

## The Old New Thing

### When you compress a drive, some files are exempted, but you can force it, and then it's your problem

27 Nov 2007 10:00 AM

29

On the drive property sheet for an NTFS volume, there is a checkbox called "Compress drive to save disk space." If you check that box, the shell marks the drive as "compress all newly-created files" and also goes through and compresses all the existing files on the drive.

Well, almost all the file.

Some files are exempted by default.

Examples of exempted files are the files involved in booting the system (NTLDR, NTDETECT.COM, HIBERFIL.SYS) and files for which write requests must succeed (PAGEFILE.SYS). (If a file is compressed, then a write to previously-committed file data may fail if the new data does not compress as well as the old data and there is no more disk space.) These files are exempted on all drives, even if they're not your system drive.

On the other hand, if you right-click one of these exempted files and explicitly compress it, then the shell will compress it (or at least try to). For boot files, this will typically succeed since boot files are used only at boot; once the system is running, they aren't needed any more and therefore there aren't any open handles to the file with restrictive sharing modes.

Of course, if you do this to your system drive, it won't boot any more. So don't do that.

Like with many things in the physical world, if you force it too hard, it may break.

#### Blog - Comment List MSDN TechNet

#### Comments



**Cody**

27 Nov 2007 10:51 AM

#

I once put my swap file on a compressed logical drive in Win 3.1. The computer didn't like booting after that.



**Nawak**

27 Nov 2007 11:52 AM

#

Interesting, I thought it was because they were in use... (A lot more than 3 files on my disk aren't compressed, so it seems at least to play a role)

Does the same happen if the drive is marked as 'compressed drive' before the install occurs? I remember that you can install windows on a disk without reformatting it.

I'm quite sure the same happens because else the system would be fragile as you described, but then I wonder what flag is passed to CreateFile when creating those files on the disk... FILE\_ATTRIBUTE\_SYSTEM?



**anonymous**

27 Nov 2007 1:33 PM

#

I wished such safety checks would apply to EFS as well. I had the %TEMP% folder of the administrator EFS-encrypted, and this ended up into some system files being encrypted when installing a regular update.



**Gabe**

27 Nov 2007 1:59 PM

#

I would expect that compressing the HIBERFIL wouldn't cause any problems because the data in there should already be compressed. In fact, it probably uses the NTFS compression routines.



**Michael Teper**

27 Nov 2007 3:08 PM

#

Compressing key system files is probably the kind of problem that would never reach support or be easy for support to identify even if it did, but is one of those things contributing to the poor image associated with Windows.

Would it be a bad idea to protect key system files from encryption similar to the way key windows files are now protected from being overwritten and offer the user a meaningful message when s/he tries to force it?



**Dan McCarty**

27 Nov 2007 3:13 PM

#

"These files are exempted on all drives"

Does that mean the compression scheme won't compress anything called

ntldr

ntdetect.com

hiberfil.sys

pagefile.sys

No matter where they are in the system and whether or not they're the official version of the file?

*[When you compress a drive, some files are exempted, but you can force it, and then it's your problem. -Raymond]*



**Chris**

27 Nov 2007 3:46 PM

#

I'd imagine (but don't know for sure) that the automatic exemptions would only exempt files named ntldr, ntldetector.com, hiberfil.sys, and pagefile.sys only in the root directory of a drive (as those are the only places those files legitimately exist).

I once had to work on a computer where someone had done just the subject of this article. The laptop booted up saying simply "NTLDR is compressed."

I suppose not enough of the filesystem driver is loaded to fix such a simple error.



**foxyshadis**

27 Nov 2007 5:41 PM

#

That's interesting, no exemption for ntfs.sys? I always thought that was one you couldn't compress, for the obvious reason, but I suppose I'll try it with BartPE handy one of these days.



**Kemp**

27 Nov 2007 5:42 PM

#

"Compressing key system files is probably the kind of problem that would never reach support or be easy for support to identify even if it did, but is one of those things contributing to the poor image associated with Windows.

Would it be a bad idea to protect key system files from encryption similar to the way key windows files are now protected from being overwritten and offer the user a meaningful message when s/he tries to force it?"

I imagine the reasoning is thus:

- 1) Average Joe will select to compress the drive and the critical files will be safe (he shouldn't even see them unless he has screwed with his settings)
- 2) Anyone else should know what they're doing before they mess with hidden system files with cryptic names

Basically, once you're at the level where you know how to view and force compression on those files you really should also be at the level where you're able to stop and think for a second. Considering why they're marked hidden and system is a clue not to mess with them.



**Dan**

27 Nov 2007 5:54 PM

#

Cody: Except Windows 3.1 wasn't really an operating system (it ran on top of DOS) and so the swap file wasn't required for system boot? I think you mean to say when Windows started up.

Nawak: From the blog entry it seems to be based on names and not on any attribute flags.

anonymous: I wouldn't try to encrypt Temp... I would just delete every file in it. If a program needed a file in it, it shouldn't have put it in Temp! I'd find a better program if I could. At the very least the safest you can do is to just delete files in Temp with a Last Modified stamp before the last boot up.

Gabe: HIBERFIL.SYS is the dump of windows memory to disk... I would guess it can't be compressed.

Although it seems like it would be easy, you have to remember that the CODE which is DUMPING memory to disk WILL ALSO BE DUMPED... if you dumped and then shutdown, when you started up whatever code was un-dumping would be overwritten, probably causing a BSoD. Next, even if that wasn't a problem, if we resumed where we left off... well when we left off we were dumping memory to disk followed by a shutdown! Oops. So it's not so simple.

But here's what I would do to work around this (keep in mind all of this is guesswork, and I know nothing about the actual process): 1) Have a specific fixed virtual memory area allocated for the memory dump and the memory restore (whichever one is happening at the moment). Don't allow the use of this block for ANYTHING else. Ever. 2) Keep this area as small as possible so minimize the impact it will have on possible memory usage and so forth. 3) I'd guess we'd also have to make it self-reliant (ie built in mini-filesystem support, no NTFS compression routines) to ensure we don't make any calls OUTSIDE the block. 4) Don't dump or restore this area. (Anyone know if I'm close?)

Chris: A bootloader has a limited size. If it needs more, you offload part of it to a hard drive partition and write a filesystem driver into the bootloader... but again... limited size, so you don't want to put in stuff like compression support.



**Mark Steward**

27 Nov 2007 7:35 PM

#

Dan: hiberfil.sys is indeed compressed, I believe in blocks of 64K at a time. Whether the dumping code is written to disk is irrelevant, as it's NTLDR that reads it back in (and presumably then passes control back to the scheduler).



-

27 Nov 2007 9:07 PM

#

I might be wrong, but I don't think PAGEFILE.SYS is just a file "for which write requests must succeed". I think it gets special treatment, as in the extent list is always stored in memory, and it doesn't use the normal FS facilities, rather operating at a lower level. Things like the free space bitmap might not be in RAM, and I think a great deal of the NTFS (and FASTFAT for that matter) code can be swapped out too, including compression/decompression.

In any case, no matter how hard you try, you won't be able to use a compressed pagefile. You can't compress it while it's in use, and even if you compress it from other installation, Windows will clear the compressed flag and reallocate it on mount.

So that one isn't a risk. The boot ones are, though.



**bd\_**  
27 Nov 2007 10:22 PM  
#

Dan, it's theoretically possible to compress a hibernation file if you have sufficient ram - essentially, you'll suspend the normal kernel and drop to a tiny copy kernel, copy all kernel memory into free space, then return to the real kernel to save user pages and the kernel memory image. On boot you'll do much the same thing in reverse. You generally won't have a problem with resuming into the suspend code, since you can just overwrite the state of the currently running thread to resume in the 'get system running again after resume' code.

One complication is that since you're going to be modifying filesystem metadata you have to be very very careful not to get confused when the system starts up again and various structures on-disk haven't changed from what they were before you took the kernel image.

A much simpler approach - and probably what windows does - is to do essentially the same thing, but preallocate the hibernation file before imaging the kernel, then just push the data out to raw disk blocks. This way filesystem metadata remains unchanged and you don't have to worry about things getting out of sync. Although it's not strictly necessary to keep the hibernation file allocated all the time, it's useful, as that way you can ensure hibernation will succeed, modulo available swap.



**Anon**  
28 Nov 2007 12:18 AM  
#

NTLDR must have been an interesting piece of code to write. As far I can tell, there is a real mode stub that works like a Bios extender and a protected mode section that contains enough code to do read only access to NTFS and FAT. It can use the Bios for disk access via the bios extender or you can even use a SCSI miniport driver. And registry access too. All in 209K.

It's a shame that this stuff isn't publically documented really.



**Johannes Rössel**

28 Nov 2007 1:47 AM

#

Hmm, I actually did compress my ntldr a long time ago ... took me a while to figure out that in the recovery console compression is just an attribute to set or unset with attrib ...

**PatriotB**

28 Nov 2007 3:39 AM

#

@Anon -- Actually, it's a good thing NTLDR and company weren't publicly documented... it would probably have resulted in tying the hands of the boot team when it comes to trying to enhance/improve it in the future. You'll note that Vista doesn't use NTLDR at all, but uses a brand new boot loader.

@Gabe and others regarding hiberfil.sys -- I would hope it isn't compressed; hibernate and resume times are slow enough as it is, and adding compression to the mix would make it even slower. And hard drive space is cheap anyways.



**Arun Philip**

28 Nov 2007 4:29 AM

#

@PatriotB:

I would hazard a guess that hibernate/resume are more constrained by spindle speed rather than CPU.



**Nawak**

28 Nov 2007 7:31 AM

#

@Dan:

"Nawak: From the blog entry it seems to be based on names and not on any attribute flags."

Well, from the blog entry it seems that the compression of existing files, and the protection of the critical files, is done by the shell and not the filesystem (else how would you override the protection).

Then the drive is marked "compress all new files", an attribute that has to be enforced by the filesystem, not the shell, since the shell is not "between" a program and the filesystem when the program creates "new files".

So my question was, do these safeguards work (and how) when the drive is already marked as "compressed" before a windows installation, since then the new files that are created (and that should be compressed) are the 'protected' files in question.

Maybe the windows setup refuses to install on a compressed partition, that would be the

most simple solution, even if then it could let some users unable to reinstall without formatting and losing data. (It would force them to backup first, something that could be problematic when the system doesn't boot correctly)

I don't want to bother anyone with this question though, it was out of curiosity. If the answer is known, then I would be glad to hear it, else it doesn't matter, I may try sometimes in the future when I have to install a new PC.



**Neil**

28 Nov 2007 7:48 AM

#

Nitpicker's corner: Cody is referring to a permanent swap file. Temporary swap files can easily be created on compressed or remote media. (Particularly useful for diskless workstations.)



**Borlock**

28 Nov 2007 10:25 AM

#

Dan,

I'm pretty sure 'anonymous' (with regard to the encrypted %temp% directory issue), referred to the installation of a program which, during installation, temporarily extracts some files into temp and then copies it to system32.

Lots of installs do this, and it's perfectly valid.

The issue with that though is that since %temp% is encrypted, the file that was extracted in there retains its encryption when it subsequently gets copied somewhere else, causing the file inside system32 to be encrypted as well.

This doesn't mean the app is trying to run itself from temp.



**brian**

28 Nov 2007 3:45 PM

#

LOL compress your pagefile.... i stopped reading comments after that. Thats funny. Awesome post.



**mv adu**

28 Nov 2007 3:47 PM

#

I don't think hibernate file is compressed. Otherwise it should take less space than the installed RAM, but Windows takes exactly same out of Harddisk space as that of installed RAM. So its just a raw dump and read back again.



**mv adu**

28 Nov 2007 3:47 PM

#

I don't think hibernate file is compressed. Otherwise it should take less space than the installed RAM, but Windows takes exactly same out of Hard disk space as that of installed RAM. So its just a raw dump and read back again.



**ERock**

28 Nov 2007 3:50 PM

#

I would LOVE a way to specify exceptions manually for other files. Like, Steam games run DOG slow if the content files are compressed but if they reach a certain age the "Compress Old Files" option in Disk Cleanup will compress them.



**GreaseMonkey**

28 Nov 2007 5:22 PM

#

Compressing your swap file shouldn't really matter speedwise if you've got a fast CPU and a slow hard disk.

But yeah, I don't think I'll be compressing NTLDR any time soon.



**DriverDude**

28 Nov 2007 7:00 PM

#

"I don't think hibernate file is compressed. Otherwise it should take less space than the installed RAM, but Windows takes exactly same out of Hard disk space as that of installed RAM."

Compression does not \*gaurantee\* that data will take less space. Pre-allocating as much space as RAM gaurantees all the RAM can be dumped in as few fragments as possible when it comes time to hibernate.

Windows 2000 dumps all RAM whereas XP excludes the cache, and I think Vista excludes anything that can be read from elsewhere (i.e., code segments can be re-read from the .exe) Vista also has readyboost to help reload code.

I also suspect that NTLDR uses real-mode BIOS calls to read the hibernate file, which might explain why resume is "slow" relative to native Win32 disk performance. Vista might have improved this with its new boot loaders.





**DriverDude**

28 Nov 2007 7:09 PM

#

Encrypting %TEMP% ensures that unencrypted temp data is never written to disk. You might be editing an encrypted Word file; the file will remain encrypted before and afterwards but MS Word might write some fragment of the doc into an unencrypted temp file.

I once encrypted my Temp Internet Files folder, until I realized everything IE downloads is encrypted (even when I don't want them to be), because IE downloads them into its Temp folder and then moves it to the final destination.



**Cody**

29 Nov 2007 10:16 AM

#

For those that care, I was 10. I may have forgotten some details since then.



**Softpedia**

29 Nov 2007 3:39 PM

#

PingBack from <http://news.softpedia.com/news/What-Not-to-Compress-in-Windows-72311.shtml>