Global Information Assurance Certification Paper

Windows Alternate Data Streams

GSEC Program Practical
Security Essentials Certification Training
GSEC – Basic Practical Assignment
Version 1.4b (August 29, 2002)
Submitted by Kurtis E. Kroeckel

Abstract/Summary:

The subject I investigated is about Windows Operating System NTFS ability to have alternate data streams attached to valid files and directories. I believe this NTFS feature has posed a threat to NTFS in the past and poses a threat today. If steps are not taken, the NTFS alternate data stream feature will be an upcoming threat in the near future.

This paper will answer the following questions. What is an Alternate Data Stream? Why are they used? Why should this be a concern? What tools are available? How to protect against or remove? Finally, I wrap up with attacks of the future related to the alternate data stream.

What is an Alternate Data Stream (ADS)?

"The NTFS file system supports multiple data streams. The stream name identifies a new data attribute on the file. Streams have separate opportunistic locks, file locks, allocation sizes, and file sizes, but sharing is per file." [1] Microsoft lists alternate data streams within the NTFS multiple data streams component. Directories treat alternate data streams no different than files do. Directories can have alternate data streams attached just like files.

An understanding of how NTFS stores information on files is needed before a person can understand how or why alternate data streams are used. All allocated sectors within an NTFS volume are associated to a file. A file is made up of all the data within the file and the file's metadata. The metadata consists of such items as the file name, attribute type code, possibility an attribute name and file security information. The file metadata and file data are considered as a combined set. The NTFS views the combined set as the file attributes. Resident attributes for a file are the attribute that can fit within the Master File Table (MFT) record and the Master File Table is 1,500 bytes in size. The MFT contains among the base file record for each file and directory within the NTFS volume besides other file records. The file name and time stamp are always kept as resident attributes. If the attributes for the file cannot completely fit in the MFT file record, some of the file attributes become nonresident. The nonresident attributes are allocated to one or more clusters of disk space. These clusters become alternate data streams within the NTFS volume. An Attribute List attribute is created that describes the location of both the resident and nonresident attribute records. This is how the file system can keep track of where the file data resides.

The NTFS file attribute "Data" contains the file data. Multiple data attributes are allowed for each file. Typically, a file has one unnamed data attribute which is the data stream default for an NTFS file. It is referred to as "$DATA". The file can have one or more named data attributes and this is where alternate data streams are held. Each stream has separate file locks, allocation sizes, and file sizes but all the stream sharing is per file. It is worthy to note that the permissions of the file extend to all attributes which include data streams. This means that to access a file, the program or individual must possess the appropriate credentials.

Why are Alternate Data Streams used?

Streams can be used for library files, file association, and other reasons. "A library of files might exist where the files are defined as alternate streams, as in the following example: library:file1:file2:file3"[2] or "A file can be associated with more than one application at a time, such as Microsoft® Word and Microsoft® WordPad. For instance, a file structure like the following illustrates file association, but not multiple files: program:source_file:doc_file:object_file:executable_file. Another way to create

an alternate data stream is to use the Win32 advanced programming interface (API) CreateFile which will create the alternate data stream."[3] Other Operating Systems use alternate data streams to perform various functions. Macintosh computers use the same type of structure to manage resource and data forks within the Mac OS Hierarchical File System (HFS). [4] Microsoft applications use alternate data streams to attach to image files like .jpg, .gif, and .bmp. One Microsoft application that does this is the Windows 2000 Content Indexing Server. The file attachment is done by design and is a filter driver that will create thumbnail images.[5]

Why should this be a concern?

Alternate data streams deal with the integrity of the file on the server or being received. A person will not know by just looking at the file if it contains an extra executable or text within an alternate data stream. Alternate data streams deal with confidentiality of the file being sent or at rest on the system. This issue can be from a program capturing keystrokes or passwords and redirecting the output to an alternate data stream. This file could be emailed innocently and the attacker could then retrieve the information.

Viruses can use this method to be stealthy. The virus files could be hidden in a number of alternate streams within one file or multiple files. The virus would only have to have a small visible footprint on the system. This will make it very difficult to discover since most of the virus information will not show up with the available native Window tool sets.

I recently took the Tripwire for Servers class given by Tripwire. I had never heard of alternate data streams until then. Tripwire for Servers can find and does monitor files and data streams. There is one caveat to doing the baseline for a NTFS system. If a file is a zero byte file, Tripwire for Servers can only baseline for one data stream.

What tools are available to locate Alternate Data Streams (ADSs)?

The computer forensics area would be interested in how to find alternate data streams. If a person believes that hidden data could be on a NTFS disk, tools and procedures would be needed to find the hidden data. A skilled programmer can use procedures that are available within the Microsoft Developer Network (MSDN) library. The WIN32_STREAM_ID structure has a pointer to it that will be used and a pointer to the tape backup functions that work with the WIN32_STREAM_ID structure. The code needed is explained in an article by David LeBlanc he did for Security Administrator. Look at this endnote to find the article and more information on the programming code. Alternate data stream detection is not a trivial process to code for. [6]

There is a product called "List Alternate Data Streams" that will scan an NTFS disk or just a directory. The executable was written by Frank Heyne and is

freely available. The link is listed in the endnote. The tool is very easy to use in the command line. Download the zip file and extract into a directory you want.

```
c:\lads>lads c:
The results of the command are:
LADS - Freeware version 3.10
(C) Copyright 1998-2002 Frank Heyne Software (http://www.heysoft.de)
This program lists files with alternate data streams (ADS)
Use LADS on your own risk!

Scanning directory c:\

     size  ADS in file
----------  --------------------------------
Error 32 opening c:\hiberfil.sys: The process cannot access the file because it
is being used by another process
         128     c:\kek.txt:• SummaryInformation
     1004032     c:\kek.txt:notekek.exe
       66048     c:\kek.txt:tasklist.txt
          17     c:\kek.txt:title
           0     c:\kek.txt:{4c8cc155-6c1e-11d1-8e41-00c04fb9386d}
Error 32 opening c:\pagefile.sys: The process cannot access the file because it
is being used by another process
       66048     c:\seeifrun.txt:notekek.exe

The following summary might be incorrect because there was at least one error!

   1136273 bytes in 6 ADS listed

C:\>dir kek.txt*
 Volume in drive C has no label.
 Volume Serial Number is ACED-A288

 Directory of C:\

04/11/2003  04:52 PM                24 kek.txt
               1 File(s)             24 bytes
               0 Dir(s)   1,395,851,264 bytes free
```

The file system shows that c:\kek.txt and c:\seeifrun.txt are only 24 bytes each. The LADS program does find and report the size of the alternate data streams.[7]

How to protect against or remove Alternate Data Streams.

There is a Registry entry that can be set to stop Windows 2000 Content Indexing Services from adding an alternate data stream to .jpg, .gif, and .bmp files on NTFS volumes. The Dynamic Link Library (DLL) filter driver makes

thumbnail images in an alternate data stream named ?Q30lsldxJoudresxAaaqpcawXc. Using the Registry Editor, navigate to HKEY_LOCAL_MACHINES\System\Currentcontrolset\Control\Contentindex and delete the FilterTrackers Value Name. Remember, that modifying the Windows Registry is at your own risk because it can make the system inoperable. [8] This is an example of what an alternate stream can be used for within an application. The situation here is that the integrity of the information could be tainted. This is from the fact the Windows operating system will not be able to detect any changes to the information. There are no native tools that can do this. Also, the confidentiality of the information is at jeopardy. The information contained within the thumbnail could have been compromised and again the operating system does not have tools to detect this.

As I researched the Internet, alternate data streams cannot be remove unless the file was copied to a FAT file system and then copied back. I did the following commands to remove these alternate data streams I had created on my system.

```
      128  c:\kek.txt:• SummaryInformation
1004032  c:\kek.txt:notekek.exe
   66048  c:\kek.txt:tasklist.txt
      17  c:\kek.txt:title
       0  c:\kek.txt:{4c8cc155-6c1e-11d1-8e41-00c04fb9386d}
   66048  c:\seeifrun.txt:notekek.exe
c:\>del c:\kek.txt
```

I ran the LADS program and there were no alternate data streams left associated with the file c:\kek.txt. The seeifrun.txt file I wanted to just remove the alternate data stream. I did the following commands.

```
C:\>ren seeifrun.txt hold.txt
C:\>type hold.txt > seeifrun.txt
C:\>del hold.txt
```

I ran the LADS program again and it showed 0 alternate data streams.
At least in Windows XP, alternate data streams can be removed but it is a clumsy procedure at best.

Let us assume you know there is a file important.exe with an alternate data stream attached to it. The file is very important and the ADS very dangerous. You need to hold the main stream and delete the ADS. Let us assume there is no FAT drive on your network, otherwise you could move the file to this drive and than move it back again. All you need to do is the following commands in Windows XP.

```
ren important.exe temp.exe
cat temp.exe > important.exe
del temp.exe
```

I wanted to see if these commands worked on a directory in removing an alternate stream but keeping the directory in tact.

```
C:\>mkdir kek
```

```
C:\>copy windows\notepad.exe kek\notepad.exe
      1 file(s) copied.
C:\>dir kek
 Volume in drive C has no label.
 Volume Serial Number is ACED-A288

 Directory of C:\kek

04/12/2003  05:58 PM    <DIR>          .
04/12/2003  05:58 PM    <DIR>          ..
08/18/2001  07:00 AM            66,048 notepad.exe
               1 File(s)         66,048 bytes
               2 Dir(s)   1,396,504,064 bytes free
```

The following is how to create an alternate data stream.

```
C:\>echo "this is an alternate text data stream" > c:\kek:textads
C:\>type windows\notepad.exe > c:\kek:executableADS.exe
C:\>dir kek
 Volume in drive C has no label.
 Volume Serial Number is ACED-A288

 Directory of C:\kek

04/12/2003  06:02 PM    <DIR>          .
04/12/2003  06:02 PM    <DIR>          ..
08/18/2001  07:00 AM            66,048 notepad.exe
               1 File(s)         66,048 bytes
               2 Dir(s)   1,396,435,456 bytes free
```

Now look at the LADS report.

```
c:\>lads c:\kek

LADS - Freeware version 3.10
(C) Copyright 1998-2002 Frank Heyne Software
(http://www.heysoft.de)
This program lists files with alternate data streams (ADS)
Use LADS on your own risk!

Scanning directory c:\kek\

     size  ADS in file
----------  ------------------------------
    66048  c:\kek\:executableADS.exe
       42  c:\kek\:textads

    66090 bytes in 2 ADS listed
```

I used the following procedure in Windows XP to keep the directory information but remove the alternate data stream.

```
C:\>move c:\kek c:\NewKEK
       1 file(s) moved.
```

c:\>lads NewKEK Will report that the alternate data streams are there.

```
LADS - Freeware version 3.10
(C) Copyright 1998-2002 Frank Heyne Software
(http://www.heysoft.de)
This program lists files with alternate data streams (ADS)
Use LADS on your own risk!

Scanning directory c:\NewKEK\

     size  ADS in file
----------  ------------------------------
    66048  c:\NewKEK\:executableADS.exe
       42  c:\NewKEK\:textads

    66090 bytes in 2 ADS listed

C:\>mkdir kek

C:\>copy NewKEK\* KEK
NewKEK\notepad.exe
       1 file(s) copied.

C:\>del NewKEK
C:\NewKEK\*, Are you sure (Y/N)? y
C:\>lads c:
```

The output of the command shows no alternate data streams.[9]

Attacks of the future

Megaworms combine features of 2002 most prolific worms and viruses. The megaworms will include the characteristics that were included in the Klez and Nimba worm. These worms used vulnerabilities that existed within Microsoft Internet Explorer and Outlook that allowed for automatic execution of malicious code. Nimba used the automatic execution of embedded MIME types within Microsoft Internet Explorer[10]. The Klez worm exploited vulnerability in Microsoft Outlook and Outlook Express. It attempted to execute itself, overwrite files and create hidden copies of the original.[11] Alternate data streams are one area that information can be hidden within an

NTFS file system without being detected with native tools. Some experts believe that the new era of megaworms will rely on multiple methods of propagations. Multiple methods of virus propagation are one way the virus threat is evolving.[12] The security dimensions that are in question with this new era are the confidentiality of the information and integrity of the information. The confidentiality of information is that malicious code could hide sensitive data that could be retrieved at a later date or emailed as an attachment without the user even knowing. Integrity of the files are in jeopardy because the file that the user sees from the system tools is not necessarily all of the data contained within the file.

"Nine of the top 10 viruses detected by all major virus-protection companies in 2002 were mass-mailing viruses that exploited known vulnerabilities in the Win32 application programming interface. And 87% of all reports of infections stemmed from Windows viruses."[13] A virus named Win2K.Stream was discovered in September of 2000. This virus attempted to conceal itself within an alternate data stream. McAfee discovered the virus. I believe as the alternate data stream within the windows operating system become better known that there will be an increase in writing malicious code in this direction. I worry more about code that is written to steal data in this manner. If a person cannot see important data that is stored within an alternate data stream of a valid file that is not important and that file is taken. Most people will not bat an eye that anything critical has been lost. This would threat is very real and another major player comes into play.

One of the major "defense in depth" tools used today is anti-virus software. This tool is relied on heavily within the security structure of organizations. Most anti-virus software cannot deal with alternate data streams easily if at all. Anti-virus experts believe that as long as the main stream virus can be detected then the anti-virus software can correct the virus. The problem with this logic is that an alternate data stream can be attached to a system file and the anti-virus software will report that it is safe. Again, the threat of the alternate data stream not being seen or recognized and then the possibility of that malicious code being executable is an upcoming major threat.[14]

I had noticed in older documents on the Internet that alternate data streams could not be executed. This is not true at least in a Windows XP environment. I was able to execute task manager that was kept in an alternate stream and store its output into an alternate stream. I then retrieved the output from the alternate stream. The commands are listed next.

```
C:\>type c:\windows\system32\tasklist.exe > c:\kek:taskhack.exe
C:\>start /B c:\kek:taskhack.exe > c:\kek:hackout.txt
Press the Enter key to return to a C:\> prompt

C:\>more < c:\kek:hackout.txt
```

```
Image Name                     PID Session Name        Session#    Mem Usage
================================================================================
System Idle Process              0 Console                    0         20 K
System                           4 Console                    0        216 K
smss.exe                       728 Console                    0        344 K
winlogon.exe                   800 Console                    0      2,240 K
SERVICES.EXE                   844 Console                    0      2,784 K
```

I deleted out most of the file but you should get the idea. Also, I used Windows XP system commands. I attached the alternate data streams to a directory. This is the output of the dir command.

```
C:\>dir c:\kek
 Volume in drive C has no label.
 Volume Serial Number is ACED-A288

 Directory of c:\kek

04/14/2003  10:17 AM    <DIR>          .
04/14/2003  10:17 AM    <DIR>          ..
               0 File(s)              0 bytes
               2 Dir(s)   1,392,217,600 bytes free
```

The directory command shows 0 bytes. This illustrates that alternate data streams can store information for later retrieval, contain program code that can be executed, and do not show up in a directory listing. This example shows that directories are just as vulnerable as files.

Alternate data streams are needed and should not be done away with. This process helps with interoperability with other operating systems such as Mac OS and its HFS. Applications are written to use this capability in proper ways. The threat that exists is the vulnerability that alternate data streams can be created without prior knowledge from the user. Two areas of the security fundamentals become subject. These are the confidentiality and integrity of the data. This puts the whole NTFS at a high risk.

I believe that the NTFS file structure because of the alternate data stream feature will be exploited in the coming years. To counter the threat that alternate data streams are providing, anti-virus software will need to do a better job not only detecting but protecting alternate data streams. The operating system needs tools written by the software vendor to not only show that alternate data streams are attached to files and directories but tools to be able to remove the alternate data stream without compromising file or directory data integrity and keeping the data confidentiality.

When the vendors provide the proper protection by modifying the anti-virus software and native tools that list and can manipulate the alternate data streams by adding, modifying, and removing, this will go a long way in mitigating the high risk that is associated with the NTFS. To date, this operating system structure has been extremely fortunate that not many malicious code attempts have been put into the wild to exploit the alternate data stream function. I believe that this will change as hackers become more resourceful in the future and write stealthier code.

[1] Microsoft TechNet "Chapter 18 - Choosing a File System", 10 April 2003 URL:http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/ntwrkstn/reskit/filesys.asp

[2] Microsoft TechNet "Chapter 17 - File Systems", 10 April 2003 URL:http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windows2000pro/reskit/part3/proch17.asp

[3] Microsoft TechNet "Chapter 17 - File Systems", 10 April 2003 URL:http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windows2000pro/reskit/part3/proch17.asp

[4] Microsoft TechNet "Chapter 18 - Choosing a File System", 10 April 2003 URL:http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/ntwrkstn/reskit/filesys.asp

[5] Microsoft TechNet "Indexing Service Adds Data Streams to Image Files", 11 April 2003 URL: http://support.microsoft.com/default.aspx?scid=kb;EN-US;q319300

[6] Security Administrator "Detecting Alternate Data Streams", 12 April 2003 URL: http://www.ntsecurity.net/Articles/Index.cfm?ArticleID=16189

[7] "FAQ: Alternate Data Streams in NTFS", 12 April 2003 URL: http://www.heysoft.de/nt/ntfs-ads.htm

[8] JSI FAQ "The Indexing Service adds data streams to image files?", 12 April 2003 URL:http://www.jsiinc.com/subj/tip4900/rh4961.htm

[9] "FAQ: Alternate Data Streams in NTFS", 12 April 2003 URL: http://www.heysoft.de/nt/ntfs-ads.htm

[10] CERT Advisory CA-2001-06 Automatic Execution of Embedded MIME Types, 12 April 2003 URL: http://www.cert.org/advisories/CA-2001-06.html

[11] Symantec W32.Klez.E@mm 12 April 2003 URL: http://securityresponse.symantec.com/avcenter/venc/data/w32.klez.e@mm.html

[12] Dan Verton, "Viruses get Smarter" 12 April 2003 URL: http://www.computerworld.com/securitytopics/security/story/0%2C10801%2C77794%2C00.html

[13] Dan Verton, “Viruses get Smarter” 12 April 2003 URL: http://www.computerworld.com/securitytopics/security/story/0%2C10801%2C77794%2C00.html

[14] Security Administrator “NTFS Alternate Data Streams Valuable Win2K feature or dangerous bug?”, 13 April 2003 URL: http://www.secadministrator.com/Articles/Index.cfm?ArticleID=19878