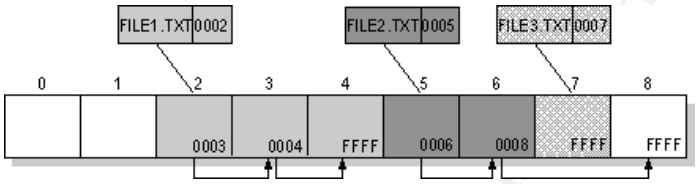
Global Information Assurance Certification Paper

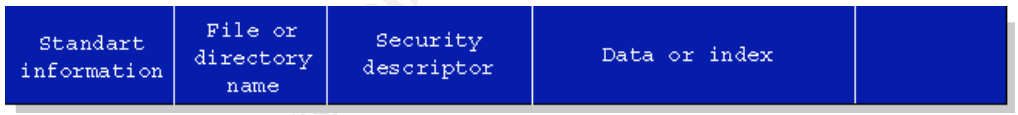
SANS/GIAC Practical Assignment
For GSEC Certification

FAT 32



VS

NTFS IN A FORENSIC ENVIRONMENT



Presented by
JOSÉE FLEURY

July 2003

TABLE OF CONTENTS

Introduction

The purpose of this paper is to shed light on the advantages of using different types of filesystems in the partition process of wiped drives in a forensic environment. The security aspect of each partition system, i.e. FAT32 or NTFS will be examined and applied in the work of forensic investigators. The uses of both systems will be explained. The choice of one system over the other and the reasons behind that choice will be presented.

The FAT 32 system for a forensic investigator is a preferable choice considering that any NTFS system can read FAT32 but the FAT 32 system, in it's native form, cannot read NTFS. Major problems occur when the imaging of a system does not support NTFS. For example, the imaging of a server with fiber channel hard drives would require a different procedure since the suspect's system would not recognize the destination external hard drive (where the image will be) because of the type of filesystem used on that hard disk. In a forensic environment, often time is of the essence and we, the investigators, have to make sure that our work is effective and representative of the object to be imaged.

The qualities of FAT 32 are more practical in a forensic situation than those of the NTFS system, especially when imaging hard drives. I understand that for a computer user the NTFS represents a better choice.

To help demonstrate the effect of different filesystems in a forensic environment, I will later describe a real life experience encountered by investigators while imaging a server.

In the forensic environment the imaging of suspect drive starts with the use of a sterilized hard drive that is used to store the image. To make sure that the hard drive is sterilized, it is wiped three times according to Department of National Defense standard 5220.22-M. To be able to utilize the hard drive on site and save some time the drives are partitioned and formatted in a lab environment.

Before the introduction of NTFS all of our hard drives were partitioned with FAT 32 and there was no problem making images at any time. Since the advent of NTFS we have encountered problems with partitioning the hard drives with the latter system. As mentioned previously, FAT 32 cannot recognize NTFS tables and that causes a problem when imaging a FAT 32 system with an external hard drive that was partitioned with NTFS. On site the investigators must repartition the hard drives before making the images (often not a good option since time on site is limited) or use a hard drive that will be seen by the system. The security features of NTFS therefore, are not relevant when it comes to imaging a suspect hard drive.

Before going any further it is essential to present the characteristics of FAT systems and NTFS.

FAT

FAT is by far the most simplistic of the file systems supported by Windows NT. The FAT file system is characterized by the file allocation table (FAT), which is really a table that resides at the very "top" of the volume. To protect the volume, two copies of the FAT are kept in case one is damaged. In addition, the FAT tables and the root directory must be stored in a fixed location so that the system's boot files can be correctly located.

A disk formatted with FAT is allocated in clusters, the sizes of which are determined by the size of the volume. When a file is created, an entry is created in the directory and the first cluster number containing data is established. This entry in the FAT table either indicates that this is the last cluster of the file, or points to the next cluster.

There is no organization to the FAT directory structure, and files are given the first open location on the drive. In addition, FAT supports read-only, hidden, system, and archive file attributes.

Advantages of FAT

The FAT file system is best for drives and/or partitions under approximately 200 MB because, FAT starts out with very little overhead.

Disadvantages of FAT

It is preferable, not to use the FAT file system when using drives or partitions over 200 MB. This is because as the size of the volume increases performance with FAT will quickly decrease. It is not possible to set permissions on files that are FAT partitions in their native form.

FAT partitions are limited in size to a maximum of 4 Gigabytes (GB) under Windows NT and 2 GB in MS-DOS.

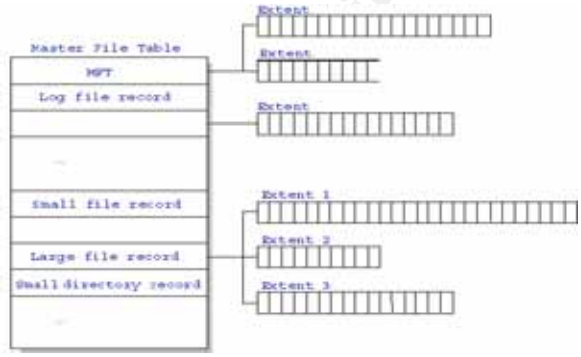
FAT 32

The successor to FAT 16, aptly-named FAT 32, improved on partition size limitations inherent in the system, and was supported in Windows 95 and is still the default file system for Windows 9x operating systems. Additionally, the filesystem is supported in the Windows NT, 2000, XP operating systems and Linux distributions. These features make it a viable filesystem to use when a

partition might need to be read and written by a number of operating systems present on the computer. One of the larger problems facing FAT partitions is that FAT 16, an outdated filesystem, only supports volume sizes of 2 gigabytes, far less than what is commonly available in current systems. The FAT 32 system increases this limit to 2 terabytes, which exceeds current hard disk sizes, but it is possible that the limitation will again be an issue when drives beyond this size become available.

Another potential issue with FAT 32 is that the cluster size is variable and can become very large. As the partition increases, two situations are possible for arranging the disk clusters. Smaller clusters obviously lead to less internal fragmentation as more clusters are necessary to compose a file. This results in higher precision, but the benefit comes at the price of there being more clusters for the filesystem to manage. This leads to an increased overhead and more seek time as more clusters have to be accessed, possibly not at concurrent places on the physical disk. The option of defragmenting the disk would possibly improve performance. On the other hand, applying the larger cluster size makes the overhead more bearable but results in a much greater risk for internal fragmentation. Any system that has a large number of files ranging from 1 to 4 KB in size, for example, would do very poorly on a drive larger than 60 GB in size, as it would result in severe internal fragmentation for any of these smaller files. The efficiency of the filesystem drops as the size of the partition increases. The reason for this increasing cluster size is that the maximum number of clusters possible on a FAT 32 filesystem partition is 268 435 445 clusters. Because of this, as the size of the hard disk partition increases, in order for the FAT 32 cluster limit to be maintained, the size of each individual cluster must be increased.

NTFS



The other predominant filesystem used by Windows operating systems is the aptly-named NT filesystem (NTFS). Originally designed for the Windows NT operating system, the filesystem is also supported by Windows 2000 and XP. Some of the easily noticeable gains brought by using NTFS are that while it has the same 2 TB limit on volume sizes just as in FAT 32, there is no practical limit to the number of clusters that can be used by the

filesystem. This would eliminate the massive ballooning of cluster size that is found on large FAT 32 partitions.

Additionally, NTFS has other inherent features not found in FAT 32 such as:

- **File security**
  Access rights can be assigned to files and directories, allowing users full access, partial access or no access at all to data on the hard disk,
- **Encryption**
  NTFS can automatically encrypt and decrypt file data as it is read and written to the disk,
- **Disk compression**
  File and directory compression can be performed without using any third party software, which saves space, while still allowing for transparent access and operation to the user,
- **Support for large hard disks**
  We are talking very large. Try a theoretical limit of 16 Exabytes, and up to 2 Terabytes,
- **File names**
  Native support of long file names and a 16-bit character standard called Unicode (likely the next generation ASCII),
- **Storage quotas**
  Disk quotas can be assigned that limit the amount of disk space users can access on a partition,
- **Sparse files**
  Let the user assign and reserve hard disk space to specific files,
- **File streams**
  Support for multiple data streams,
- **Fault tolerance**
  An enhanced ability to seamlessly respond to unexpected hardware and software errors.

Due to the nature of the filesystem, NTFS performs poorly on small disk volumes but much better on larger volumes than FAT 32. This makes it a better choice for the futures for two reasons; security is more and more of an issue and hard disk size continues to increase.

From a user's point of view, NTFS continues to organize files into directories. However, unlike FAT, there are no "special" objects on the disk and there is no dependence on the underlying hardware, such as 512 byte sectors. In addition, there are no special locations on the disk, such as FAT tables or HPFS Super Blocks.

The goals of NTFS are to provide:

- Reliability, which is especially desirable for high end systems and file servers,
- A platform for added functionality,
- Removal of the limitations of the FAT file system.

Reliability

To ensure reliability of NTFS three major areas were addressed: recoverability, removal of fatal single sector failures, and hot fixing.

NTFS is a recoverable file system because it keeps track of transactions against the file system. When a CHKDSK is performed on FAT or HPFS, the consistency of pointers within the directory, allocation, and file tables is being checked. Under NTFS, a log of transactions against these components is maintained so that CHKDSK need only roll back transactions to the last commit point in order to recover consistency within the file system.

Under FAT or HPFS, if a sector that is the location of one of the file system's special objects fails, then a single sector failure will occur. NTFS avoids this in two ways: first, by not using special objects on the disk and tracking and protecting all objects that are on the disk. Second, under NTFS, multiple copies (the number depends on the volume size) of the Master File Table are kept.

Added Functionality

One of the major design goals of Windows NT at every level is to provide a platform that can be added to and built upon and NTFS is no exception. NTFS provides a rich and flexible platform for other file systems to use. In addition, NTFS fully supports the Windows NT security model and supports multiple data streams. No longer is a data file a single stream of data. Finally, under NTFS, a user can add his or her own user-defined attributes to a file.

Advantages of NTFS

NTFS is best for use on volumes of about 400 MB or more. This is because performance does not degrade under NTFS, as it does under FAT, with larger volume sizes.

The recoverability designed into NTFS is such that a user should never have to run any sort of disk repair utility on an NTFS partition.

One of the key qualities of the NTFS filesystem is that inherent in the filesystem is transaction logging and recovery behavior, which makes the filesystem recoverable and almost guarantees the volume's consistency. While generally not important, in the event of a disk failure the NTFS filesystem restores consistency on the disk by executing recovery procedures, accessing information stored in the log file. This recovery procedure is done the first time a program attempts to access a NTFS volume after a system reboot following a failure. It guarantees that the volume structure is not corrupted by mismatching entries in the master file table.

Since NTFS has no realistic limit on the number of clusters it can manage, it does not suffer from the same constraints on cluster size as FAT 32. Cluster size is variable allowing the user to set up his or her NTFS drive to determine what cluster size is best for the system: small clusters for smaller files, and vice versa for larger multimedia files. This user-determined cluster size, when used properly, results in much less internal fragmentation as the cluster size is better suited for the files on the volume.

If one wants to make the computer a multi-boot system, one may or may not want to consider FAT32.

Disadvantages of NTFS

It is not recommended to use NTFS on a volume that is smaller than 400 MB, due to the amount of space overhead involved in NTFS. This space overhead is in the form of NTFS system files that typically use at least 4 MB of drive space on a 100 MB partition.

It is not possible to format a floppy disk with the NTFS file system; Windows NT formats all floppy disks with the FAT 12 file system because the overhead involved in NTFS will not fit onto a floppy disk.

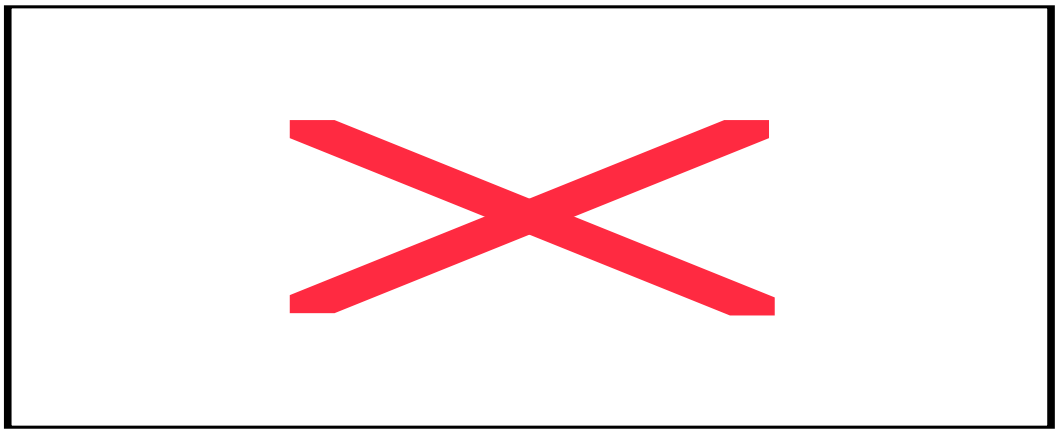
Forensic experience

Now that we are aware of all the characteristics of both file allocation systems, I will present our case and the reasoning behind our forensic choices.

In my capacity as a forensic computer investigator, one of my functions is to make perfect images of suspect hard drives with no data pollution. We have many ways of performing this task but the preferred method is to use the ENCASE (www.guidancesoftware.com) tools. ENCASE can easily be used in a Windows or DOS environment. ENCASE can be used for single hard disks or servers. Usually, the drives we have to image are as large as the largest hard disk on the market. To access the suspect's drive we plug in the FASTBLOC

through our laptop (PCMCIA) and send the raw data to an external hard disk by using the ENCASE program in a Windows 2000 environment on our forensic laptop. While using this procedure, the evidence drive (the one containing the image) can be formatted in either FAT 32 or NTFS, since Windows 2000 recognize both. That process is mainly how we proceed with our forensic imaging of IDE drives.

With the level of technology available to us today, IDE drives are easier to image. The following is an example of a setup scenario.



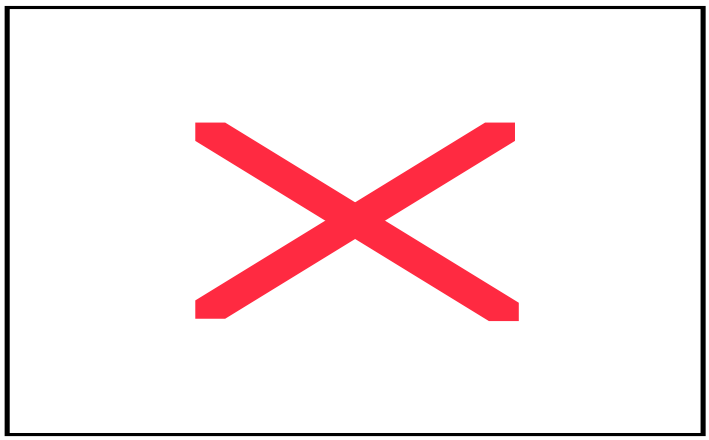
As for SCSI hard drives we have to use a slightly different way to obtain the image because FASTBLOC does not support them.

In the diagram above we use FASTBLOC to power the SCSI hard disk so the investigator can image the hard drive. Once the SCSI hard disk is "write protected" we use the ENCASE program in a Windows 2000 environment on our forensic laptop. The image is sent to an external hard drive which is formatted in either FAT 32 or NTFS.

I will now present a situation where we encountered problems with evidence drives that were formatted using NTFS.

Our investigators went to a business site where images of two servers were required. The servers totaled 10 Terabytes of information on a fiber

channel, storage area network. Unfortunately, FASTBLOC cannot be used with fiber channel hard drives, that is when we have to resort to ENCASE in a DOS environment. Using our laptop with FASTBLOC was not an option since all the hard drives were fiber channel and we had no means to power them externally. The only option available to us was to boot directly with the suspect's machine using a "sanitized boot disk" (Diskette containing: boot process, fdisk, ENCASE for native DOS, etc.). A promise card Ultra 133 TX2 was installed in the server and an IDE hard disk was connected to it, furthermore, all the fiber channel drives were pinned as read-only. By starting the suspect's machine with our boot diskette the investigators would be able to image the hard disks and send the image to the added IDE drive. (See the diagram below)



The targeted external drive (one receiving the image) had been formatted with NTFS and the server could not recognize it because we used a native DOS bootdisk which does not support NTFS.

The investigators were trying to figure out the reasons for the problem and after many attempts and brainstorming they decided to try another external hard disk that had been formatted and partitioned in FAT 32. As it turned out the server recognized the media and the image was captured. We discovered that the formatting of our wiped drive was very relevant and influential on our forensic work. Before this experience, we had never encountered such a problem, mainly because we avoid using the suspect's machine to image the hard drives. The reason behind our failed attempts at imaging using a hard disk that has been formatted in NTFS is that native DOS does not recognize NTFS.

Conclusion

In order to facilitate our work in the future, we would need to either use FAT 32 hard disks or switch our Native DOS program to NTFS DOS.

Regardless of all the characteristics that NTFS has to offer, in our line of duty it would be better to format and partition an evidence hard disk in FAT 32 since it is the filesystem that is most commonly supported. This is not to denigrate the qualities of the NTFS partition table but after experimenting on site with such a problem and based on our specific needs, it became obvious that the practice of partitioning and depending only on NTFS was not a good approach. In the case presented above, the end result was not affected by the problem encountered because other investigators on site had spare hard disks that were partitioned with FAT 32.

In the forensic world we know that our expertise will be challenged repeatedly by the judicial system and we have to be consistent with all the procedures that we use and be able to explain the unexplainable. Basically we need to defend our work and the results attached to it. In our case the problem experienced by the investigators with the different file allocation table was reviewed and the process of wiping, formatting and partitioning was modified to be uniform and problem free when on site.

As of now every investigator has to include in their tool box at least one hard disk of both format (FAT 32 and NTFS). These hard drives will be wiped, formatted and partitioned the same way all the time. We use a wiping utility in accordance with the Department of National Defense standard 5220.22-M (DND) meaning three passes are performed on every disk, the disks are formatted then partitioned in FAT 32 (irrelevant of the disk sizes) or NTFS.

It was a learning experience that fortunately did not affect the end result of our investigation but made our inquiring minds work overtime to figure out the problem.

Resources

Fat 32 or NTFS: Making the choice
URL: http://www.theeldergeek.com/ntfs_or_fat32_file_system.htm

Hui, Andy, Fat 32 vs NTFS, August 8th, 2001
URL: http://www.anandtech.com/guides/viewfaq.html?I=63

XP Mania, NTFS vs. Fat 32
URL: http://www.pro-networds.org/XPMaNiA/fat32.shtml

Russel, Charlie, NTFS vs. Fat 32: Which is right for you? October 1st, 2001
URL:
http://www.microsoft.com/windowsxp/expertzone/columns/russel/october01.asp

Moir, Robert, Fat 32 or NTFS? March 30th, 2002
URL: http://www.robertmoir.co.uk/windows/FAT32or NTFS.html

Lewis, Brian, K, NTFS vs. Fat 32, Teck Talk 03/02
URL: http://www.spcug.org/reviews/bl0203.htm

Mikhailov, Dmitry, FAT and NTFS performance
URL: http://www.digit-life.com/articles/ntfs/index3.html

Stephan, Brian, S. Design of filesystems in modern operating systems, February 3rd, 2003
URL: http://people.msoe.edu/~taylor/cs384/stephanb.pdf

NTFS vs FAT
URL: http://ntfs.com/ntfs_vs_fat.htm

Encase Forensic
URL: http://www.guidancesoftware.com/