Realtime
publishers

May 2009
Volume 2, Number 5

Windows Administration

in Realtime

2 **Letter from the Editor**
Fanboyism: Does It Help or Hurt Us?

3 **Answers from the Experts**
The Difference Between a Full and Log Backup in SQL Server

5 **Product Review**
JDiskReport

6 **Introduction to Network Access Protection**
A New Tool for Maintaining Machine Health Standards
By: Steve Murawski - Discover how the NAP framework provides you with the capability to enforce the configuration and health status of machines accessing your network.

10 **The Deep Dive**
Data Leakage—Is Your Company at Risk?
By: Lori Cotton - Your data is your business. Is that business adequately protected through the prevention or control of access to specific devices, data, information, assets, and facilities?

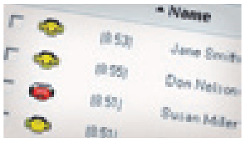
13 **Practical PowerShell**
Uptime Availability
By: Jeffery Hicks - Create a report that tells you when a computer was available and for how long.

20 **Exclusively Exchange**
Unified Messaging Planning and Best Practices
By: J. Peter Bruzzese - Gather the tools and tips you need to successfully plan for, install, and configure UM.

Avaya Unified Communications

You have a reduced IT budget, an outdated infrastructure, and 1,500 users who don't have time for downtime.

Communication Manager
Consolidate your infrastructure and save money.



Meeting Exchange
Travel less and collaborate everywhere.

Proactive IP Support Services
Detect network problems in real-time.

AVAYA
INTELLIGENT COMMUNICATIONS

Visit **www.avaya.com/IT** to see how Avaya simplifies business communication.

Letter from the Editor

Fanboyism: Does It Help or Hurt Us?

by Greg Shields

Technology fanaticism. Fanboyism. Really, really, really liking a particular product, vendor, or service. All of these are common in our industry. You've heard the lines before: "UNIX is a superior operating system." "I bleed red for Novell." "Mac's don't get viruses."

Funny how many of these get overcome by events as time goes on. I've recently found myself growing aggravated by the amount of fanaticism coming out of virtualization circles these days. Not long ago, I penned a passionate piece over at the www.realtime-windowsserver.com community titled *NEW RULE: Your Use of the Term "Virtualization" Must Be Platform Agnostic* (read it here: http://www.realtime-windowsserver.com/virtualization/2009/02/new_rule_your_use_of_the_term.htm). In that post, I waxed philosophic about the internal linkage between one particular vendor's virtualization technology and the concept of virtualization itself, as if virtualization defined this product and the product defined the concept. I wrote:

> In my opinion, this unconscious one-to-one linkage between a *tactic* and a *technology* is dangerous. Virtualization is a concept, not a product. ESX is a product. There is danger in limiting one's focus to only a single vendor or product as the sole solution for a particular concept. The same held true in the Microsoft vs. Novell days, with the Novell crowd having a remarkably similar impression in their speech (and presumably minds) that "directory services" and "Novell" were one-to-one. In the end, Novell lost that war.

In my nearly 15 years of experience in the IT industry, I've had the opportunity to work with many individuals. Virtually all of those people are hard-working individuals who just want to do right for their IT environment. But many choose to go down the road of single-mindedness when it comes to particular technologies or technological solutions, even when better (cheaper/faster/more user-friendly) options exist. Unfortunately, in almost every case, the "winner" was the technology whose supporters shouted the most.

With so many potential solutions available for solving problems in our industry, there's always a layer of politics that surrounds us. Yet it is that layer of politics that sometimes forces us down the road of bad solutions and unsuccessful projects. In the same vein, though, you do sometimes have to shout if your "better" solution is to be selected. Fanboyism, is it good or bad? Often, the answer is dependant on the social environment that surrounds the technological environment.

Have you had an experience where fanboyism has impacted the selected solution for an IT problem? Was it helpful, or did it hurt the project? Do you believe that technology fanaticism of any sort is helpful? Drop me a line at gshields@realtimepublishers.net and let me know. ◆

Answers from the Experts

The Difference Between a Full and Log Backup in SQL Server

by Don Jones

Q: In SQL Server, what's the difference between a "full" backup and a "log" backup?

A: This requires a bit of background knowledge about how SQL Server works. Or, as Ricky would say, "Lucy, you got some 'splainin' to do."

When SQL Server makes a modification to a database, it goes something like this: First, SQL Server writes a copy of the modifying query to something called a transaction log. This isn't exactly a text file full of SQL language queries, but it's not far off from that. It's basically a record of every modifying query that SQL Server has run.

SQL Server stores data in 8kb-chunks called *pages*, and so next, it grabs whatever pages need to be modified off the disk. It modifies them in memory, but it *does not save them to disk* right away. After all, those pages might be modified again in the near future, so it would be a waste of effort to save them to disk right away. But this creates a problem if SQL Server crashes for some reason, as right now, all those changes exist only in volatile RAM.

Eventually—usually on the order of seconds for a busy server—SQL Server will write those pages back to disk. When it does so, it goes back to the transaction log and marks those items as "saved to disk," or *committed* in SQL-speak. When SQL Server is shut down normally (for example, not a crash), it makes sure all transactions are committed before it actually allows itself to stop.

Let's talk briefly about what happens when SQL Server starts up: It checks the transaction log of every database to see whether there are any *uncommitted* transactions. If there are, it briefly puts the database into *recovery mode*, starts reading those uncommitted transactions from the log, and starts re-executing those transactions—in effect "replaying" all the changes that were lost due to a crash.

So a *log backup* is literally just a backup of the transaction log. These are super-quick because logs usually aren't that large. A normal log backup also clears out the active log of any committed transactions because the data in it is now safely backed up—and that helps keep the log files small. A *full* backup grabs all the data in the database, and under normal circumstances, also clears the active log of any committed transactions. There's also an *incremental* backup, which grabs all the data since the last full (or incremental) backup, and clears the log of any committed transactions.

So in the event of disaster, here's what you do: You restore the latest full backup, using a command (or selecting a check box) that tells SQL Server not to launch recovery mode yet. Then you restore any incrementals made since that full, again telling SQL Server to hold off on the recovery train. Last, you restore any logs made since the most recent incremental—and when you restore the *last* of those logs, you give SQL Server the go-ahead to begin recovery. It starts running through the transactions in those log files, bringing the database up to speed. The database is actually usable during this phase, although performance may be a little sub-optimal.

Those log backups are a great tactical idea because they're so small: I have customers who can only do a full backup on the weekends, but they grab log backups nearly every hour. Sure, restoring means getting a *bunch* of log backups from tape or near-line storage, but it's better than losing everything, and that way, their biggest risk is losing an hour of work. ◆

Don Jones is a co-founder of Concentrated Technology. Join him and cohort Greg Shields for intense Win2008 and Windows PowerShell training—visit ConcentratedTech.com/class for more details. Ask Don a question by visiting ConcentratedTech.com and using the "Contact" page.

Product Review

JDiskReport

by Eric Schmidt

Large enterprises generate gigabytes of data every month, and all that data needs to be stored and organized. For many organizations, the default solution is to buy more and more disk space; however, this approach becomes expensive and cumbersome very quickly. Couple this reality with shrinking IT budgets that keep file servers, SANs, and backup solutions from growing indefinitely. Organizations must look at ways to identify inefficient use of their limited storage resources. This is where the free tool JDiskReport by JGoodies can be an effective solution for analyzing and documenting what, how much, and where stuff is being stored.

JDiskReport is a small Java-based utility that will interrogate a file system and then report back statistics on what it finds. The data that it collects can be viewed from multiple perspectives depending on what is being looked for. For example, the size perspective can be used to gain an understanding as to the sizes of files that are on the volume. Figure 1 shows a graphical representation of files broken down by size. It also shows how much space a particular file size range is consuming on the disk.

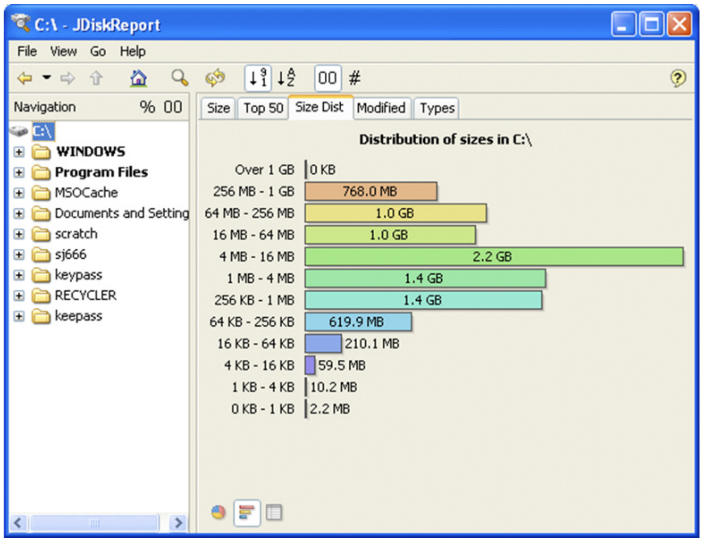


Figure 1: Sample file size distribution report.

Another feature when looking at the report is the ability to drill into the file system to see the same breakdown for a particular folder. In situations in which the volume has run out of space, this view can be used to focus on the largest files so that disk space can be freed up quickly.

The modified view organizes files based on the last modified time. This can be helpful with archiving, as it will show what files haven't been modified for an extended period. An example of when this would be useful is evaluating the viability of a tiered storage or archive solution.

The third perspective JDiskReport offers is file type. With file type, you can quickly see how much space a particular file type is consuming. One of the best uses of the file type perspective is looking for non-business-related data (music, movies, pictures, and so on) that is being stored on company file servers. By leveraging the drill-down feature, you can quickly identify unapproved files so that appropriate action can be taken.

JDiskReport's usefulness is not limited to file servers. It is also very helpful on the local disks of servers and workstations to provide insight for troubleshooting issues. In situations in which an application fills a disk with cache files, JDiskReport can make it easier to identify and locate those files.

JDiskReport is an excellent utility that every systems administrator should have in their toolkit. The only negative is that it does have to be installed, therefore making it less portable. The utility is small and provides a wealth of information about a file system from multiple perspectives, which can be used for troubleshooting, policy enforcement, and archive analysis. ◆

Eric Schmidt works as Enterprise Microsoft Security Technologist, with Honors, for Raytheon Company and has worked in Information Technology for 13 years. Eric has a Masters degree in Computer Information Technology and has developed extensive experience in systems administration, engineering, and architecture specializing in Microsoft Active Directory and Systems Management. Eric has been well recognized throughout his career for his contributions to designing and implementing enterprise-wide solutions using Microsoft Windows-based technologies.

Introduction to Network Access Protection

by Steve Murawski

Have you ever had a user come back from an extended period off-site with a laptop and see your intrusion detection system turn red or watch the network traffic exiting your network shoot up? Perhaps a vendor came to work on-site and 2 weeks into the engagement, you can't send email to anyone as your domain has been blacklisted? Does the thought of a remote user using a VPN client from a home computer give you night sweats? Network Access Protection (NAP) in Windows Server 2008 might help you sleep a bit better and help your pager stay a bit quieter.

Windows Server 2008 contains a new tool for maintaining machine health standards on your network. Maintaining these standards is becoming increasingly more difficult as more methods of remote access are required and device portability continues to increase. NAP is a framework that provides you with the capability to enforce the configuration and health status of machines accessing your network. The important part of the previous sentence is "machines accessing your network." As a Windows administrator in the current mobile and connected environment, you have to deal with people accessing resources from managed and unmanaged computers via various VPN clients, Terminal Services Gateways, and laptops that are not continuously on your network. The NAP framework provides you with options for limiting the risk factors in dealing with these situations.

NAP provides several methods of controlling access to your network. IPSec enforcement is the most flexible and complex of the enforcement methods. IPSec enforcement requires access to protected servers and computers to be done through IPSec and uses a domain Certificate Authority (CA) to grant access to only those machines that meet the health requirements. The second type of enforcement provided by NAP is 802.1X enforcement. 802.1X enforcement uses 802.1X and dynamic VLANs or access control lists (ACLs) on capable network hardware to restrict network access to non-compliant computers. DHCP enforcement controls access by providing IP addressing to a limited network for clients whose health status fails validation. Fourth, there is VPN enforcement. VPN enforcement leverages Routing and Remote Access to limit the access of a machine to the appropriate remediation server(s) until it is compliant, at which point it would be granted the appropriate access. The fifth method is the Terminal Services Gateway enforcement, where the TS Gateway determines access based on the health of the client machine. The TS Gateway enforcement will work only with Windows Vista or Windows XP Service Pack 3 (SP3) clients. Windows Server 2008 will not work as a NAP-capable client in a TS Gateway scenario. Finally, there is a level called "No Enforcement." This level is ideal in the planning stage, as it allows you to gather data and report on the current status of your network. By running with "No Enforcement," you can identify problem computers and test your remediation capabilities.

The NAP framework is made up of several components. System Health Agents (SHAs) validate the compliance status of the computer accessing your network. Also working client side are the Enforcement Clients, which request access to the network, transmit the health information to the appropriate Enforcement Server, and communicate the resulting status of the health validation. System Health Validators (SHVs) are the server-side complement to the SHAs, and they provide the minimum requirements for the health of the client computer on your network. The Health Policy Server, the Health Requirement server, and the Enforcement Server use the response from the SHAs and the requirements from the SHVs to take the appropriate action and either grant full network access or take the proper enforcement action. Rounding out the NAP framework are the Certificate Authority Server and Remediation Server.

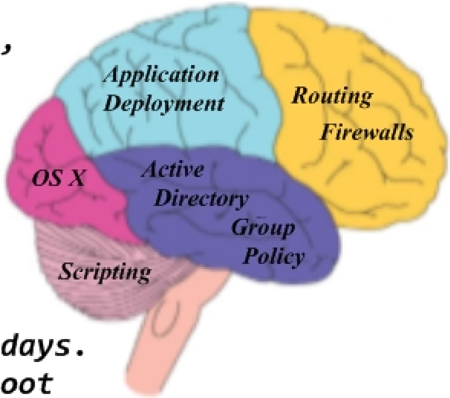
The first component of the NAP is the SHA. NAP requires SHAs to verify the configuration and status of client computers. Currently, Windows XP SP3, Windows Vista, and Windows Server 2008 have a SHA that monitors the status of the Windows Security Center. The SHA monitors the firewall, virus protection, spyware protection, automatic updates, and security updates. There are additional

SHAs available from Microsoft (via System Center Configuration Manager and Forefront Client Security). Third-party vendors can also provide SHAs that can monitor additional health criteria.

SHAs each prepare a Statement of Health, which is then transmitted by the Enforcement Client to the Enforcement Server, or in the case of 802.1X enforcement, the Enforcement Point, which is an 802.1X-capable network device. An Enforcement Server is a server running the NAP Service and has the appropriate services enabled for the type of enforcement. For IPSec enforcement and "No Enforcement," the server must be a Health Registration Authority running IIS and Network Policy Server. VPN enforcement requires the Routing and Remote Access service. DHCP enforcement necessitates the Network Policy Server as well as the DHCP service. The TS Gateway enforcement requires that the Network Policy Server be running on the TS Gateway server. The Enforcement Server or Enforcement Point passes the Statement of Health to the Health Policy Server for validation.

SHVs provide the Health Server and the Health Requirement Server the criteria with which to evaluate the reported status of the client. Each SHA will have a corresponding SHV. The Health Policy Server is a server running the NAP Service and is configured to evaluate Statements of Health. The Health Policy Server processes the validation of the Statements of Health that were received by the Enforcement Server or Enforcement Point. If the SHV requires additional information, you may need a Health Requirement Server. Unlike the Health Policy Server, which is a part of the NAP role, a Health Requirement Server is an outside source of validation (like an antivirus signature server or other third-party service). The default SHVs (firewall, virus protection, spyware protection, automatic updates, and security updates) only require a Health Server for configuration.

If a NAP client fails to have their Statement of Health validated, you can set up Remediation Servers. If maintaining a current patch level is a requirement and a computer is behind on its patch level, access to a Windows Server Update Services server or Windows Update might be allowed. Like the Health Requirement Server, Remediation Server is a role filled by an outside service, not a feature of NAP. After remediation, the Enforcement Client will re-send the Statement of Health to the Enforcement Server or Enforcement Point.



After a NAP client's Statement of Health is validated, the Health Policy Server will notify the Enforcement Server or Enforcement Point to grant access. The NAP client will continue to monitor the computer for a change in state, in which case it will attempt to validate its current status with an Enforcement Server or Enforcement Point. The health status is also re-evaluated when the NAP Agent service is restarted. There are several other criteria for access renewal as well, depending on the type of enforcement. The NAP client will attempt to renew its health certificate 15 minutes before it expires for IPSec enforcement. For DHCP enforcement, the health status is re-evaluated halfway through the DHCP lease. VPN enforcement dictates evaluation when a VPN session is established. 802.1X enforcement re-authorizes the health status of the client when it re-authenticates the 802.1X connection. In addition to these enforcement specific evaluations, certain SHVs can specify a validity period for the Statement of Health, and if the NAP client does not provide a satisfactory Statement of Health before that validity period is over, the client will be considered non-compliant.

NAP provides three levels of access restriction for all the methods of enforcement other than IPSec. The first level of network access allowed is full access. Just like it sounds, the client's access to the network is not restricted by NAP (but it could be impacted by other access control methods). Full access would be granted to computers that meet all the NAP requirements. The next level of network access is full access for a limited time, also called deferred enforcement. NAP clients are notified that they are not compliant. They are notified that they have until the specified date and time to come into compliance or their access is limited. The final access level is restricted access. This is the level of access given to non-compliant computers. The method of limiting access is determined by the method of enforcement.

IPSec access restriction is more complicated. With IPSec enforcement, access is controlled at a peer-to-peer level rather than a network level. Non-compliant computers will not be able to communicate with protected computers. Configuration of protected computers is done via Group Policy (IP Security Policy Management for Windows XP and Windows Server 2003, or Windows Firewall with Advanced Security for Windows Vista and Windows Server 2008). When setting up IPSec enforcement, computers that are not NAP capable (Windows XP SP2 and earlier and Windows Server 2003) will need to be issued NAP exemption certificates in order to communicate with protected clients.

NAP provides a logging and reporting infrastructure that can be used in your planning and evaluation for deploying NAP in your environment. On the Health Policy Server, all access events are logged either in text or to a SQL Server. Text logging is better than nothing, but the preferred method would be to log to a SQL Server with Reporting Services. The lowest barrier to entry would be to use SQL Server Express on your Health Policy Server. Another option would be to use a full install of SQL Server 2005 or 2008 on your Health Policy Server. Finally, the most dangerous option would be to log to a remote SQL Server instance. This option is dangerous because if the Health Policy Server loses its connection with the SQL Server, it will refuse any network access request. After logging the request data to SQL Server, you will be able to use Reporting Services to display the data that you need.

Just the basic functionality built-in to NAP can satisfy a number of scenarios, but if you use System Center Configuration Manager 2007, the configuration scenarios you can validate increase dramatically. You are able to define NAP policies that validate the patch level of any software managed by System Center Configuration Manager. Integrating System Center Configuration Manager with NAP allows System Center Configuration Manager to serve as a Health Requirement Server and a Remediation Server, identifying and providing needed software updates to non-compliant clients. This enables some interesting scenarios. Normally with System Center Configuration Manager, the client computer scans periodically to see if any updates for installed software are available. When you integrate System Center Configuration Manager with NAP, you can change that dynamic. Now, every time the NAP client authenticates, there is a check of the software update status. If there is a critical update for an application in your environment, NAP can be employed as a method for ensuring deployment in a short period of time after a computer becomes active on your network. You may be wondering if System Center Essentials will fill this role as well, but it does not integrate with NAP.

Another Microsoft product that fits right in with NAP is Forefront Client Security. With the provided NAP integration kit, additional SHAs and SHVs are made available to help evaluate the clients' health status and verify that protection is in force and up to date. If not, the Forefront server can act as a Remediation Server and help bring the client back into compliance.

If you already have an investment in Cisco Network Admission Control (NAC), NAP can fit right in. Combining NAP with NAC allows you to take advantage of an existing health evaluation infrastructure and add to it the flexibility of NAP. NAP can be deployed in-line with NAC, in addition to NAC, or in replacement of NAC. There are a number of considerations when looking at a NAP-NAC solution. A combined NAP-NAC solution supports only Windows Vista SP1 or Windows Server 2008 as clients.

Additional third-party vendors are adding support for the NAP framework. As these vendors release products supporting NAP, maintaining configuration standards on your network becomes even more manageable. Microsoft maintains a list of these vendors at http://www.microsoft.com/windowsserver2008/en/us/nap-partners.aspx.

As portability of devices and remote access scenarios continue to rise, maintaining quality control on the devices accessing your network becomes increasingly more difficult. NAP is a viable tool for IT professionals to help establish a baseline for secure access to their networks. NAP provides you with tools to control access to your network via IPSec, DHCP, 802.1X, VPN, and TS Gateways from a central configuration point. With the reporting infrastructure built in, you can use NAP to help plan your deployment and monitor its effectiveness throughout its life cycle.

Now you can see the scenario… you've implemented NAP. Your high security data center is protected by IPSec policies that allow only authenticated known clients to access it. You no longer fear unpatched computers being granted access to your network via a TS Gateway or your wireless network. Your wireless network provides limited access to Windows update and certain services such as your Help desk ticketing system and antivirus updates for non-compliant computers. And, your guest network provides limited Internet access via a different, locked-down router to computers that don't meet your standards. You might just knock off a bit early this evening. ◆

Steven Murawski is an IT Specialist for a municipal law enforcement agency as well as an independent instructor and consultant. Steven's experience includes designing secure distributed networks, developing IT automation strategy, and implementing quality service desk operations, focusing primarily on Microsoft technologies. Steven specializes in working in small to medium-sized environments, leveraging the built in features of the Microsoft platform and open source technologies to support IT operations. Steven is a member of the PowerShellCommunity.org advisory board, which is a community-driven site dedicated to those learning and using PowerShell. As a co-host on the Mind of Root podcast, Steven participates in weekly efforts to educate himself and others regarding various topics of interest to systems administrators and IT professionals.

The Deep Dive

Data Leakage — Is Your Company at Risk?

by Lori Cotton

The news of today is fraught with stories of data leakage that impact companies and often hundreds, if not thousands, of innocent consumers by creating a fissure in their financial foundation. News that an unknown has not only accessed but disseminated personal, sensitive information can put nest-eggs and credit ratings at risk. It can damage corporations, reputations, trust, and confidence beyond repair. Data leakage occurs far more frequently than the general public is made aware of, often with effects that are prolonged and devastating. Even more frequent are breaches that go unnoticed, and could escalate before they are addressed.

The threat of data leakage is multi-faceted. Proprietary information in the hands of the unauthorized can incur costs in terms of dollars, reputation, and competitive edge. The ability to prevent data leakage can mean the difference between success and failure. Although stories of devastating losses proliferate the market, many of these leaks could have been prevented, were a comprehensive solution in place.

The topic of data leakage encompasses a very broad and deep array of risks, ultimately distilling to the prevention or control of access to specific devices, data, information, assets, and facilities. Many of these topics overlap, several may seem redundant, but any sensitive matter that is not protected adequately is a risk to the livelihood of business or personal matters. It is also critical to recognize that the threats could be internal or external in nature.

The Nature of the Problem

Are you sure your most sensitive information and data are safe? The nature of data leakage could stem from the lack of adherence to corporate security policies or the lack thereof.

Intentional or unintentional breach of security policy is a considerable risk. Whether referring to the electronic breach of security via unauthorized communication of proprietary or sensitive information outside the network, or breaching physical security policy simply by holding the door open to a stranger who seems honest enough, are both a compromise to the security of the business. Intentional breaches often occur at the hands of disgruntled employees or those looking for a means for more easily circumventing the system. Unintentional breaches are often a result of policy ignorance.

In general, divulging proprietary or sensitive information to others is a risk that can have far-reaching detrimental effects. Generally, an email to a colleague or a message on a chat board has become such an emergent and destructive issue that many companies have implemented stringent traffic-monitoring solutions to ensure that they are made aware of proprietary or sensitive information sent across unapproved channels. Alternatively, if proprietary or sensitive information is sent across approved channels but without encryption enabled, that information is left exposed. To determine your company's level of risk, evaluate the following factors.

Personal and/or Unauthorized Use of Company-Issued Hardware

Telecommuting as a green approach to the work environment puts a laptop in the hands of the employee, who then works from a location remote to the corporate office. Although this approach saves time and money, the farther the equipment is from the corporate office, the less control the corporation has over its use. Although telecommuting has obvious merits to both the employee and the employer, if the hardware is used for purposes beyond that which is specified in security policy, that use puts sensitive data and information at risk as new applications and routes of communication are established that might not be secure. Using your work computer on a cable modem for personal use outside of the VPN? Anyone on the network can find you.

As another example, how many of us use USB devices for the alternative, mobile storage that used to be a floppy disk? Do you know where all your USB devices are, or have been? Now how about all the USB devices of your employees? Some companies are locking down USB ports or requiring that specific USB devices are used, then requiring encryption on them, which render the data and information on them useless if found or used by another individual.

Use of Unapproved Software and Applications

Many applications will pull data from your computer to share with another source, or place in another location, such as money-management software or budgeting applications. Unapproved software is not monitored or reviewed by corporate IT services, and can contain bugs or expose a computer to malware or a virus.

Changes to Security Settings

Sitting in a Wi-Fi zone at the airport or a major city and changing your Internet connectivity settings in order to access the Internet outside of the corporate VPN may seem like a good idea when you are traveling and behind on responsibilities. However, this innocent act by an anxious workaholic exposes a computer chock-full of proprietary data and information to a network of strangers.

Physical Facility Breach

Someone can't find their badge, seems to be in a hurry, or is friendly enough to say hello. Why not hold the door for them? This is how it begins. It only takes a few seconds and can expose the company to numerous risks. Hardware theft, identity theft, proprietary information theft—the list is infinite.

Safeguarding Your Company's Assets

A comprehensive data leakage prevention solution is imperative for any business, large or small. The protection of your sensitive and proprietary information and data is imperative to the success of your business. To ensure that you have considered the most complete approach, both the physical and digital nature of the risks of data leakage must be considered. These factors include how people and information are flowing into and out of all means of egress of the business.

The next question is usually centered on whether to create a homegrown solution or to purchase a vendor-supplied solution. To make this determination, consider the following questions:

- Cost/benefit analysis. Is the business able to implement a solution that encompasses all aspects of data and information protection using a homegrown solution? If not, a vendor must be considered.
- Manage internally or outsource. With the resource constraints of today's business environment, do you have the staff and manpower to adequately implement and manage a solution? If not, can the business afford to outsource? This should be considered a strategic initiative not to be easily dismissed.
- Comfort level. If outsourcing is the preferred option, is the business at ease with another entity managing such a critical and sensitive strategic initiative? It could save time, money, and resources if implemented successfully; however, uneasy minds often suffer duplication of efforts, which is counterproductive to the goal at hand.
- Comprehensiveness. What is the best means for devising the most comprehensive solution possible? Is it homegrown or off the shelf? Is it outsourced or managed internally? Compromise is out of the question, considering the alternatives.
- Checks and balances. Regardless of who is creating and managing the solution, a system of checks and balances is imperative. You must ensure not only that a thorough testing mechanism in place but also that backup is readily available to make certain that decisions made concerning policy and implementation are never accomplished in a vacuum.

Strength in Knowing

Evaluating your company's risk level may require an audit including dry-runs and blind tests. Although it is always a good idea to consider a third-party evaluation, many details (such as tracking USB devices) are difficult to determine and may unfortunately persist despite the highest level of corporate diligence. Putting solid effort into the creation of a foundation upon which to build will ensure assets are protected, which ultimately translates into security and peace of mind. ◆

Lori Cotton has written extensively on topics including networking, security, and business services, with a focus on the translation of technical features to business benefits. Lori has worked for numerous Fortune 100 and 500 companies such as Intel, BATM, and CA, as well as smaller companies such as Shiva and OpenReach.

Practical PowerShell

Uptime Availability

by Jeffery Hicks

You can download a zip file with all these scripts from http://www.realtime-windowsserver.com/code/v2n5_Practical_PowerShell.zip.

It's amazing the amount of useful information squirreled away in your network. Naturally, the challenging task is to retrieve it and make sense of it. Here's an example: when a computer starts up, an event is written to the system event log with an event code of 6005 that indicates the event log service has started. When a computer shuts down, another event is recorded indicating the service stopping with event code 6006. Knowing when those two events occurred, you could calculate how long the server was up and available, at least close enough to generate a reasonable result. Given that scenario, what about searching the entire system event log for these records and calculating the interval between the two events? You could create a report that shows when a computer was available and for how long. Of course, if you can do it for one computer, you should be able to do it for 10 or 100. This month, I have a PowerShell script called Get-UptimePerformance.ps1, which you can download here. Here's the complete script.

NOTE: I'm well aware that this approach isn't for everyone and there are most likely third-party, or even Microsoft, products that can accomplish this task easier. In addition, organizations with servers isolated behind firewalls will find this approach difficult to implement. But small to medium-sized organizations might find this solution helpful. I hope everyone finds the PowerShell techniques educational.

```
#Get-UptimePerformance.ps1
Param([string]$computername=$env:computername,
      [datetime]$since="2/17/2000")

#make computername upper case so it is nicer to look at.
$computername=$computername.ToUpper()

#convert to DMTF format
$start=[system.management.managementDateTimeConverter]::ToDMTFDateTime($since)

#only get entries that match the computername. This ignores entries where a computer
#was renamed. Also make it a wildcard search. Newer OSs use the FQDN.
$filter="Computername LIKE '$computername%' AND logfile='system' AND (EventCode=6006 OR
Eventcode=6005 OR EventCode=6008) AND TimeGenerated >='$start'"

#used for Write-Progress
$activity="Processing Event log on $computername"
```

```
Write-Progress -Activity $activity -status “Running Query” -currentoperation $filter

$logs=Get-WmiObject -Class win32_ntlogevent `
-filter  $filter -computername $computername | sort TimeGenerated

if ($logs.count -eq 0 -or !$logs) {
#bail if no records were found
     Write-Warning “$computername has no records since $since”
     return
}

#The first entry has to be event 6005
$x=0
while ($logs[0].EventCode -ne 6005) {
  $logs = $logs[$x..$logs.count]
  $x++
}

for ($i=0;$i -lt $logs.count-1;$i+=2) {
 [datetime]$started=$logs[$i].ConvertToDateTime($logs[$i].TimeGenerated)
 $percomplete=($i/$logs.count)*100
 Write-Progress -Activity $activity -status “Analyzing $($logs.count) entries” `
 -currentoperation “Calculating from $started” -percentcomplete $percomplete

  #if next record is 6005 then the computer likely did
  #not shut down properly
  if ($logs[$i+1].Eventcode -eq 6005) {
  #next event log should be 6008, if not then we should be able to skip
  #logging. It may be that the system stopped the event log service and restarted
  #it which may not leave a 6008 entry.
    if ($logs[$i+2].Eventcode -eq 6008) {
     [datetime]$stopped=”{0} {1}” -f $logs[$i+2].InsertionStrings[1],$logs[$i+2].
InsertionStrings[0]
      Write-Warning (“{0} {1}” -f $logs[$i+2].ConvertToDateTime($logs[$i+2].
TimeGenerated),$logs[$i+2].message)

    #get corresponding shutdown for first 6005 event
    [datetime]$stopped=$logs[$i+1].ConvertToDateTime($logs[$i+1].TimeGenerated)

    #bump $i to skip next entry
    $i+=2
    }
```

```
   }
  else {
   [datetime]$stopped=$logs[$i+1].ConvertToDateTime($logs[$i+1].TimeGenerated)
   }
   #write data to the pipeline in a custom object
     $obj=New-Object PSObject
     $obj | Add-Member -MemberType Noteproperty “Computer” -value $computername
     $obj | Add-Member -MemberType Noteproperty “Started” -value $started
     $obj | Add-Member -MemberType Noteproperty “Stopped” -value $stopped
     $obj | Add-Member -MemberType Noteproperty “Uptime” -value ($stopped-$started)
     write $obj

} #end FOR

#end script
```

The script uses Windows Management Instrumentation (WMI) via Get-WMIObject to search the event log on a remote computer for the necessary events and calculate uptime. The script takes a computer name as a parameter, defaulting to the local computer:

```
Param([string]$computername=$env:computername,
      [datetime]$since=”2/17/2000”)
```

The script also takes a parameter for a datetime value, $since. The script uses this value to limit the search query to find only records that have been recorded since this date. Otherwise, all records will be returned.

What happened on 2/17/2000? You may be wondering why I set this value as the default. To simplify things, I needed to include some default value in my WMI query, so I needed a date that would essentially return all events. Do you know this date? It is the official release date for Windows 2000. Since WMI wasn’t included with Microsoft operating systems (OSs) until Windows 2000, I didn’t see any reason to search for records older than this date. I also doubt you have many 9-year-old servers still running in your original configuration.

The datetime value is converted into a WMI-formatted date so that I can use it in my WMI filter:

```
$start=[system.management.managementDateTimeConverter]::ToDMTFDateTime($since)
```

With this information, I can build a filter query:

```
#only get entries that match the computername. This ignores entries where a computer
#was renamed. Also make it a wildcard search. Newer OSs use the FQDN.
$filter=”Computername LIKE ‘$computername%’ AND logfile=’system’ AND (EventCode=6006 OR
Eventcode=6005 OR EventCode=6008) AND TimeGenerated >=’$start’”
```

Armed with this, I can now query the remote computer:

```
$logs=Get-WmiObject -Class win32_ntlogevent `
-filter $filter -computername $computername | sort TimeGenerated
```

This version of the script doesn't support alternate credentials, but you can add that if you want. It shouldn't be necessary, but I'm also sorting the results by TimeGenerated to make sure they are processed in chronologic order.

If, for some reason, no records are returned, a warning message is displayed and the script ends:

```
if ($logs.count -eq 0 -or !$logs) {
#bail if no records were found
     Write-Warning "$computername has no records since $since"
     return
}
```

In order to get the right results, I have to make sure the first record is a 6005 event that indicates the EventLog service has started, so I loop through $logs until the first record has an EventCode property of 6005 and updates $logs:

```
$x=0
while ($logs[0].EventCode -ne 6005) {
  $logs = $logs[$x..$logs.count]
  $x++
}
```

On a perfect server, there will be an alternating pattern of 6005 and 6006 events. Using a For loop, I can get a start time value. The next record should be a shutdown event, which means I need to increment the counter by 2.

```
for ($i=0;$i -lt $logs.count-1;$i+=2) {
 [datetime]$started=$logs[$i].ConvertToDateTime($logs[$i].TimeGenerated)
…
[datetime]$stopped=$logs[$i+1].ConvertToDateTime($logs[$i+1].TimeGenerated)
```

Now comes the "tricky" part. In the event of a server crash or hard reboot, there will be no matching shutdown event. What will happen, though, is that event 6008 will be written when the computer restarts, indicating the previous shutdown was not planned. The challenging part is trying to juggle the event records to match everything up. I also found situations where the system restarted the EventLog service which further complicated matters. Here's the code that seems to work and account for all situations.

```
#if next record is 6005 then the computer likely did
  #not shut down properly
  if ($logs[$i+1].Eventcode -eq 6005) {
  #next event log should be 6008, if not then we should be able to skip
  #logging. It may be that the system stopped the event log service and restarted
  #it which may not leave a 6008 entry.
  if ($logs[$i+2].Eventcode -eq 6008) {
     [datetime]$stopped=”{0} {1}” -f $logs[$i+2].InsertionStrings[1],$logs[$i+2].
InsertionStrings[0]
      Write-Warning (“{0} {1}” -f $logs[$i+2].ConvertToDateTime($logs[$i+2].
TimeGenerated),$logs[$i+2].message)

    #get corresponding shutdown for first 6005 event
[datetime]$stopped=$logs[$i+1].ConvertToDateTime($logs[$i+1].TimeGenerated)

    #bump $i to skip next entry
    $i+=2
    }
```

If a 6008 event is detected instead of a matching 6006 event, a warning message is displayed that the computer did not shut down properly.

```
     Write-Warning (“{0} {1}” -f $logs[$i+2].ConvertToDateTime($logs[$i+2].
TimeGenerated),$logs[$i+2].message)
```

The purpose of all this PowerShell jujitsu is to get values for when the computer started, when it stopped, and how long it was up. The start and stop times are calculated from converting the WMI dates to a more user-friendly format.

```
[datetime]$stopped=$logs[$i+1].ConvertToDateTime($logs[$i+1].TimeGenerated)
```

Subtracting $stopped from $started returns a timespan object.

All of this information is written to the pipeline as a custom object:

```
#write data to the pipeline in a custom object
    $obj=New-Object PSObject
    $obj | Add-Member -MemberType Noteproperty “Computer” -value $computername
    $obj | Add-Member -MemberType Noteproperty “Started” -value $started
    $obj | Add-Member -MemberType Noteproperty “Stopped” -value $stopped
    $obj | Add-Member -MemberType Noteproperty “Uptime” -value ($stopped-$started)
    write $obj
```

Here’s sample output from this script:

```
Computer Started                Stopped                 Uptime
-------- -------                -------                 ------
CHAOS    9/14/2008 3:27:42 PM   9/14/2008 4:56:32 PM    01:28:50
CHAOS    9/14/2008 9:59:39 PM   9/15/2008 10:32:23 AM   12:32:44
CHAOS    9/15/2008 11:03:47 AM  9/15/2008 7:51:11 PM    08:47:24
CHAOS    9/15/2008 8:21:50 PM   9/16/2008 11:03:19 AM   14:41:29
```

After I worked with this for a while, I realized I could take this further. By adding the uptime values, I should be able to calculating server availability.

```
PS C:\> $uptimes=c:\scripts\posh\get-uptimeperformance.ps1 -computer “CHAOS”
PS C:\> [timespan]$total=0
PS C:\> for ($i=0;$i -lt $uptimes.count;$i++){$total+=$uptimes[$i].Uptime}
PS C:\> $total.toString()
149.06:02:38
```

Or I can calculate uptime %. All I need to do is figure out the time range from the output:

```
PS C:\> $span=$uptimes[-1].stopped - $uptimes[0].started
```

The last step is to divide the total time by the reported time span and format as a percentage:

```
PS C:\> “{0:P4}” -f ($total.ticks/$span.ticks)
 82.8989 %
```

I liked this so much that I wrote a wrapper script called Get-Availability.ps1, which is included in the download zip file. I'll let you explore the script on your own and leave you with a usage example:

```
PS C:\Scripts\> get-content y:\computers.txt | foreach {.\get-availability.ps1 -comp $_
-since 1/1/2009} | format-table -autosize
Computer       Availability StartDate             EndDate
--------       ------------ ---------             -------
xp01           86.6467 %    1/2/2009 10:58:50 AM  3/9/2009 11:20:50 AM
mycompany-dc01 63.5488 %    1/2/2009 5:07:52 PM   1/26/2009 12:05:25 PM
win2k801       81.6312 %    1/20/2009 11:51:24 AM 3/11/2009 3:00:19 AM
```

I expect that some of you might encounter issues running these scripts; I know had a number of issues developing them. Please join me in the PowerShell forum at ScriptingAnswers.com and we can work them out. ◆

Jeffery Hicks (MCSE, MCSA, MCT) is a Microsoft PowerShell MVP and Scripting Guru for SAPIEN Technologies. Jeff is a 17 year IT veteran specializing in administrative scripting and automation. Jeff is an active blogger, author, trainer and conference presenter. His latest book is Managing Active Directory with Windows PowerShell: TFM (SAPIEN Press). Follow Jeff at Twitter.com/JeffHicks and blog.sapien.com. You can contact Jeff at jhicks@sapien.com.

Exclusively Exchange

Unified Messaging Planning and Best Practices

by J. Peter Bruzzese

In the Unified Messaging (UM) world there are typically three groups of administrators. Those who have heard of UM but aren't quite sure what it is and what it does. Those who have heard of it and know how to install it and configure items such as dial plans and hunt groups along with the AutoAttendant and so forth. And finally, those that almost have the whole concept down but just need some good advice in planning and best practices. That is where we come in for this article.

Jack of All Trades: The UM Administrator

Being that most of us are Exchange specialists and not IP telephony experts, it may be difficult to get UM up and running in a real-world environment. It requires expertise at three levels:

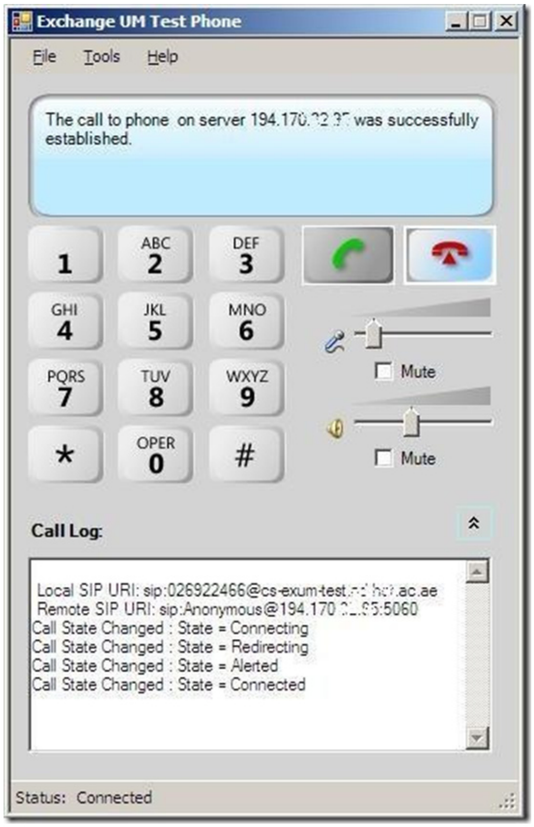
- Exchange 2007 and Active Directory (AD)
- Your specific VOIP gateway (configuration)
- Your specific PBX (configuration)

To assist, you might want to consult the Microsoft Telephony Advisor at http://technet.microsoft.com/en-us/library/cc164342.aspx. Keep in mind that you may not have the ability to have both telephony and Exchange expertise under your belt, but you need to pull in the telephony experience if that is where you are lacking.

Note: The February 2008 "Exclusively Exchange" column, discusses the purpose and configuration of UM.

Improving UM Through Simulation

You may need to get a little bit of practice in before you start deploying your UM servers. One way to do so with real equipment is to purchase a cheap VoIP solution to work with. However, before you go spending serious cash in this difficult economy, if you want to work with and test your UM configuration, try the Exchange UM Test Phone (see Figure 1), which will allow you to test the functionality of your UM Exchange Server.



The call to phone on server 194.170.[redacted] was successfully established.

```
 Local SIP URI: sip:026922466@cs-exum-test.[redacted].ac.ae
 Remote SIP URI: sip:Anonymous@194.170.[redacted]:5060
Call State Changed : State = Connecting
Call State Changed : State = Redirecting
Call State Changed : State = Alerted
Call State Changed : State = Connected
```

Status: Connected

Figure 1: The Exchange UM Test Phone.

For an interesting video on the Exchange UM Test Phone, check out Ilse Van Cirekinge on TechNet Chopsticks at http://www.microsoft.com/belux/TechNet/nl/chopsticks/default.aspx?id=868#.

Once the simulation is complete and you are feeling a bit more secure with UM, consider deploying a limited pilot and work through issues that will arise.

Watching and Improving Performance

In addition to testing the actual functionality of your UM server, you may have one that is up and running in your production environment already. What do you want to watch for? You want to avoid both latency and jitter when working with UM because now you have more than data packets coming through in an email; you have voice being streamed into voicemail boxes.

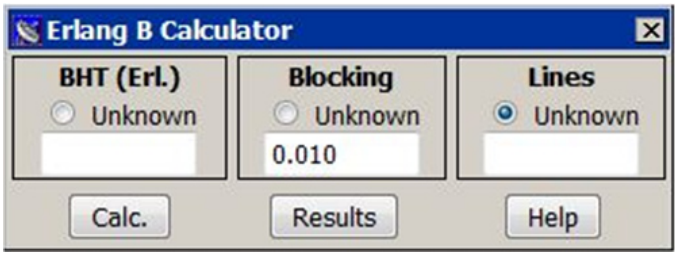
Latency involves the time between when something is sent and when it is received. This is different from jitter, which relates to the variation in the arrival of packets. When considering voice and video streams, you need to be more worried about jitter than latency.

So, how do you measure these items? One of the best network protocol analyzers (that is free) is called Wireshark. It has the ability not only to capture packets but also to read data with the ability to decrypt many protocols, decompress on the fly, and much more.

Note: There is a jitter buffer on the UM Server and the IP gateway that helps mitigate the amount of jitter.

Another part of your planning may involve knowing how many UM trunks you will need in relation to the number of UM calls that come in. Keeping in mind that a person who calls will leave only an average-length message (say 30 seconds to a minute). One of the ways you can calculate this is by using an Erlang B calculator.

The formula was derived by Agner Krarup Erlang and is used in planning telephone networks. Basically, you have three numbers: the Busy Hour Traffic (BHT), the Blocking, and the Lines. The BHT is the number of hours of call traffic there are during the busiest hour of operation. The Blocking is the failure of calls due to an insufficient number of lines being available (so it is the likelihood that a resource will not be available). The Lines is the number of lines in a trunk group. If you know two of the numbers, the third one can be worked out for you. You can work with the online calculator available at http://www.erlang.com/calculator/erlb/ (see Figure 2).



BHT (Erl.) ○ Unknown

Blocking ○ Unknown 0.010

Lines ● Unknown

Figure 2: An Erlang B calculator.

In addition, there are performance counters that you should watch on the UM server to ensure you have information about how you are performing. These include items such as Current Auto Attendant Calls, Current Play on Phone Calls, Current Subscriber Access Calls, Average Voice Message Size, and/or Average Greeting Size. These will provide you with the metrics you need to either make some changes or feel comfortable with the UM setup.

Taking UM to UC to UC&C

UM is just one piece of the puzzle in the new world of complicated communications and collaboration. Combining UM with Office Communications Server is what is being termed Unified Communications (UC), and if you add a Microsoft Office SharePoint Server (MOSS), you now have Unified Communications and Collaboration (UC&C). In future articles, I will begin introducing you to these other services and how they work with Exchange. Why settle for just UM when you can have UC, or better yet, UC&C? ◆

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. His most recent book "Exchange 2007 How-To" was published by Sams in January 2009. 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.

Copyright Statement