly the new Server Core installation option, which provides a smaller footprint and fewer lines of code for critical infrastructure services such as DNS, Active Directory, DHCP, file serving, and more. Experts suggest that the majority of up-front Win2008 migrations will be moving infrastructure services to Server Core. Have you started evaluating the impact of Win2008 on your environment? Visit www.realtime-windowsserver.com to join the conversation.

Say No To Homogenous Clients

An increasing number of businesses are deliberately introducing more variety into their desktop client OS mix, including Linux variants and Apple Mac computers. The US Army is under a directive to increase "client diversity" in part to help reduce the damage any single platform-specific attack, such as a virus, can do. As Macs continue to gain consumer market share, and continue to pick up third-party software that makes them more enterprise-friendly (including product that allow GPO-style administration of Mac clients), businesses are conducting feasibility pilots and other tests. Microsoft continues to hedge its bets on the OS front by offering Mac versions of key business products, including Microsoft Office.

What's New with Group Policy in Windows Server 2008 and Windows Vista

by Derek Melber

Group Policy has gone through some major changes over the years. Since it was released with Windows 2000 Server, it has gone from around 600 settings to now with Windows Server 2008, more than 5000 settings. The administration of Group Policy has gone from archaic to efficient and straightforward. The changes that have occurred to some of the technologies are also revolutionary: slow link detection, delegation, archiving of Group Policy Objects (GPOs), and management of some of the key files that Group Policy employs. The changes delivered via Windows Vista and Windows Server 2008 have brought Group Policy a long way. These improvements will certainly make Group Policy a technology more effectively utilized all companies that have Active Directory (AD).

Group Policy Technology Changes

The overall foundation that Group Policy is built upon is still the same. This includes the concepts that Group Policy uses Client-Side Extensions to take the settings delivered from the domain controllers to apply the settings to target computers. Other core concepts that are still the same include:

- GPOs apply to user and computer accounts
- GPOs still don't apply to groups
- GPOs can be filtered based on group membership
- GPOs can be configured to be enforced
- AD nodes can be configured to block inheritance of policy application down through the AD structure
- WMI filters can be linked to GPOs to control which objects receive settings
- GPOs can be linked to the domain, organizational units(OUs), and sites

However, there are some changes that make Group Policy more powerful, stable, and loaded with new features.

New Group Policy Service

Group Policy has always relied on the WinLogon service to perform all of its actions. This was not a horrible decision or design, it just made one more service leverage WinLogon. Since the WinLogon service already has enough relying on it, Microsoft made a change with Windows Vista and Windows Server 2008 in that there is no more reliance on the WinLogon service.

Instead, Group Policy now relies on its own service. The new service is called gpsvc and can be seen running when you look at the Task Manager or the list of services in Computer Management. With this separation from the WinLogon service, Group Policy can leverage more flexibility and have more benefits than before.

First, the Group Policy infrastructure stability does not rely on WinLogon. Since there were so many other facets to WinLogon, it could become unstable and, as a result, Group Policy would fail. The new Group Policy service uses a new architecture to perform all notifications and processing of Group Policy. In addition to the stability that the new Group Policy service provides, it offers these benefits:

- Any and all files that are used in conjunction with Group Policy can be delivered without the computer receiving the files needing to be restarted. This is very important for computers that will be managing Group Policy, as they will potentially require file updates for the management of the 5000 plus settings that are now available.
- The application of Group Policy is more efficient due to fewer resource requirements to process background refreshes.
- For computers that are consuming Group Policy settings, there is less memory used for Group Policy processing. This will increase overall performance and eliminate the need for Group Policy to rely on multiple services.

- The Group Policy service is now started automatically and can't be disabled. This creates a stable environment from bootup through Group Policy refresh.

ADMX and ADML files

There has been a historical issue with one particular type of file that is associated with Group Policy. The .adm templates have been a bone of contention for a long time. This file type has been around since Microsoft released the concept of "policies" back with the release of System Policies in the Windows NT era. The files have been around a long time, but now are no longer shipped with Windows Server 2008 or Windows Vista.

Instead of .adm templates, Windows Server 2008 and Windows Vista use ADMX and ADML files. These new files use XML formatting and are more flexible than the older .adm templates. These files are used for a couple of reasons, with the main reasons highlighted in the following list:

- The combination of ADMX and ADML files allow for nearly any language to be supported. The .adm templates were limited to English-only, which was a problem for IT professionals that did not use English as their first language to customize the .adm templates.

- ADML files contain the text that shows up in the Group Policy Management Editor (GPME). The file is XML-based and supports many languages, so it creates a very flexible and advantageous format for use with other languages.

- ADMX files are very specific. There are more than 130 ADMX files, which allows for easy updates and moving around of files.

- ADMX and ADML files are not stored in the SYSVOL on domain controllers. The local versions of these files are used or the central store is used.

> More information about the central store is available later in the article.

Local GPOs

Most companies will not be relying on the use of local GPOs with Windows Vista, but if you work in a small office or run a network of Windows Vista computers at home, the changes to local GPOs is something you should look at. With Windows 2000, Windows XP, and Windows Server 2003, there was only a single local GPO that could control the computer and every user that logged on. This caused quite a problem due to the variety of different users that could log on, primarily affecting the administrators that logged into the computer.

Now, with Windows Vista, there are four local GPOs. The goal of the new structure is to allow for generic settings all the way down to individual settings if that is necessary. The local GPOs still apply before AD-based GPOs; as well, all local GPOs have weaker precedence compared with the GPOs that are linked to AD nodes.

The four different local GPOs include the following:

- Local Policy Object—The Local Policy Object is the same as the local GPO in past OS versions. This policy is designed to be generic, configuring settings that will apply to the computer and any user that logs in.

- Administrators Local GPO—As the name implies, the Administrators Local GPO is geared toward managing all the user accounts that have membership in the local Administrators group on the computer where the GPO is configured. With this new GPO option, users that have administrative privileges to the computer can have more capabilities and features than non-administrators.

- Non-Administrators Local GPO—In the same light as the Administrators Local GPO, there is a GPO for non-administrators. All users that don't have membership in the local Administrators group will be affected by this GPO. This allows the local computer to configure settings for both types of users without having to create special groups or multiple GPOs to accommodate the settings. Figure 1 illustrates what the Administrators and Non-Administrators local GPOs look like when being administered.

- Specific User Local GPO—If there is a special user that is configured in the local Security Accounts Manager on a computer that needs unique settings beyond the three mentioned local GPOs, this user can have a unique GPO. This is a great setting for computers that might have service type accounts, multi-user accounts, or a kiosk user.

Group Policy Management Changes

There have been some dramatic changes to what can be done to manage GPOs. The new GPMC brings much of these changes, including

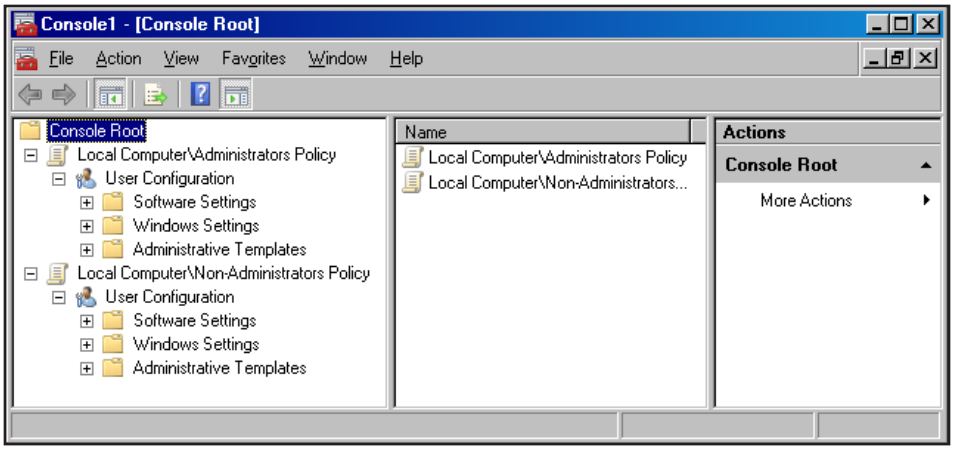

Figure 1: The Administrators and Non-Administrators local GPOs are managed using the MMC.

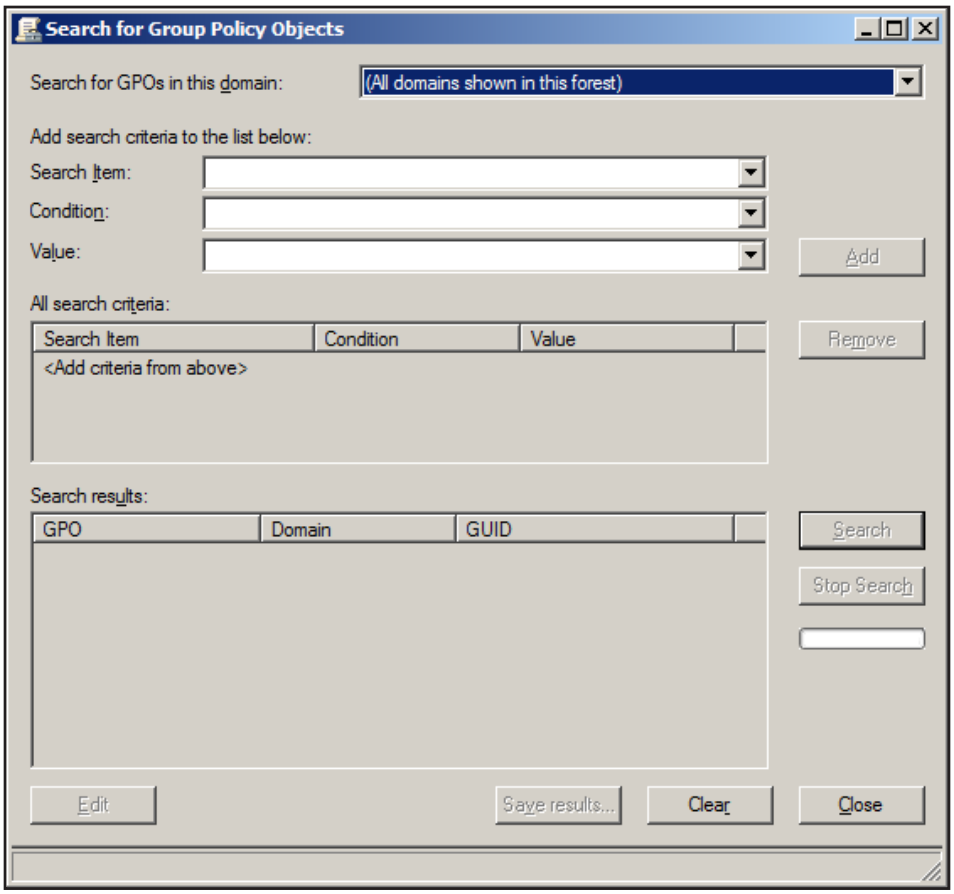

Figure 2: You can perform a search on nearly every aspect of a GPO and the Group Policy infrastructure in the new GPMC.

searching, filtering, and commenting. Combine these new features with the ADMX central store and Advanced Group Policy Management (AGPM) and there is not much missing from what you need to control GPOs in your environment.

GPMC

The GPMC that ships with Windows Server 2008 provides some much-needed and powerful new features. These features include the ability to search, filter, and comment on GPOs and the settings. In order for you to take advantage of these new features on your Windows Vista computers, you will first need to install Windows Vista SP1, then download and the install the Remote Server Administration Tools (RSAT) from Microsoft. At the time of writing, the RSAT was still in Beta.

Search

The search feature that comes with the new GPMC provides a superior method to find settings with a GPO. You are able to perform searches for many aspects of the GPO and infrastructure, as Figure 2 shows, including the following areas:

- GPO Name
- GPO-links
- Security Group
- Linked WMI Filter
- User Configuration
- Computer Configuration
- GUID

Filtering

Filtering can be done on the settings that exist under the Administrative Templates nodes, which are located under both the Computer Configuration and User Configuration sections in a GPO. The filtering allows you to view settings based on different criteria, including whether the GPO setting is managed, configured, or has comments associated with it. In addition, you can search on keywords.

Commenting

You can document the reasoning for all the GPO settings that are configured as well as any changes to

the GPO through the commenting feature. Comments can be provided at the GPO level as well as at the individual setting level for the settings that fall under the Administrative Templates nodes.

Central Store

The administration of .adm templates has always been a problem. Now with the inclusion of the ADMX and ADML files, you have a central repository that you can use to manage these files. The central store for storage and management of these ADMX and ADML files is easy to create and configure. To create the central store, you need only to create a new folder named PolicyDefinitions under the *%systemroot%\sysvol\<domainname>* policies folder on your domain controllers. Then copy all the ADMX and ADML files, within their proper folder structure, to this folder. Once the folder and files exist on the domain controllers, the central version (rather than the local version) of the files will be used.

AGPM

AGPM is a tool that can be used to manage GPOs like you have wanted to manage them for years. The tool is not free or included with Windows Server 2008 or Windows Vista, but it is designed to work with these OSs. AGPM is part of the Microsoft Desktop Optimization Pack (MDOP) and can be obtained by those customers that have software assurance for Windows Vista desktops.

The tool is loaded with functionality and benefits:

- Offline editing of GPOs
- Change management
- Workflow for creation and deployment of GPOs
- Roll back and roll forward of archived GPOs
- Reporting
- Use of templates for new GPO creation
- Recycle Bin for deleted GPOs
- Restoration of GPOs and GPO links

Group Policy Settings Changes

If you have been bored with the limited number of settings that came with Windows XP SP2 (more than 1600 settings in all), you are in store for quite a shock. Windows Server 2008 now comes with more than 5000 settings in a single GPO. The majority of these new settings fall under a new category referred to as Preferences. These Preferences add a ton of new functionality and control that was always desired but never available until now. In addition, there is a new feature that controls slow link detection!

Group Policy Preferences

These settings were introduced into the Microsoft OS from the acquisition of PolicyMaker from DesktopStandard. These new settings provide more than 20 new Client-Side Extensions and give control over key areas of the server and desktop:

- Local user accounts, including the password
- Local group membership, including the local Administrators group
- Registry values, including multivalue and binary entries
- Power options
- Files and folders, including creation, deletion, and modification of them
- Applications, including control over Outlook, Word, and PowerPoint

Network Location Awareness

No longer does the network speed depend on ICMP; it is now linked to network location awareness. With this new reliance on network location awareness, Group Policy is more respondent, reliable, and more secure. Both the link speed and whether or not the computer is connected to a domain is determined by network location awareness. This feature I employed to help make background refreshes occur more efficiently when a computer is connected to the network via a VPN and when it comes back onto the network in the case of a laptop that is taken off the network frequently.

Summary

Windows Server 2008 and Windows Vista takes Group Policy to a new level of support, features, and capabilities. Microsoft has spent much time and effort in making Group Policy one of the most used and relied upon technologies in AD. With the new management features and settings that come with Group Policy in Windows Server 2008 and Windows Vista, this technology is sure to grab the attention of those that have not used it in the past and will gain momentum for those companies that have been using it along the way.

Derek Melber is a Founding Partner of BrainCore.Net, LLC, an independent

technology consulting and education firm specializing in Microsoft-centric solutions. Derek is the nationally known trainer and author, focusing on Windows 2000, Windows Server 2003, Windows XP, Active Directory, and Group Policy. Some of Derek's publications and videos/CDs can be seen from companies such as Coriolis, Sybex, and KeyStone Learning Systems. Derek is also a well known consultant for designing and implementing Windows 2000 Active Directory and Group Policy solutions for large companies. Derek can be reached online at www.BrainCore.Net.

The Deep Dive

12 Reasons You'll Quickly Upgrade to Server 2008

by Greg Shields

Pundits everywhere have been extolling the virtues of Windows Server 2008. It's more stable! It has better security! It's not Windows Vista! To shore up their claims that Server 2008 is going to be the "Next Great Thing," they troll out all kinds of features and capabilities that arrive with Server 2008 that'll make you squeal with delight.

In a lot of ways, these techno-pundits are right. Server 2008 is a great upgrade to Microsoft's server operating system (OS). But its timeline, released a year after Windows Vista's lackluster reception in the marketplace, means it has a difficult headwall to scale in the minds of IT people everywhere if its adoption is to take off.

All that being said, Windows Server 2008 isn't Microsoft Vista. Though it shares the same code base, the annoying and cumbersome parts of Vista that are railroading its adoption in many environments simply aren't there with the server OS. Though the much-maligned User Account Control (UAC) makes a return in this version, people now understand how best to use it—or at least how best to disable it. Vista's Aero interface isn't readily available in Server 2008, which means that its hardware requirements aren't nearly as high. Most important, in the year's time span between the release of Vista and Server 2008, Microsoft has learned a lot about what people want out of this version of the OS. They want easy manageability. They want central configuration and control. They don't want Microsoft-generated bloatware.

This issue of eJournal celebrates in many ways the release of Server 2008 with this issue arriving shortly after Microsoft's *Heroes Happen Here* launch event for Server 2008, SQL 2008, and Visual Studio 2008 on February 27th in Los Angeles. If you haven't yet signed up for the launch event happening in your area, check out http://www.microsoft.com/heroeshappenhere/register/default.mspx to sign up. Events are happening all across the country and indeed all across the world, so there's little excuse not to get in on the excitement.

Reasons to Party

That being said, there really are a lot of good reasons why you should consider rapidly moving to Server 2008 in your IT environment. Along with these are reasons why the upgrade to Server 2008 is likely going to be much easier on you than was XP to Vista. Since most of us are waiting on our own local launch party, let's put a positive spin on the top-12 reasons why you'll quickly upgrade to Windows Server 2008.

Reason 1: Security

One of the major priorities linked to Server 2008 is the idea of improved security. Server 2008 includes a much improved firewall, one that you'll actually want to turn on and use everywhere. Its Group Policy linkages to the Windows Firewall with Advanced Security are designed in a way that you'll want to configure them for both on- and off-domain use. Configuring the firewall all across your network can significantly reduce the type and virulence of certain exploits that make their way into your network.

Also linked to this concept of security are Server 2008's levels of componentization. It appears that with Server 2008 and the currently in-development Windows

7 that Microsoft has finally heard the word: monolithic OSs aren't going to work anymore. Server 2008's componentization means that the code for capabilities you aren't using isn't even installed onto the hard drive of your servers. No code means no potential for exploit.

Reason 2: Management

Server 2008 gains the new Server Manager console that aggregates many of the old consoles from previous OSs into a single location. Although it's gotten much of the press, Server Manager is neither the only nor the biggest improvement in the realm of management. The Windows Remote Management service, incorporated natively into Server 2008, will significantly enhance both Microsoft and third-party management toolsets. Windows Remote Shell allows for the remote launching of command-line tools in a way similar to the Sysinternals PSExec tool. Event logs and scheduled tasks both get major overhauls. Even old PerfMon gets a facelift, integrating the old Server Performance Advisor toolkit to become the Reliability and Performance Monitor.

Reason 3: Group Policy

One of the only major changes between Server 2008 RC0 and RC1 was the incorporation of Group Policy Preferences. Preferences allow you the administrator to configure a set of "preferred" settings for computers in your domain. Depending on how you create the preference, it can be enforced on users or you can allow them to adjust their own configuration once set. The single most useful part of preferences is the ability to now easily set registry settings and other common needs without custom administrative templates or login scripts. Want to ensure that WinZip always uses the highest level of compression? Encode that registry key into a preference. Eliminate Acrobat's splash screen? Same process. Assign specific printers, ODBC settings, scheduled tasks, or power options? It's all in there.

Reason 4: Server Core

Windows Server without the Windows. Server Core is a brilliant idea that will likely become a tool for specialized uses. With Server Core, the only console interface is the command line. Its inability to run managed code and lack of GUI means that most of the behaviors that cause system problems, like installing unnecessary software or Internet surfing on the server itself, go away because they simply aren't supported. Server Core's very, very low hardware requirements also will endear it to shops that simply can't afford the constant upgrade cycle. Right now, Server Core needs one major element to make it a big success: third-party application support. But once our antivirus and backup software vendors reformulate their products to work with it, you'll start seeing the Windows form of UNIX in many places.

Reason 5: Hyper-V

Hyper-V is Microsoft's still-in-the-oven virtualization solution that bolts onto Server 2008. Likely unavailable for up to 6 months after the server's RTM date, Hyper-V has one major thing going for it—price. Though Hyper-V has its detractors, Microsoft's $28 pricing model makes it a compelling add. Also announced is a major coup in Microsoft's decision to support managing other hypervisors within System Center Virtual Machine Manager. With this capability, you'll be able to manage your VMware hosts using the same interface as you manage your Hyper-V ones. Integration? Cross-platform support? This is new but very welcome behavior for Microsoft.

Reason 6: NTBackup

A lesser-known update, with Server 2008, the venerable NTBackup gets a host of new features that make it a serious backup solution contender. Now called Windows Server Backup, this tool can do full backups of any server that supported bare-metal restoration. Thus, if a server dies, it can be restored back to like hardware with a single step. Its backups now integrate with the Volume Shadow Copy Service. VSS' snapshots mean no more worrying about open files being skipped in your backups. All of this ties together into a comprehensive solution that ensures all-around better backups.

Reason 7: Domain Controller's Restartable Service

There's plenty to talk about with Active Directory (AD) and its updates in Server 2008. But one feature of great excitement is its reconfiguration to a restartable service. By making AD Domain Services a service, with all the benefits that arrive with such a change, it is now possible to do many kinds of off-line work on AD without having to reboot into Directory Services Restore Mode. Less rebooting means more availability.

Reason 8: Terminal Services RemoteApps

RemoteApps are the Microsoft realization of Citrix Presentation Server's Published Applications, and they're a welcome addition to Terminal Services. With Server 2008's Terminal Services, it is now possible to publish to a user just an application rather than a full desktop. You no longer need to lock down the desktop simply because that's the only option you have for your users. With RemoteApps, you can give them an application experience just like they're used to seeing with those installed locally.

Reason 9: IIS Management

IIS version 7.0 is a big boon for Web site administrators. With this upgrade comes the elimination of the Metabase in favor of an Apache-like file-based configuration mechanism. This movement to file-based management makes the administration of IIS significantly easier, allows for the copying of configuration files between servers, and enables multiple servers to all lean on the same configuration file. Thus, you can update one file and every IIS server simultaneously sees the change.

Reason 10: Installation

Over the years, the Windows Server installation has gotten incrementally more complex, with many questions being asked of the user during the initial installation routine. This was great because at the conclusion of the installation you would have a fairly complete server ready for applications. It was bad because it meant sitting in front of that installation until the bitter end. With Server 2008, the installation goes very far in the other direction. Every question needed as part of the installation occurs at the very beginning, and the number of questions are few. Even if you don't use any of Microsoft's rapid deployment options like the Windows Automated Installation Kit or Windows Deployment Services, you'll find that this version of Server is easier than any other to simply get installed.

Reason 11: Fewer Instances to Upgrade

The first 10 of these reasons are feature-related. But as we all know, an upgrade has a lot more to do with the business motivators than with any set of features. Reason 11 as to why you'll upgrade to Server 2008 faster than any desktop upgrade is the simple reason that there are often fewer server instances than desktop or laptop instances. Adding to this are the relatively few third-party applications installed on your servers in comparison to your desktops. Simple upgrades with few chances for application conflicts like domain controllers, file servers, and infrastructure servers are all excellent candidates for an early adoption. Start with the easy upgrades to get a few instances in your environment.

Reason 12: Little Hardware Upgrades Needed

The last reason is an important one, and one that I hear all the time when talking about Vista upgrades. Vista's hardware requirements are so big that an upgrade will likely involve an expensive desktop hardware upgrade as well. Unlike Vista, Server 2008's hardware requirements aren't all that much different from Server 2003. That means that your existing set of server hardware—which is usually more well-appointed than desktops anyway—will likely handle the upgrade with few problems and no need for extra RAM or more processors.

Summary

So, there you have it—12 reasons why you'll quickly upgrade your servers to 2008. Admittedly, there are some detractors to an upgrade as well. But we're only a few days after the start of the multiple month-long launch parties. We've been waiting for this release for 5 years, and Microsoft has dedicated a lot of effort to making this server release a success. So let's toast a glass to its success and your success in getting it incorporated into your network.

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.

Focal Point

The Hard Facts on Upgrading to Vista

by Don Jones

Look, like it or not you're going to have to make -- or live with, if someone else makes it -- the decision: upgrade to Windows Vista or not? This column isn't another Vista-bashing, nor is it intended to convince you to stay with XP or upgrade to Vista. Instead, I want to look at the hard facts -- indisputable evidence, not opinion -- surrounding this decision. Whether it's your decision or one you'll merely be subjected to, knowing the consequences is absolutely critical.

Ignore the Hype

I'm going to ignore a lot of factors that aren't considered indisputable, hard fact by everyone in the IT industry. Microsoft says Vista is the most secure version of Windows ever. I'm sure it's true, but I have no proof. In any event, Vista may be "more secure" but nothing is "totally secure." I think most businesses have come to grips with the fact that they'll need to constantly manage security, and if you're doing that, it might not matter as much if Vista is all it's cracked up to be. I'm also going to ignore Vista's new user interface -- some love it, some hate it, but it really shouldn't be part of a business decision.

Choose to Stay

Back in the day, there was never a question of whether you'd upgrade to the new version of Windows -- the only question was when you'd make the jump. With Windows XP, however, came a very long rollout; some companies decided just to skip a version and stick with Windows 2000, in fact. That pattern has emerged again with Vista, along with suggestions that companies may wait even longer -- if for no other reason than the resources that a client OS migration requires.

So let's start by looking at the benefits of sticking with XP:

- You know it. It's a tried-and-true operating system (OS) that users are comfortable with and that you probably know how to support quite well. That won't be the case with Vista. You'll need end user and IT pro training to come up to speed, and no amount of training will make you a Vista expert super-fast. In some cases, Vista rearranges how to accomplish common tasks without really offering a benefit, which can frustrate users until they become accustomed to the new way of doing things.

- Everything works. Odds are, all of your software and hardware works with XP, and that definitely won't be the case with Vista. At the very least, you'll probably have some applications to upgrade, and you may have some that just won't work under the new OS.

- It's faster. On the hardware you currently have, that is. Some studies suggest that XP might be faster on leading-edge hardware too. Vista's first service pack was supposed to address performance issues, but as of this writing, the promised improvements haven't materialized.

- Requires no effort. Perhaps the most compelling reason to stay with XP is that it's just easier. Companies today are dealing with a lot of IT projects and something as major and significant as a client OS migration has the ability to bring all other projects to a standstill. Unless there's a clear benefit to upgrading, the argument goes, why fix what ain't broke?

- It's supported enough. Mainstream support for XP will end in January of 2009 with extended support -- mainly bugfixes -- continuing until at least 2012. That's probably enough time for Windows v7 (the version after Vista) to come out, meaning you could play "skip a version" and still get bugfixes for your current OS.

Choose to Switch

Vista isn't all bad news, of course, and there are a number of factual, objective reasons why a migration to it could be a good idea:

- It's more secure. No matter what you're doing to manage security, the fact is that Vista's User Account Control (UAC) and other features make it less susceptible to certain kinds of security attacks -- especially those that rely on end user ignorance and trust -- than prior versions.

- It's supported. This is no minor consideration. Although Microsoft has extended Windows XP's support life, Vista is the "current version" of Windows and will receive the most and longest-lasting support from this date.

- Future purchase compatibility is assured. Anything you buy from here on will work (most likely) with Vista, although it may not work with XP. This is especially significant for hardware: Although companies may purchase new computers pre-loaded with Vista and then install XP before deploying them, that tactic won't last forever. Eventually a generation of hardware will exist for which XP drivers aren't available. This is a concern with software, too. I'm sure the Microsoft Office team has asked themselves when they can safely commit to a version that won't support XP, and I doubt they have a comfortable, solid answer for that, yet -- but they eventually will.

- More features. As with any new version, you get new features that could be important to you, such as vastly expanded manageability through Group Policy, new networking capabilities, new power-management capabilities, and more.

- Tuned for newer hardware. Dual-core PCs didn't exist when XP was born, and it can't run processes across them like Vista can. If your company is one that continually brings in new, up-to-date hardware, you may get the most out of it by running Vista.

Of Course, There's More

Of course, there are a hundred or more subjective reasons to switch or stay, which we could debate endlessly. Although most folks feel that Vista offers a lot of compelling functionality for consumers, they feel it's less of a requirement for corporations -- after all, Media Center features and photo management aren't often high in the IT department's list of requirements. Others feel that under-the-hood improvements in security and stability warrant an upgrade, while still others want to upgrade simply to get the sexy new user interface. These factors will come into your decision, too, but be sure to look carefully at the objective reasons on either side of the argument. Consider that thousands of companies with tens of thousands of computers combined are planning to stay with XP until someone convinces them otherwise, so you'll hardly be alone.

Don Jones is an author and Series Editor for Realtime Publishers.

Practical PowerShell

How to Tell Who Is Logged On

by Jeffery Hicks

The Problem: You need to know who is logged on to a particular desktop or server.

As you might expect, in PowerShell we'll need to use the Get-WMIObject cmdlet to discover the identity of a logged on user. In a perfect world, we could simply use the Win32_LogonSession class and retrieve only interactive sessions.

```
PS C:\> gwmi win32_logonsession -filter "logontype=2"
AuthenticationPackage : NTLM
LogonId               : 240198
LogonType             : 2
Name                  :
StartTime             : 20080112134015.308878-300
Status                :

AuthenticationPackage : NTLM
LogonId               : 250156
LogonType             : 2
Name                  :
StartTime             : 20080112134017.664478-300
Status                :
```

We can use an AssociatorsOf query with the LogonID to return a user name.

```
PS C:\> gwmi -query "ASSOCIATORS OF {win32_logonsession.logonid=240198} where
ResultClass = win32_useraccount" | select Caption,FullName

Caption                                                 Fullname
-------                                                  --------
PUCK\Jeff                                                Jeffery Hicks
```

But if you recall, I had two interactive sessions. The other one turns out to be for VMware. There's nothing wrong with this; it simply adds a little unpredictability to my task. I've also seen and heard of other situations in which querying the Win32_LogonSession falls a little short, and if the session was authenticated with Kerberos, this type of information isn't available.

I prefer to take a more round-about approach. Any interactive user session will have an instance of Explorer.exe running as its shell. All I have to do is find all instances of this process, find the associated logon session, and then the associated user account. It actually is pretty easy with the Get-LoggedOnUser function I created. You can download this function from the Realtime Publishers Web site.

```
Function Get-LoggedOnUser {
    Param([string]$computer="localhost",
    [System.Management.Automation.PSCredential]$credential)

    $errorActionPreference="SilentlyContinue"
    $DebugPreference="SilentlyContinue"

    Write-Debug "Connecting to $computer"

    #define repeated script blocks
    $WMI={Get-WmiObject -query $query -computername $computer -ea "SilentlyContinue"}
    $WMI_Cred={Get-WmiObject -query $query -computername $computer -Credential
$credential -ea "SilentlyContinue"}

    #get process object for explorer.exe
    $query="Select Handle from Win32_Process WHERE Name='explorer.exe'"
    Write-Debug "Executing $query"

    if ($credential) {
        Write-Debug "Connecting with alternate credential"
        $proc=&$WMI_Cred
        }
    else
    {
        $proc=&$WMI
    }

    Write-Debug "Continuing with function"

    #there may be more than one session
    foreach ($p in $proc) {
     # only continue if we were able to get a connection to the computer
     if ($p.handle) {
     Write-Debug $p.handle

     #get Associators Of for logon session
     $query="ASSOCIATORS OF {Win32_Process.Handle='"+$p.Handle+"'} WHERE
ResultClass=Win32_LogonSession"
     Write-Debug "Executing $query"
```

```
if ($credential) {
            $LogonSession=&$WMI_Cred
        }
        else
        {
            $LogonSession=&$WMI
        }

    #get process owner
    #use this method because if the authentication is Kerberos, the logon session
    #class won't return the user account information. The GetOwner method will
    #return an object with Domain and User properties.
    $owner=$p.GetOwner()
    Write-Debug $owner

    #create custom object
    Write-Debug "Creating new object"
    $obj=New-Object system.Object
    $obj | Add-Member -MemberType NoteProperty -Name "Computer" -Value $computer.
ToUpper()
    $obj | Add-Member -MemberType NoteProperty -Name "StartTime" -Value $logonsession.
ConvertToDateTime($logonsession.StartTime)
    $obj | Add-Member -MemberType NoteProperty -Name "UserAccount" -Value ($owner.
domain+"\"+$owner.User.toUpper())
    $obj | Add-Member -MemberType NoteProperty -Name "SessionID" -Value $logonsession.
LogonId
    $obj | Add-Member -MemberType NoteProperty -Name "Authentication" -Value
$LogonSession.authenticationpackage

# Optionally, you may want to include the process ID
#$obj | Add-Member -MemberType NoteProperty -Name "PID" -value $p.Handle
    write $obj

       }
    else
     {
        Write-Debug "Failed to connect to $computer"
     } #end if
    } #end foreach
```

The function will accept a computer name as an input parameter as well as a PSCredential:

```
PS C:\> $jdh=get-credential jdhitsolutions\da-jhicks
PS C:\> get-loggedonuser jdhit-dc01 $jdh | format-table -auto

Computer   StartTime            UserAccount              SessionID Authentication
--------   ---------            -----------              --------- --------------
JDHIT-DC01 1/14/2008 4:08:20 PM JDHITSOLUTIONS\DA_JHICKS 344581    Kerberos
```

The $jdh variable is a stored PSCredential. The function's output is a custom object with the properties you see here.

Let me point out some highlights about how the function works so that not only will you learn a little PowerShell but also you'll be able to enhance the function. The function can accept two input parameters as defined in the Param section:

```
Param([string]$computer="localhost",
    [System.Management.Automation.PSCredential]$credential)
```

I'm following the recommended best practice of casting my variables to a specific type. The $computer variable will also default to localhost if not supplied. This prevents the function from generating an error if the user forgets to specify a computer name.

In fact, I wanted to make this function work as smoothly as possible, regardless of what might happen. Given that the function uses Get-WMIObject, it is likely that the remote computer might not be available or there might be an access denied message. I didn't want those error messages intermixed with the function's output, so I set the $ErrorActionPreference variable to SilentlyContinue.

```
    $errorActionPreference="SilentlyContinue"
    $DebugPreference="SilentlyContinue"
```

No exceptions will be raised and no error messages displayed. You can modify this setting if you want to alter how the function handles errors or if you are troubleshooting. It is for the latter that I also set the $DebugPreference to SilentlyContinue.

Here's why. Throughout the script I've added lines like this:

```
    Write-Debug "Connecting to $computer"
```

With the debug preference set to silently continue, these messages are never written to the screen. But when debugging or troubleshooting, I can change the setting to Continue. Now, when the function is executed, all the Write-Debug messages will be displayed. I can use these messages as trace code to follow what my function is doing every step of the way. I added the trace lines as I was developing the function. There's no performance penalty in leaving them in because you might need them or I might need them again should I decided to modify the function. This is a good technique to use

in all your PowerShell script development. Add Write-Debug lines to help trace code execution and turn on or off the debug pipeline with $DebugPreference.

Continuing on, in the function, I've defined two script blocks that will be used repeatedly throughout the script. All I have to do is modify the $query variable.

```
#define repeated script blocks
    $WMI={Get-WmiObject -query $query -computername $computer -ea "SilentlyContinue"}
    $WMI_Cred={Get-WmiObject -query $query -computername $computer -Credential -ea
"SilentlyContinue"}
```

The second version will connect to the specified computer with the user-supplied PSCredential. In both instances, I'm setting the ErrorActionPreference cmdlet common parameter to "SilentlyContinue". Again, I don't want any errors to interrupt the function.

The first major step in the function is retrieving the Explorer.exe process on the specified computer.

```
#get process object for explorer.exe
 $query="Select Handle from Win32_Process WHERE Name='explorer.exe'"
 Write-Debug "Executing $query"

 if ($credential) {
    Write-Debug "Connecting with alternate credential"
     $proc=&$WMI_Cred
}
 else
 {
     $proc=&$WMI
 }
```

I've defined an appropriate WMI query and executed with the appropriate script block, depending on whether an alternate credential was specified.

The query may return more than one instance if there are several simultaneous sessions, as in the case of a remote desktop session. Therefore, I'll use a foreach loop to look at every instance.

```
foreach ($p in $proc) {
```

Now, it may be that I was never able to connect to the specified computer for any number of reasons. If so, then $proc will be empty; therefore, the first thing I need to do is verify I have a handle property to query. I'll need this property for a subsequent query, so if it doesn't exist, there is no reason to keep going with the function.

```
      if ($p.handle) {
```

If $p.handle exists, the main part of the function will continue; otherwise, the else portion is reached

```
     else
      {
         Write-Debug “Failed to connect to $computer”
}
```

I’ve chosen not to return any error messages, but you might want to modify this line.

Assuming there is a process handle, I’ll next run an AssociatorsOf query to find the associated Win32_LogonSession class.

```
     $query=”ASSOCIATORS OF {Win32_Process.Handle=’”+$p.Handle+”’} WHERE
ResultClass=Win32_LogonSession”
     Write-Debug “Executing $query”

        if ($credential) {
            $LogonSession=&$WMI_Cred
        }
        else
        {
            $LogonSession=&$WMI
        }
```

One useful piece of information this class will provide is the session’s Start Time. Normally, this class would let me find an associated Win32_UserAccount class, but when the logon session is authenticated with Kerberos, I don’t get that information. That’s easy enough to work around, though. I’ll merely invoke the GetOwner method of the current process object:

```
    $owner=$p.GetOwner()
```

Now I have enough data to start returning information to the user. Because I want the function to participate in the pipeline, I’ll need to create a custom object:

```
    $obj=New-Object system.Object
```

The $obj object is a generic blank object. But by piping it to Add-Member, I can define my custom properties:

```
    $obj | Add-Member -MemberType NoteProperty -Name "Computer" -Value $computer.
ToUpper()
    $obj | Add-Member -MemberType NoteProperty -Name "StartTime" -Value $logonsession.
ConvertToDateTime($logonsession.StartTime)
    $obj | Add-Member -MemberType NoteProperty -Name "UserAccount" -Value ($owner.
domain+"\"+$owner.User.toUpper())
    $obj | Add-Member -MemberType NoteProperty -Name "SessionID" -Value $logonsession.
LogonId
    $obj | Add-Member -MemberType NoteProperty -Name "Authentication" -Value
$LogonSession.authenticationpackage
```

Once the object's properties have been defined, I write it to the pipeline:

```
    write $obj
```

This process repeats for every instance of explorer.exe. Here's how it might look from the console.

```
PS C:\> get-loggedonuser

Computer       : LOCALHOST
StartTime      : 1/14/2008 9:19:29 AM
UserAccount    : PUCK\JEFF
SessionID      : 382059
Authentication : NTLM
```

What if you want to check several computers? Here's an example using alternate credentials:

```
PS C:\> $admin=get-credential mycompany\administrator
PS C:\> "exch07","exch01","mycompany-dc01" | % { get-loggedonuser $_ $admin } | `
>> format-table -auto
>>

Computer       StartTime            UserAccount             SessionID Authentication
--------       ---------            -----------             --------- --------------
EXCH07         1/14/2008 2:50:11 PM MYCOMPANY\BIV           1248099   Kerberos
EXCH01         1/14/2008 2:42:37 PM MYCOMPANY\JHICKS        33224500  Kerberos
MYCOMPANY-DC01 1/14/2008 2:40:02 PM MYCOMPANY\ADMINISTRATOR 4251839   Kerberos
```

Because PowerShell treats every comma-separated list as an array, I can pipe it to the ForEach-Object cmdlet, which has an alias of %. For every name in the list, I call the Get-LoggedOnUser function passing it the current name in the pipeline and my stored set of credentials. That output is then piped to Format-Table.

Of course, I might have a much longer list and maybe I want to sort the results and export them to a CSV file. I could use an expression like this:

```
PS C:\> cat servers.txt | % { get-loggedonuser $_ $admin } | sort StartTime  | export-
csv loggedon.csv
```

Or suppose I want to find all the servers logged on by Administrator:

```
PS C:\> cat servers.txt | % { get-loggedonuser $_ $admin } | where {$_.UserAccount
-match "Administrator"}
```

Because the function's custom object is in the pipeline, you can slice and dice the information just about any way you please. Enjoy!

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.

Exclusively Exchange

Managed Folders Control Message Lifespan

by J. Peter Bruzzese

If you are a veteran Exchange admin, you might recall that mailbox quotas and mailbox management settings (handled by the Recipient Update Service—RUS) in previous versions of Exchange offered limited capabilities in terms of email records management. Going to the next level, Exchange 2007 implements new features that focus on an organization's Email Life Cycle (ELC).

Exchange 2007 utilizes a technology termed Messaging Records Management (MRM) to assist administrators in meeting legal requirements or company policy through managed folder settings. Essentially, you can intervene in the normal process of users handling the message lifespan of their own mailboxes by establishing retention times and/or journaling.

This article discusses the ability to manage mailbox content settings through folder management, including the ability to journal (copy) messages in this way. However, Journal Rules enforced on the Hub Transport level are a more substantial method of recording all messages that flow in and out of your organization. Journal Rules will be discussed in a future article.

The Purpose of Managed Folders

There is no one purpose behind the way you manage folders within your organization. Your motive in doing so may be something as serious as compliance toward government regulations that requires you to retain all messages that fall under the rules pertaining to electronic communications, or you might simply want to prevent users from keeping items in their Deleted Items folder for decades. You might even want to allow users to have a part in the process by giving them the ability to classify messages on their own, utilizing special folders where they can sort mail according to personal needs for retention.

The "folders" referred to here can be one of two types: default folders or custom folders. You can manage your default folders (Inbox, Sent Items, Deleted Items, and so forth) or you can create an entirely new folder, a custom folder, to manage and affect policy. A custom folder created for this purpose is obviously leaning more toward allowing the users to have a greater share in the process because this folder will suddenly appear among their Outlook folders and you will need to instruct them as to its purpose and use.

> Managed folders are unique in that they cannot be moved, renamed, or deleted. With custom folders, if you choose to create these, you can direct your users to employ them for the retention/journaling purposes or you can enforce messaging rules that move messages automatically into default/custom folders.

Thus, you can utilize managed folders via various methods. Perhaps you want to permanently delete messages that have reached a threshold of 100 days in the Deleted Items folder for your users. Going one step further, perhaps you want to remove only voicemail, fax, or calendar items from Deleted Items. In the process, you might want to journal (copy) those messages prior to their deletion. Or, utilizing good faith in your users and their training, you might want to create a custom folder where users place important content that you ensure is retained through managed folder settings. All of these possibilities highlight the new functionality offered by Exchange 2007 managed folders.

The Process of Implementing Managed Folder Policies

In truth, the process for implementing managed folders is a bit backwards and you might find it initially confusing. Let's clear up some of that confusion. You can configure these settings through the Exchange Management Console (EMC) or through the Exchange Management Shell (EMS), and, in fact, you probably will use both. You will use the EMC to create your managed policies and the EMS to enforce those policies in bulk.

From the EMC, under Organization Configuration | Mailbox, select the Managed Default Folders tab (see Figure 1) or the Managed Custom Folders tab. You can use the default folders to proceed to the next step. However, you might want to create another instance of a default folder (not that users will see another instance in their display, simply to create multiple content settings). Or you can create a new custom folder. Both actions can be performed through the Actions pane.

Right-click the folder for which you want to establish settings, and choose New Managed Content Settings (see Figure 2). If you select the folder, this option will also appear in the Actions pane. Follow the prompts and provide the following information:

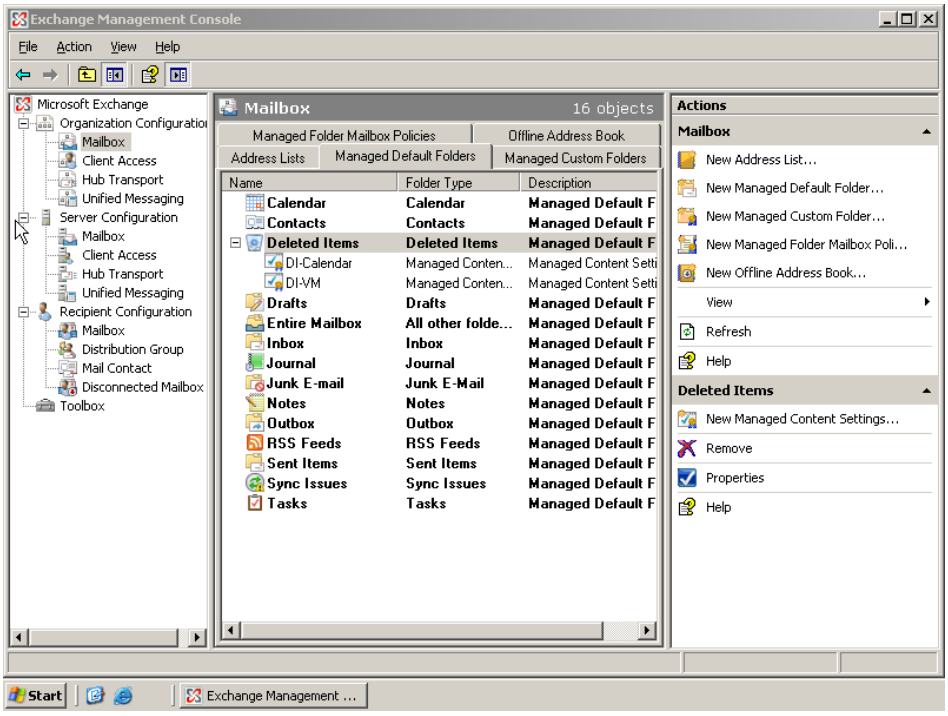

Figure 1: Working with default folder managed folder settings.

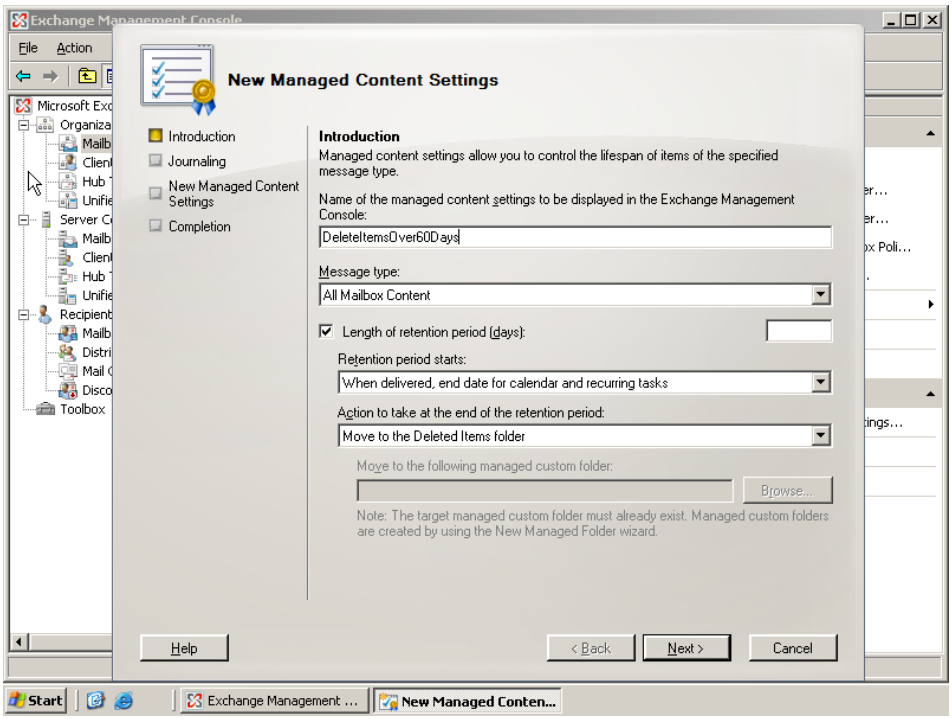

Figure 2: Configuring new managed content settings.

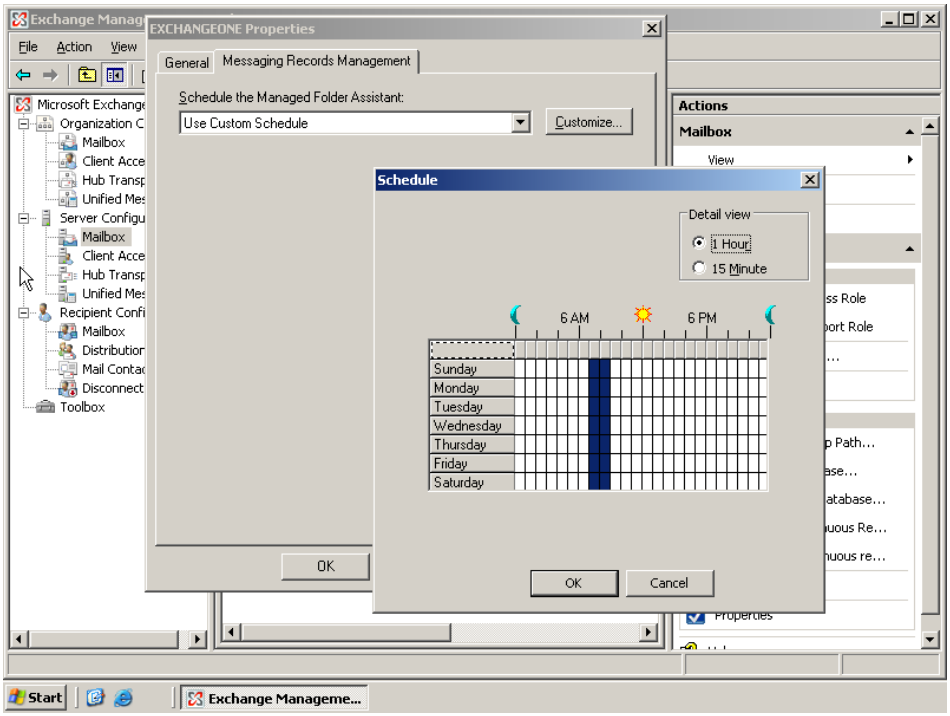

Figure 3: Configuring your Managed Folder Assistant to run.

- Name the setting something memorable.
- Choose the message type. This feature is excellent for focusing on all mailbox contents or certain items such as voicemail, notes, calendar items, and the like.
- Select the Length of Retention check box if the purpose of these settings is to control length of time. If so, provide a number of days to wait until springing into action. At that point, you can determine when the retention time begins and what should happen to that message/voicemail/calendar item when the time is up. Where does it go? Is a copy made as well? You provide the answers and then click Next.
- Journaling options are also requested if you plan to retain copies of your managed content. Will you journal toward an email address, a SharePoint Server, or elsewhere?

Once you have all your managed content settings, you can go to the Managed Folder Mailbox Policies tab to create policies that contain one or more of the settings you've established, both default and custom.

Finally, you can apply those managed folder mailbox policies to users' mailboxes. But don't forget to schedule the Managed Folder Assistant to run. This is the caveat to this technology that is causing much turmoil: The Managed Folder Assistant is in a completely different location from where we've been working. You have to go through Server Configuration | Mailbox, select the server you want, and go into Properties (see Figure 3). The default is that it never runs, so you need to set up a schedule to get it going. (For a quick manual trigger, through the EMS, type

```
Start-ManagedFolderAs-
sistant
```

Once the Assistant is scheduled to run, it will enforce those policies you created.

Messaging Records Management

Avoid having your company slammed with a million dollar fine for mismanaging records. Or simply avoid having storage space wasted because users are pack-ratty about holding on to every piece of mail. With the tools in Exchange 2007, admins have a better opportunity to enforce messaging records management. Managed folder settings are one of the many new improvements in this area that Exchange has to offer.

J. Peter Bruzzese is an MCSE (NT,2K,2K3), and MCITP: Enterprise Messaging Administrator, and MCT. His expertise is in messaging through Exchange and Outlook. J.P.B. is the Series Instructor for Exchange 2007 for CBT Nuggets. His latest book is "Tricks of the Vista Masters". He is co-founder of ClipTraining.com, a provider of short, educational screencasts on Exchange, Windows Server, Vista and Office 2007. You can reach Peter at jpb@cliptraining.com.