

## Re: SBS 2003 backup

**Source:**

<http://www.tech-archive.net/Archive/Windows/microsoft.public.windows.server.sbs/2004-07/2284.html>

---

**From:** Jeff Middleton [SBS-MVP] ([jeff\\_at\\_cfisolutions.com](mailto:jeff_at_cfisolutions.com))

**Date:** 07/09/04

Date: Fri, 9 Jul 2004 09:33:37 -0500

It's a little hard to explain a "wrong statement" like what has been made, and try to speculate about why they think their wrong answer is right. So I guess we could approach this like a FAQs list that is instead FWAs (Frequently Wrong Answers).

Q. Can I restore an entire SBS server with NTBackup alone?

FWA says no, thinking that if you don't do mailbox brick level backup, you can't restore the mailboxes. Wrong, backing up the Information Store will backup the mailboxes...they are a container within the Information Store, therefore they are backed up.

Another FWA is "you can't do a bare metal restore with it because you have to reinstall the OS". Well, duh, so does Veritas. The Bare Metal Restore methods known as Independent Disaster Recovery (IDR) or a similar name are based upon packaging the component drivers of Windows which are required to get the machine back to the point that a System State restore can be performed. While it's true that the IDR process attempt to make this look smoother and simpler than doing an install Windows from Setup, add the drivers and then restore from tape (or whatever), the fact is that as often as not, building the IDR media or meeting the requirements is both a hassle, and not completely reliable on a very complex server like an SBS. Besides, the IDR is typically yet another \$\$\$ option you have to buy, so at somepoint, if you don't really need it to do the restore, why use it to justify the difference between NTbackup vs Veritas or anyone else?

Another FWA is "I can't restore an SBS with NTbackup because of SFN errors". Wrong, the SFN issues are in the Windows API, so they are native to all products that do a file by file restore. You will have the same SFN errors on any product used for file by file based upon what the OS does to deal with that. In Windows 2000 and XP and all earlier products, SFN errors are created during a restore to bare metal which can (and do) cause breaks in an SBS Server because the restored registry will almost certainly refer to SFN paths for installed components using SFN registry entries, but that the file by file restore placed them in the folder space in a location that now has a different SFN. It happens with NTbackup, just like with Veritas, just like with Xcopy or Robocopy or whatever. Only a drive imaging product solves

that, and that's generally not what has been being discussed. SBS 2003 will restore from backup without the same risk of SFN errors because the Windows 2003 API now included the ability to attempt to restore any available SFN requested, provided it's not already in use by another file/folder. This solves Bare Metal Restore, but it doesn't provide a way to solve "restore over existing files to refresh the SFNs" if for some reason your SFN get out of sync by relocating a critical folder tree with a move, then moving it back.

With that all said, I personally don't prefer to use NTbackup, I prefer not to. But it has nothing to do with whether or not it can restore a server. In fact, I frequently use it in utilitarian work to do such stuff, but historically I've not deployed customers on it. Why? Because most of my customers are running multi-server, and NTbackup doesn't provide a convenient multi-server management process.

In addition, I have preferred the Veritas management auditing and logging, and the easy way to watch the behavior of the backup process at a glance. The SBS Dev team has added many similar features to SBS 2003 now, and it makes the NTbackup feature much more functional. Still, without multi-server, I don't want to run 2-3 different backup processes each night, not to mention that only the SBS is going to give me a decent management console and daily logging report.

Q. NTbackup doesn't backup Open Files, does it?

FWA identifies that certain files are skipped when you perform a backup, so this proves that SBS doesn't backup open file, right? Wrong

Well, actually yes NTbackup does backup open files, and no that doesn't mean that it won't skip files. You might look at this kb:

Files That Are Automatically Skipped by the Backup Program (NTBackup.exe) During the Backup and Restore Processes

<http://support.microsoft.com/default.aspx?scid=kb:en-us:104169&Product=winsvr2003>

It's not that it's impossible to miss a file that you want in a backup with NTbackup, rather it is just a likely to happen with any basic backup program including Veritas. That's been the historical truth, that's still fact now. Open File backup is a technology that is offered in some configurations to attempt to work around that issue when you have the need for snapshot backups of live data that you can't shutdown and you can't use an application aware method to captures. Exchange and SQL are both supported by "application aware" backup support with NTbackup. If you go to Veritas, you need to by the Agent for Exchange and Agent for SQL to get that, not just Veritas Backup Exec.

NTbackup in SBS 2003 (read: Windows 2003) supports Volume Shadow Copy techniques that emulate the concepts of Open File backup, but not in an identical method. Volume Shadow Copy is more of a snapshot before you run the backup method of capture rather than a snapshot file by file as the

backup runs or snapshot as the backup process starts. Regardless, Volume Shadow Copy isn't really essential to get a backup of your SBS core files, it's yet another method of capturing any sort of files that you don't already have another method to capture. One way of looking at Volume Shadow Copy is that you could use it to restore a system to a point in time even if you didn't make a tape backup...the backup of the snapshot is already on the drive in a different location. Therefore, if the snapshot is in VSC, and you run a backup to tape that captures the VSC contents...doesn't that imply that you have a snapshot of the system, including the files that were open?

I'm not suggesting that VSC, Open File backup and backup in general are not complicated concepts, rather I'm just suggesting that too many people bite on a marketing line from someone like Veritas that implies that "buy our product and you have no problems" when in fact, you still have the exact same problems, you just have a different vendor for support now. On that basis, you probably could make the argument that you can PAY for better support from Veritas for a full server recovery condition than you can with Microsoft for the simple reason that Veritas as a product line intends that they will support recovery and restore of all the 3rd party conditions that you bought agents to recover and manage with, but MS doesn't. MS limits their concerns to the MS product. Is there some room for debate? Sure, but I don't think it's a fair statement to say that "the only way you can recover an SBS server is with something other than SBS native backup" because this just bull.

If you are really serious about doing backups, you will find that you have exceptions and management issues that you need to address in tuning any backup process, and it doesn't mean you have to go to a different product to get the results. But as the simplest statement possible on this topic, NTbackup will recover an SBS server and all the components that ship with it and restore it completely to functional condition, and with reasonable ease of use. I would further add that you are just as likely to have problems restoring any 3rd party application or even client accessed databases stored on the server in filespace whether you use NTbackup or 3rd party backup, the issues are the same. If you choose to use Open File backup on a 3rd party product, you have to be aware that you are just as likely to capture files you can use as you are to capture useful one. What I mean is that many applications that are not providing agents to backup the program while open are just as likely to not work with a captured copy of the "open file", you will have a corrupted database structure. For such programs, you need to know what they are, and how you will restore them, and the chances are that you should be doing an independent data recovery process within that program. For instance, Accounting applications may provide an option for "would you like to make a backup of you data now" option that basically writes a complete "cold" copy of the data to a backup file, much like Quickbooks does. Therefore, if you have to restore, you are restoring that backup, and using it...not the open files that were left open overnight when the backup job ran on the server with no ability to access the open database files without an agent. The Open File manager would restore the live database with flags set to appear that a user is still in the program, and some transaction logs on more sophisticated accounting programs would be

in a condition that can't be restarted that way without recovery steps.

Of course, an open file recovery isn't useful for applications running at the workstations, stored on the server, but living live in the memory of the workstations. So if the debate is going to be about recovery of the entire system, lets talk about the entire package. If you don't have a process in you network to capture cold files at some point, or use an agent aware of the application itself, you won't have a better chance of recovery with one product vs another.

Q. What about Exchange on the System Drive issue?

I don't even have a FWA for this. Who suggested this and why, I don't know.

Exchange can and will run just fine on a single partition with the OS. Now, if the idea is that you should split the logs from the database files, yes I know that is frequently discussed, that there's also that part of the discussion that if you are running on a RAID5, it's not as much a concern. In a small business, nobody should suggest that you have to install a second RAID5 for the log files or even a separate mirror for them because of some mistaken idea about this being a best practice. Do the people who talk about this even know WHY it's suggested to be on a different partition?

The reason it's recommended is so that if you run out of disk space or if you crash the drive, you have the logs on a separate partition so that you can take the last backup, then replay the log files in a restore process, even if you have lost the previous EDB/STM files that are from the same time period. This is fine if you are planning on coming out of the debris of a crash with only half the information, but wouldn't the better plan be to come out with all the information...logs and databases?

The fact is that most people who operate Exchange on an SBS should expect that running on a RAID5 is sufficient protection of the transaction logs, and doesn't imply they have to be on a different logical partition if they are all on the same logical volume of the RAID. If the RAID dies, they all die, it doesn't matter what logical partition they are on. Anyone suggesting that they must have different drives for the log files than for the database files in an SBS strictly for the Exchange is just reading too many Enterprise MCSE books that they don't really understand. The logic is both out of scale, and not even high priority in the context of a business running everything on a single server.