

Testing the Date Maintenance of the File Allocation Table File System

Tom Waghorn

Edith Cowan University
e-mail: twaghorn@student.ecu.edu.au

Abstract

The directory entries used in the File Allocation Table filesystems maintain a significant amount of file metadata that is of interest to a forensic examiner. This information is maintained by the operating system under normal conditions and may be amended by activities undertaken by the computer user. This paper examines the maintenance of date information when files are being moved between different versions of the FAT filesystem

Keywords

File systems, dates, file allocation tables, date maintenance, directory entries

INTRODUCTION

Microsoft DOS and Windows have used the File Allocation Table file system in various implementations for a number of years. Many computer users have used and continue to use this file system to store data on their computers. Consequently, the detailed operational parameters of this file system are of interest to forensic examiners and computer security personnel who make use of information that this file system can provide.

The purpose of the test was to determine the effect on file directory entries when files are moved between different implementations of the File Allocation Table file system. Files, when moved between different disks and locations, have been observed to exhibit anomalous behaviour, particularly in relation to the differing dates relating to each file. The tests were intended to determine whether date information is altered or lost and under what circumstances any changes to the date information can occur

FILE ALLOCATION TABLE FILE SYSTEM

There are three variations of the FAT file system namely: FAT12, FAT16 and FAT32. The FAT12 file system was the original implementation used by Microsoft for MS-DOS and is currently still in use for floppy disk media. This system uses 12 bits for each entry in the file allocation table. Similarly FAT16 and FAT32 use 16 and 28 bits for each table entry respectively. The number of bits used for each entry determine the number of allocation units that can be addressed in the file allocation table.(Anon, 2003a, 2003b; Sammes & Jenkinson, 2000).

So theoretically for FAT12 it is 212 or 4096 allocation units, however, this system can actually address only 4078 allocation units as the first two and several other addresses are reserved. Since each allocation unit is 512 bytes, this allows a total of 2 287 936 bytes to be catered for by this system. (Sammes & Jenkinson, 2000) This is adequate for a floppy disk but clearly is inadequate for a modern hard disk that may be several hundred gigabytes in size.

Using the file allocation file (FAT) system, the filename and various pieces of date and time information are stored in the directory entries for each file rather than as a part of each file. Each file has a corresponding 32 byte directory unless long file names are used, in which multiple 32 byte entries are used.(Sammes & Jenkinson, 2000)

Directory entries in the File Allocation Table (FAT) file system, in its current implementation, record three different types of date or date and time information for each file. These times are:

- the date and time of the last change to file,
- the date and time the file was first written, and
- the date the file was last accessed.

Earlier implementations of the FAT file system recorded only the date and time of the last change to a file. (Sammes & Jenkinson, 2000)

Structure of directory entry

Each directory entry, ignoring long file name entries, contains information about its corresponding file. This information includes the name of the file, file attributes such as read-only, hidden status, whether the entry refers to a volume or sub-directory rather than a file as well as the date and time information detailed previously.

Offset	Byte Offset	Information Recorded
00 - 07h	1 - 8	Filename - 8 bytes padded with spaces
08 - 0ah	9 - 11	File extension - 3 bytes padded with spaces
0bh	12	File attributes - 1 byte
0c - 15h	13 - 22	Reserved (MS-DOS) / Additional time and date fields
	13	Reserved
	14	10 millisecond units past creation time - 1 byte
	15 - 16	File creation time - 2 bytes
	17 - 18	File creation date - 2 bytes
	19 - 20	Last access date - 2 bytes
	21 - 22	High word of start cluster (FAT32) - 2 bytes
16 - 17h	23 - 24	Time of last change - 2 bytes
18 - 19h	25 - 26	Date of last change - 2 bytes
1a - 1bh	27 - 28	First cluster - 2 bytes
1c - 1fh	29 - 32	File size - 4 bytes

*Table 1: Structure of 32 Byte Directory Entry
(Sammes & Jenkinson, 2000p 160, p 166)*

The Byte Offset value in table 1 is the number of bytes from the start of the individual directory entry. The locations of the critical date information for each separate directory entry are located at the following byte offsets :

- Byte Offset 17 - 18 File Creation Date
- Byte Offset 19 - 20 Last Access Date
- Byte Offset 25 - 26 Date of Last Change

Calculation of Dates

Each of the 2 byte or 16 bit values can be divided into three distinct sections that provide the relevant date information.

- The first seven bits provide the number of years from 1980.
- The next four bits provide the number of the month.
- The remaining five bits provide the day of the month.

An example of this is the first creation date for document Floppy_1.doc that was created as a part of the testing process. The date set on the computer was 1 January 2001 when the document was created.

An examination of byte offset 17 - 18 for the relevant directory entry, revealed the value "212Ah". As this is recorded in a little endian fashion, the translation is actually 00101010 00100001.

Byte Offset	Value (Hex)	Value (Binary)	Year	Month	Day
16 - 17	212A	00101010 00100001	7 bits 0010101	4 bit 0001	5 bit 00001
		Decimal Value	21	1	1

*Table 2: Calculation of Date information from Directory Entry Data
(Sammes & Jenkinson, 2000)*

To finalise the calculation process, the year value is added to 1980, while the other values represent the month and day respectively. Thus the date being represented is :

(1980 + 21 = 2001), month value of 1 and the day value of 1 or 1 January 2001 which corresponds with the known date of the file creation.

TESTING PROCESS

The test involved the creation of a series of Microsoft Word documents in each of the different FAT file systems. These documents were moved to a different file system in a sequence of steps and the date and time information was recorded after each move.

Software / Equipment	Purpose
Windows 98 Second Edition	Operating system supporting FAT12, FAT16 and FAT32 file systems
Encase 3.22g	Forensic imaging software
FastBloc write protection device	Enables imaging of disk while preventing the operating system from changing the disk contents
Floppy disk	Container for FAT file system
340 Mb hard disk	Container for FAT 16 and FAT32 partitions

Table 3: Table of Software and Equipment used for the Test

The test platform chosen was Windows 98 Second Edition. This platform was chosen as Windows 98 was the first Microsoft Windows system that supported FAT32 file systems natively.

To further preserve the integrity of the test results, both the floppy disk and the hard disk containing the FAT16 and FAT32 partitions were imaged using a forensic imaging software called Encase. This software creates an image of the target disk without making any changes to the disk and enables users to examine the contents of each sector of the disk.

The reason for imaging the disks was that it enabled the date information to be obtained without the operating system being able to contaminate the test results. This process has the added advantage of allowing the results to be preserved and examined at a later time without risking the integrity of the data.

The images of the floppy disk were acquired using the write protect tab activated and, the hard disk was acquired using a "Fastbloc" hardware write protection device.

The testing process was designed to ensure that any date contamination would not occur due to operating system behaviour. To this end, the dates were set in the BIOS of the test computer prior to booting into the operating system. Additionally, the imaging of the disks took place using the same dates as the movement process.

The testing process was conducted in four phases. The Base phase was used to set the experimental conditions and each subsequent phase was used to copy files between the different file system to examine the effects on the dates and times associated with each file being moved.

BASE PHASE

This phase was intended to provide the baseline for the experiment.

Phase	Document	File System	Location
Base Phase Date : 1 January 2001	Floppy 1.doc	FAT12	Floppy Disk
	Floppy 2.doc	FAT12	Floppy Disk
	FAT16 1.doc	FAT16	Primary Partition
	FAT16 2.doc	FAT16	Primary Partition
	FAT32 1.doc	FAT32	Extended Partition
	FAT32 2.doc	FAT32	Extended Partition

Table 4: Creation of Files for Base Phase

The computer to be used had a fresh install of Microsoft Windows 98 Second Edition and the date was set to 1 January 2001 using the BIOS. This date was selected because the computer BIOS was date limited and did not allow any date prior to 1999. The year 2001 was selected to assist in numerically identifying each phase of the experiment starting at 1.

All of the disks to be used in the experiment, a 1.44 Mb floppy disk and a 340 Mb hard disk, were wiped using Encase. This process overwrites every sector of the disk with a preset value (00h) and prevents the possibility of date contamination from data recorded on the disk prior to the experiment.

The floppy disk was then formatted with a FAT12 file system. The hard disk was partitioned into a 100 Mb primary partition, formatted with a FAT16 file system and a 240 Mb extended partition formatted with a FAT32 file system.

Two Microsoft Word documents were created in each disk or partition with the filenames reflecting the locations where the documents were created. Each document was created separately and originally saved into designated creation location.

PHASE 1

Phase	Document	Originating File System	Destination File System
Phase 1 Date : 2 February 2002	Floppy_1.doc	FAT12	FAT16
	Floppy_2.doc	FAT12	FAT32
	FAT16_1.doc	FAT16	FAT12
	FAT16_2.doc	FAT16	FAT32
	FAT32_1.doc	FAT32	FAT12
	FAT32_2.doc	FAT32	FAT16

Table 5: Movement of Files for Phase 1

This series of tests provides for documents that were natively created in a particular file system to be copied to each of the other two file systems. The intent of these tests is to determine the effects on the three dates recorded by the file system directory entries. In particular, the tests were intended to determine whether the date information is affected differently moving to or from a file system employing a greater or lesser number of bytes in each table entry.

These tests allow the moved files to be directly compared with the source files to determine what changes are made to the recorded dates. Any changes to the recorded dates of the source files can be directly compared with the data obtained the same files in the Base Phase.

PHASE 2

Phase	Movement	Document	Originating File System	Destination File System
Phase 2 Date: 3 March 2003	1	Floppy_1.doc	FAT16	FAT32
	2	Floppy_2.doc	FAT32	FAT12
	3	FAT16_1.doc	FAT12	FAT16
	4	FAT32_1.doc	FAT12	FAT16
	5	FAT32_2.doc	FAT16	FAT12

Table 6: Movement of Files for Phase 2

This series of tests provides for documents that have been previously moved to be subject to a secondary movement. The intent is to determine the whether secondary movement exhibit the same date change activity or to detect whether subsequent movement exhibit changes that are different to original movements. Again the

movements involve both movements to file systems involving greater and lesser numbers of bytes in their table entries.

Movements

1. Floppy_1.doc, which originated in a FAT12 system and was moved to a FAT16 system is subsequently copied to a FAT32 system.
2. Floppy_2.doc is being returned to the FAT12 system and has not passed through the FAT16 system. This move will overwrite the original document.
3. FAT16_1.doc was originally moved from a FAT16 system to a FAT12 system is being returned. The original file will be overwritten.
4. FAT32_1.doc was originally moved to a FAT12 system is being subsequently moved to a FAT16 system.
5. FAT32_2.doc was originally moved to a FAT16 system is being subsequently moved to a FAT12 system.

PHASE 3

Phase	Movement	Document	Originating File System	Destination File System
Phase 3 Date: 4 April 2004	1	Floppy_1.doc	FAT32	FAT16
	2	FAT16_1.doc	FAT16	FAT32
	3	FAT16_2.doc	FAT32	FAT16
	4	FAT32_1.doc	FAT16	FAT32
	5	FAT32_2.doc	FAT12	FAT32

Table 7: Movement of Files for Phase 3

This series of tests provides a continuation of the tests employed in Phase 3. Some files are subjected to a secondary movement while other files are subjected to a further secondary movement.

Movements

1. Floppy_1 .doc is subject to a further secondary movement from a FAT32 system to a FAT16 system.
2. FAT16_1 is subject to a further secondary movement. This file has originated in a FAT16, moved to a FAT12, returned to a FAT16 and is now being moved to a FAT32 system.
3. FAT16_2.doc was moved to a FAT32 system in phase 1 and the return was delayed to allow FAT16_1.doc to pass through that partition. This will overwrite the original file.
4. FAT32_1.doc having passed through the FAT12 and the FAT16 is being returned to the FAT32 system. This will overwrite the original file.
5. Fat32_2.doc having passed through the FAT16 and FAT12 systems is being returned to the FAT32 system. This will overwrite the original file.

PHASE 4

Phase	Movement	Document	Originating File System	Destination File System
Phase 4	1	FAT16_1.doc	FAT32	FAT16

Table 7: Movement of Files for Phase 4

This test provides for a continuation of the tests employed in previous phases. The file being moved is completing a sequence of movements through all three file systems and is being returned to its originating file system to determine the changes made to its date information.

Movements

1. FAT16_1.doc having passed through FAT12, FAT16 and FAT32 systems, will be returned to the FAT16 system and overwrite the existing file.

RESULTS

The results presented in the following tables demonstrate the calculated dates of dates and times for each file.

During the different testing phases no amendment were made to any of the files. They were not opened by any application and the only process to which they were subject was the actual copy process.

The File Creation date could be reasonably be expected to have remained unchanged as the file were created only once. Again the Last Written date could also be expected to remain unchanged as these files have all remained unedited or unchanged by any application.

BASELINE

Disk		File Name	Last Accessed	Last Written	File Created
Floppy	1	Floppy 1.doc	01/01/01	01/01/01 01:29:02PM	01/01/01 01:28:58PM
	2	Floppy 2.doc	01/01/01	01/01/01 01:29:28PM	01/01/01 01:29:24PM
FAT16	3	FAT16 1.doc	01/01/01	01/01/01 01:32:02PM	01/01/01 01:31:58PM
	4	FAT16 2.doc	01/01/01	01/01/01 01:33:24PM	01/01/01 01:33:22PM
FAT32	5	FAT32 1.doc	01/01/01	01/01/01 01:34:12PM	01/01/01 01:34:10PM
	6	FAT32 2.doc	01/01/01	01/01/01 01:34:32PM	01/01/01 01:34:30PM

Table 8: Date Information for Baseline Files

The date and times displayed by the files correspond with the dates and times of the creation of each of the files.

The Last Written dates again match the creation dates of the files, which corresponds with the dates of their original creation. However, it is worth noting the slight differences in the times between the File Created values and the Last Written values.

The FAT12 and FAT16 file systems (Table 8, ref 1, 2, 3 & 4) uniformly show a four second difference between the time the file was first created and the time it was last written. This would indicate that the actual writing process took four seconds between the start (File Created) and completion (Last Written) of the file creation process.

The same process on the FAT32 file system (Table 8, ref 5 & 6) uniformly shows a two second difference between the File Created and the Last Written times. The reason for this difference between the file systems is not readily apparent, however it can be conjectured that the FAT32 file system may be handled more efficiently as the FAT16 and FAT32 file systems both reside on the same hard disk.

PHASE 1

Disk		File Name	Last Accessed	Last Written	File Created
Floppy	1	FAT16_1.doc	02/02/02	01/01/01 01:32:02PM	02/02/02 02:25:46PM
	2	FAT32_1.doc	02/02/02	01/01/01 01:34:12PM	02/02/02 02:25:56PM
	3	Floppy_1.doc	02/02/02	01/01/01 01:29:02PM	01/01/01 01:28:58PM
	4	Floppy_2.doc	02/02/02	01/01/01 01:29:28PM	01/01/01 01:29:24PM
FAT16	5	FAT16_1.doc	02/02/02	01/01/01 01:32:02PM	01/01/01 01:31:58PM
	6	FAT16_2.doc	02/02/02	01/01/01 01:33:24PM	01/01/01 01:33:22PM

	7	FAT32_2.doc	02/02/02	01/01/01 01:34:32PM	02/02/02 02:26:04PM
	8	Floppy_1.doc	02/02/02	01/01/01 01:29:02PM	02/02/02 02:25:30PM
FAT32	9	FAT16_2.doc	02/02/02	01/01/01 01:33:24PM	02/02/02 02:25:50PM
	10	FAT32_1.doc	02/02/02	01/01/01 01:34:12PM	01/01/01 01:34:10PM
	11	FAT32_2.doc	02/02/02	01/01/01 01:34:32PM	01/01/01 01:34:30PM
	12	Floppy_2.doc	02/02/02	01/01/01 01:29:28PM	02/02/02 02:25:36PM

Table 9: Date Information for Phase 1 Files

The significant changes that were noticed for all of the files were the changes to the Last Accessed dates. In all cases, this was changed to reflect the date that the files were copied from one location to another. This updating of the Last Accessed date to the current date, even for the files that were sourced and not written, indicates that this date is amended by a process that does not necessarily open the file in an application.

The document Floppy_1.doc has the last accessed date of 2 February 2002 for both the copy located on the FAT16 partition as well as the original document, which was unchanged or unused in any other fashion other than to provide a copy source for the document being written to the FAT16 partition.

Another point of significance is the changes to the File Created date. The original copies of the documents retain their original File Created dates, however the copies of those documents have a File Created date that reflects the dates they were copied to the new locations. (Table 9, ref 1, 2, 7, 8, 9 & 12) This results in a file having a Last Written date that predates the corresponding File Created date. This at first appears to be contradictory but may be seen as an indicator that the file was originally created elsewhere and copied to that location.

It should also be noted that there is no loss of date information when the files are moved from one location to another, including to the floppy disk. All three dates are recorded under each of the three file systems.

PHASE 2

Disk		File Name	Last Accessed	Last Written	File Created
Floppy	1	FAT16_1.doc	03/03/03	01/01/01 01:32:02PM	02/02/02 02:25:46PM
	2	FAT32_1.doc	03/03/03	01/01/01 01:34:12PM	02/02/02 02:25:56PM
	3	FAT32_2.doc	03/03/03	01/01/01 01:34:32PM	03/03/03 02:50:42PM
	4	Floppy_1.doc	02/02/02	01/01/01 01:29:02PM	01/01/01 01:28:58PM
	5	Floppy_2.doc	03/03/03	01/01/01 01:29:28PM	01/01/01 01:29:24PM
FAT16	6	FAT16_1.doc	03/03/03	01/01/01 01:32:02PM	01/01/01 01:31:58PM
	7	FAT16_2.doc	02/02/02	01/01/01 01:33:24PM	01/01/01 01:33:22PM
	8	FAT32_1.doc	03/03/03	01/01/01 01:34:12PM	03/03/03 02:51:04PM
	9	FAT32_2.doc	03/03/03	01/01/01 01:34:32PM	02/02/02 02:26:04PM
FAT32	10	Floppy_1.doc	03/03/03	01/01/01 01:29:02PM	02/02/02 02:25:30PM
	11	FAT16_2.doc	02/02/02	01/01/01 01:33:24PM	02/02/02 02:25:50PM
	12	FAT32_1.doc	02/02/02	01/01/01 01:34:12PM	01/01/01 01:34:10PM
	13	FAT32_2.doc	02/02/02	01/01/01 01:34:32PM	01/01/01 01:34:30PM
	14	Floppy_1.doc	03/03/03	01/01/01 01:29:02PM	03/03/03 02:44:54PM
	15	Floppy_2.doc	03/03/03	01/01/01 01:29:28PM	02/02/02 02:25:36PM

Table 10: Date Information for Phase 2 Files

The results for this phase reflect the results obtained in phase 1. The files involved in any movement reflect the change in the Last Access dates to the date of the file movement. (Table 10, ref 1, 2, 3, 5, 6, 8, 9, 10, 14 & 15) Waghorn (Paper #13)

Those files not involved in any movement as either a source or destination show no changes. (Table 10, ref 4, 7, 11, 12 & 13)

The file Floppy_2.doc (Table 10, ref 5) was copied from the FAT32 partition and the original file overwritten. Had the previously observed behaviour continued, the File Created date should have changed to 3 March 2003. This however has not occurred and the File Written date reflects the same date and time as the original file. This is also observed to have occurred with file FAT16_1.doc. (Table 10, ref 6) These movements involved the files being overwritten and the existing directory entries would have been expected to be updated.

The FAT32 documents, FAT32_1.doc (Table 10, ref 8) and Fat32_2.doc (Table 10, ref 2) as well as Floppy_1.doc (Table 10, ref 14), each exhibited the previously observed behaviour of the change of the File Created date to reflect the current date and time. The thing that these movements have in common is that a directory entry for a file of that name did not exist in the target partitions prior to these moves occurring.

PHASE 3

Disk		File Name	Last Accessed	Last Written	File Created
Floppy	1	FAT16_1.doc	03/03/03	01/01/01 01:32:02PM	02/02/02 02:25:46PM
	2	FAT32_1.doc	03/03/03	01/01/01 01:34:12PM	02/02/02 02:25:56PM
	3	FAT32_2.doc	04/04/04	01/01/01 01:34:32PM	03/03/03 02:50:42PM
	4	Floppy_1.doc	02/02/02	01/01/01 01:29:02PM	01/01/01 01:28:58PM
	5	Floppy_2.doc	03/03/03	01/01/01 01:29:28PM	01/01/01 01:29:24PM
FAT16	6	FAT16_1.doc	04/04/04	01/01/01 01:32:02PM	01/01/01 01:31:58PM
	7	FAT16_2.doc	04/04/04	01/01/01 01:33:24PM	01/01/01 01:33:22PM
	8	FAT32_1.doc	04/04/04	01/01/01 01:34:12PM	03/03/03 02:51:04PM
	9	FAT32_2.doc	03/03/03	01/01/01 01:34:32PM	02/02/02 02:26:04PM
	10	Floppy_1.doc	04/04/04	01/01/01 01:29:02PM	02/02/02 02:25:30PM
FAT32	11	FAT16_1.doc	04/04/04	01/01/01 13:32:02PM	04/04/04 15:17:10PM
	12	FAT16_2.doc	04/04/04	01/01/01 13:33:24PM	02/02/02 14:25:50PM
	13	FAT32_1.doc	04/04/04	01/01/01 13:34:12PM	01/01/01 13:34:10PM
	14	FAT32_2.doc	04/04/04	01/01/01 13:34:32PM	01/01/01 13:34:30PM
	15	Floppy_1.doc	04/04/04	01/01/01 13:29:02PM	03/03/03 14:44:54PM
	16	Floppy_2.doc	03/03/03	01/01/01 13:29:28PM	02/02/02 14:25:36PM

Table 11: Date Information for Phase 3 Files

The results obtained from this phase are similar to those attained in phase 2.

The only movement that involves moving a file to a new location, where a directory entry does not exist for a file of that name, is the movement of file FAT16_1.doc to the FAT32 file system (Table 11, ref 11). This exhibits the expected behaviour of updating the Last Access date and the File Created date to the current date of the test computer.

All of the other moves involve moving files to partitions where a directory entry previously exists for a file of that name. Floppy_1.doc was moved from the FAT32 system to FAT16 system where it already exists (Table 11, ref 6). The file should overwrite the existing file and the directory entry should be updated. The Last Access dates are updated for both files, but the File Created date is unchanged in the target partition.

Where a file is overwriting an existing copy, the Last Accessed date reflects the current date and time of the test computer. The File Created date however, reflects the date and time of the original directory entry.

PHASE 4

Disk		File Name	Last Accessed	Last Written	File Created
Floppy	1	FAT16_1.doc	03/03/03	01/01/01 01:32:02PM	02/02/02 02:25:46PM
	2	FAT32_1.doc	03/03/03	01/01/01 01:34:12PM	02/02/02 02:25:56PM
	3	FAT32_2.doc	04/04/04	01/01/01 01:34:32PM	03/03/03 02:50:42PM
	4	Floppy_1.doc	02/02/02	01/01/01 01:29:02PM	01/01/01 01:28:58PM
	5	Floppy_2.doc	03/03/03	01/01/01 01:29:28PM	01/01/01 01:29:24PM
FAT16	6	FAT16_1.doc	05/05/05	01/01/01 01:32:02PM	01/01/01 01:31:58PM
	7	FAT16_2.doc	04/04/04	01/01/01 01:33:24PM	01/01/01 01:33:22PM
	8	FAT32_1.doc	04/04/04	01/01/01 01:34:12PM	03/03/03 02:51:04PM
	9	FAT32_2.doc	03/03/03	01/01/01 01:34:32PM	02/02/02 02:26:04PM
	10	Floppy_1.doc	04/04/04	01/01/01 01:29:02PM	02/02/02 02:25:30PM
FAT32	11	FAT16_2.doc	04/04/04	01/01/01 01:33:24PM	02/02/02 02:25:50PM
	12	FAT32_1.doc	04/04/04	01/01/01 01:34:12PM	01/01/01 01:34:10PM
	13	FAT32_2.doc	04/04/04	01/01/01 01:34:32PM	01/01/01 01:34:30PM
	14	Floppy_1.doc	04/04/04	01/01/01 01:29:02PM	03/03/03 02:44:54PM
	15	Floppy_2.doc	03/03/03	01/01/01 01:29:28PM	02/02/02 02:25:36PM

Table 12: Date Information for Phase 4 Files

There is only one file being moved in this phase. File FAT16_1.doc is being copied from the FAT32 system to the FAT16 system (Table 12, ref 6), where a directory entry already exists.

The Last Accessed date reflects the date of the test computer and the File Created date and time reflects the original creation of the file.

DISCUSSION OF RESULTS

The behaviour of the three different variations of the File Allocation Table file system appear to be consistent.

Any process that involves a file, appears to cause the operating system to update the Last Accessed date of the directory entry for that file, whether the file is opened by an application or not. This is demonstrated by the updating of the Last Accessed dates for all file movements. This includes the source file as well as the target file. The source file is not opened by any application not is it subjected to any process likely to change the file contents, however the Last Accessed dates are still updated.

Where no amendments are made to a file, no changes are made to the Last Written dates. This was the case for all movements observed during these tests and no changes to these dates were observed in any of the four phases.

Copying files form one location to another appears to trigger a standard set of actions depending on whether a directory entry exists or otherwise. Where a directory entry for a particular file does not exist, the File Created date is set to reflect the current date and time of the computer. Where an entry already exists, the File Created date is retained.

While this behaviour makes determining the exact history of file movements possible under certain conditions, it adds a level of complexity to determining the exact File Creation dates unless the original location can be accurately determined. If the original location can be determined then overwriting a file will not change the original information.

It should be noted that none of the files involved in this series of tests were changed in any way during the tests. As the directory entry also records information such as the file size, it is possible that any changes to other parts

of the directory entry will trigger an update to the File Created date on an overwrite operation. This was not tested during this series of tests.

Clearly the movement of files between partitions is insufficient to trigger the operating system to overwrite an existing directory entry, other than amending the Last Accessed date.

Logically, it would be reasonable to assume that once a directory entry has been created then no operation on an existing file should affect the File Created date. This is based on the assumption that the File Created date is recorded when the directory entry is initially created and that subsequent changes to the directory entry would be recorded as amendments rather than a new creation. Proving this would involve changing each of the other directory entry components, both individually and in groups.

The directory entry would need to be updated by copying an amended file into the directory, triggering an overwrite operation of the existing directory entry. It is possible that the only triggering factor may be the changing of the filename, which would be expected to result in a new directory entry being made. A filename change that did not occur during a copy operation would be expected to result in an amendment to the existing directory entry and should therefore not be expected to change the First Written date.

If this proves to be the case, then the First Written date for a file would only be expected to be recorded when a new directory entry is made, not during any amendment to an existing entry.

CONCLUSION

From a forensic practitioner's viewpoint, the times and dates recorded on a file cannot be accepted as completely accurate without other supporting evidence due to the ease in which it possible to manipulate the computer's system clock. Additionally, the system clock could be incorrectly set or not be completely accurate. However, the practical value of this research is in the understanding of the relationship between the File Created and Last Written dates.

Files that have dates in the sequence of File Created date prior to Last Written date, assuming no manipulation of the directory entry, would be strongly indicative of the creation of the file in it's current location. This creation process could involve opening and saving a new file or downloading the file into it's current location as this would result in the creation of a new directory entry. Similarly, it is reasonable to expect that where these dates are found to be in reverse order, the file was not created in that location but was created elsewhere and copied to the current location.

This difference can be used to provide a relatively accurate way of determining whether files have been 'downloaded from the Internet' or 'copied from another disk or location'. This is particularly so where a single directory has files that exhibit both forms of behaviour. This aspect has been of significance in indicating the origin and history of files can be used to verify personal accounts of how particular files came to exist in their current situation.

FURTHER RESEARCH

There are several avenues that appear to warrant further investigation based on the results of the experiment. As much of the behaviour exhibited may be attributed to the operation of the operating system, repeating this experiment with other variations of Windows such as Windows 95, Windows 2000 and Windows XP would reveal whether this behaviour is consistent for all forms of the operating system.

Adding a partition formatted in New Technology File System (NTFS) would also be of value in determining the effect of a non-FAT file system on the observed behaviour. Additionally, the use of VFAT file system handlers in one or more the Linux distributions would provide a Windows independent platform for further comparison.

Additionally, further experimentation to determine any possible triggering event that might result in an existing directory entry being given an update File Creation date would be of some value. These experiments would include overwriting existing directory entries with a variety of changes to see if any specific factor or combination of factors would result in this date value being changed or proving that it is only written for a new directory entry.

REFERENCES

- Anon. (2003a). FAT Filesystem. Retrieved 29 April 2003, 2003, from <http://xinit.port5.com/files/fat.htm>
- Anon. (2003b). The FAT filesystem. Retrieved 29 April 2003, 2003, from <http://www.win.tue.nl/~aeb/linux/fs/fat/fat-1.html>

Sammes, T., & Jenkinson, B. (2000). *Forensic Computing*. London: Springer-Verlag.

BIBLIOGRAPHY

Leclercq, X. (1999, 26 June 1999). The FAT filesystem. Retrieved 19 April 2003, from <http://www.xaff.org/GI/fat.html>

Anon. What is the file allocation. Retrieved 29 April 2003, from <http://modecideas.com/faq81.html?newitems>

Giese, C. (2002). Filesystems. Retrieved 29 April 2003, from <http://osdev.berlios.de/osd-fs.html>

COPYRIGHT

Thomas Waghorn © 2003. The authors assign to ACIS and educational and non-profit institutions a non-exclusive licence to use this document for personal use and in courses of instruction provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive licence to ACIS to publish this document in full in the Conference Papers and Proceedings. Those documents may be published on the World Wide Web, CD-ROM, in printed form, and on mirror sites on the World Wide Web. Any other usage is prohibited without the express permission of the authors.