Alternative Data Streams in NTFS – A Gateway for Subverting Endpoint Security Systems

Nenad Stojanovski †,

Institute of Informatics, Faculty of Sciences,
St. Cyril and Methodious University,
Skopje, Republic of Macedonia,

and

Danilo Gligoroski †† **Svein J. Knapskog** ††

Centre for Quantifiable Quality of Service in Communication Systems,
Norwegian University of Science and Technology, Trondheim, Norway

Summary
In this paper we use "alternative data streams" that were introduced with the occurrence of Windows NT and its file system NTFS, as a gateway for subverting several commercial endpoint security systems. We give a simple set of commands by which it is possible to copy data from PCs that have installed endpoint security access systems, with- out been detected or by making the endpoint security access system to generate incomplete log entries that again do not reveal an information about the files that have been copied.

Key words:
NTFS, Alternative Data Streams, Subversion, Endpoint Security Systems

1. Introduction

Data theft and confidential information leakage is one of the biggest security issues that today's corporations face. The main source of corporate data theft and sensitive information leakage has always been from the inside of the corporations.

Regarding to all types of protection that are implemented in the corporate LANs, PC and laptops are the weakest points in the whole security chain. Because of this, they are the ones that need extra protection measures. The protection measures that are used to protect them are not always because of an external threat, but also because of the inside threat of data theft.

A typical example is the stealing of 20 million VISA and 13.9 million MasterCard accounts that was discovered in June 2005 [1]. From the reports, the stealing happened in CardSystems Solutions of Tucson, Ariz. – a partner company of VISA and MasterCard that processed yearly more than $15 billion in payments for small to midsize merchants and financial institutions. An infiltrator (acting as insider) had managed to place unnoticed a computer code or script on the CardSystems network that made it possible to extract information.

In U.S. the legislative part of the increasing public awareness for more strict rules and standards, that will force financial companies to implement in-depth information security standards, started with adoption of Gramm-Leach-Bliley Act in 1999 [2]. It went into effect on July 1, 2001.

Then, in 2002 U.S. congress adopted the Federal Information Security Management Act [3] – an act that addresses information security issues in governmental institutions but its information security recommendations can be applied also on other private companies and corporations.

A special role for specifying all levels of security policies for both acts was given to the National Institute of Standards and Technology (NIST) who published a Computer Security Handbook [4]. There, a special part is devoted to implementing special controls that will prevent data from stealing inside the organizations.

Manuscript received December 5, 2007
Manuscript revised December 20, 2007

According to this legislation, in order to ensure security of their corporate (organizational) PCs and laptops and prevent inside data theft and information leakage, corporations and governmental institutions started to implement so called endpoint security access systems.

Commercial applications for implementing the endpoint security have been on the market for more than seven years. During that period, generally they have matured and adapted to the new operating systems, new types of mediums for data storage, and new types of networks. However, there are still some basic ways (or unintentional backdoors), deeply hidden in the properties of the operating systems, that enable overriding some of the protection measures against data theft. Actually, in this paper we give concrete simple set of commands that subvert the installed systems for endpoint security measures. Our work can be seen as a "proof of concept" how those systems in some situations fail to protect against data theft, or how they are not able to detect the programs that are brought from the outside and that can furthermore perform malicious activities.

2. Endpoint Security Systems

Endpoint security access systems are designed to protect against unwanted information leaking from within the corporate network. They are represented as the cure for all of the unwanted information leakage incidents. They can protect the affected systems in several ways that depends on their setup and configuration. The whole protection is done by controlling, monitoring and logging the ways how the information is copied to or from the portable media devices that can be attached to the system. The following hardware can be defined under the terms of portable media devices:

- USB memory devices
- FireWire memory devices
- Wireless data connections
- Optical media

As previously mentioned, endpoint security systems are there to protect the systems from unwanted information copying. In order to achieve that, an agent has to be installed on the systems that can be the source of the information leakage. Endpoint security systems are centrally managed systems, which implies that the agents are centrally controlled from some kind of management software.

Since it is hard to predict the behavior and needs of different target groups, agents act by following some predefined policy. The policy is defined on the management software and through it it is deployed on the agents. In this way it is possible to define policies for different groups of agents in order to satisfy different needs of the target groups. The policies can allow or deny the users from connecting to the portable media devices, or they can allow or deny read and write operations. This means that the agents intercept the copy operations and execute them or discard them depending on the defined policy. On the other hand, the agents are also used to collect data that come from portable devices.

The logged/collected data is sent to the management server periodically where it is stored into a database. Through the management software the security analyst is able to analyze the collected data. Log data contains detailed information such as device events or file audits on portable devices. For example a typical file copy operation is logged as Read and Write operation. Endpoint security systems are produced by dozens of vendors, but if we look at how they are functioning, we can realize that they generally use the agent/policy concept and that they all protect the systems in a similar way Data theft and confidential information leakage is one of the biggest security issues that today's corporations face. The main source of corporate data theft and sensitive information leakage has always been from the inside of the corporations.

3. Subverting Endpoint Security Systems

3.1 Alternative Data Streams

Alternative data streams were introduced with the new file system NTFS that Windows NT brought with it [5]. They were created to provide compatibility with the Macintosh's old file system HFS [6]. HFS functioned in a way that it used both resource and data forks to store content. The data fork was used to store the entire file content and the resource fork was used to store information that were used to identify the

file type as well as some other information concerning the file.

As far as we know, in 2004, Windows-Security.com published the first article describing the security threat of alternate data streams [7]. However, it seems that it did not attracted much attention, especially (as we will show further in the paper) not an attention by the designers of Endpoint Security Systems.

So, nowadays this relatively unknown Windows (or NTFS) compatibility feature can be still used to hide malicious software like rootkits or backdoors. It can be used to fork file data into existing files without affecting their functionality, size, or their occurrence in the reports of the traditional file browsing commands/utilities like "dir" or "Windows Explorer". What needs to be mentioned is that alternative data streams are not viewable using the usual Microsoft Windows tools like "Windows Explorer".

When it comes to the practical point of view, alternative data streams are easily created. They are created only with the use of common Windows tools that are present by default in Microsoft Windows. The only prerequisite that needs to be fulfilled is that the alternative data stream is created on an NTFS partition. By using alternative data stream an attacker is able to hide whole applications in the alternative data stream. As shown in Figure 1, a simple set of commands can be used to create an alternative data stream.

Next, on Figure 2 we illustrate how it is possible to put a whole application into an alternative data stream and the easiness of executing that application.

```
C:\>type c:\WINDOWS\NOTEPAD.EXE > out1.txt:note.exe
C:\>_
```

Fig. 1. Creation of an alternative data stream

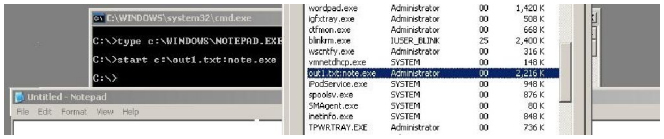

Fig. 2. Execution of executables hidden in alternative data streams

Unfortunately, alternative data streams cannot be disabled and there are no read/write/execute restrictions on files on which the user already has write permissions. The only way to detect them is to use some third party utilities that can scan the partition and can give notifications about the existence of alternative data streams. Some third party applications also monitor the integrity of the file system with the use of file checksums and raise alerts if an alternative data stream is present or created.

3.2 Subverting Endpoint Security Systems

Previously we mentioned that endpoint security systems are used to prevent or to log data that is copied from PCs that are used in the corporate LAN network. For the purpose of our research we created a test environment where we tested several of endpoint security systems. Due to the specific agreements with the vendors of these endpoint security systems (that were kind to give them to us for testing purposes), we are unable to reveal their concrete names and in the figures in this paper (screen-dumps) we have deleted or blurred all information about the names of the products or the names of the vendors. However, we claim that all other scientific/security attacks and the results from those attacks are performed on real commercial endpoint security systems.

The created test environment defined a security policy that was centrally managed and distributed to the test PCs. According to the defined policy, we allowed read and write operations to all portable memory devices, but the logging policy was established for all the operations towards the portable devices in order to monitor the execution of operations. Since all of the tested commercial endpoint security systems were agent based, an agent was installed on the test PCs. We want to stress that all the policies were created based on real world scenarios for the corporate LAN networks where at least portable USB memory devices are allowed for corporate use.

During the testing phase we discovered that all of the tested products suffer from a weakness that allows the users to copy the files undetected or to falsely document/log the copy activity. The above mentioned anomaly is due to the existence of alternative data

streams which, as stated before, is not very well known Windows (NTFS) compatibility feature.

In the first test we created a file on one of the local partitions on the internal hard drive. After the creation of the file, an alternative data stream was created and it was attached to the previously created file. The alternative data stream contained a message which we treated as confidential information. The file that was created first was copied to the external USB hard drive that was NTFS formatted, which is illustrated on Figure 3.

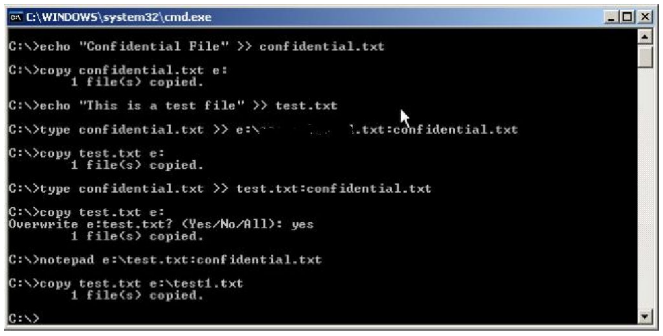

Fig 3. Creation of an alternative data stream

After analyzing the logs we discovered interesting results that concern the alternative data streams. To be more precise, on the management station the whole copy operation (the read/write operation) was stored as partial information into the database. This means that the alternative data stream was detected, but reported as the normal file. An illustration of this is shown on Figure 4.

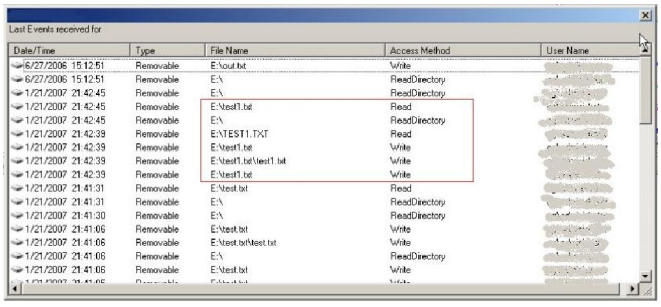

Fig. 4. Logs collected by the agent

In the second test we created a file on the external USB hard drive and added an alternative data stream to it. The alternative data stream that we added was actually the Windows application "notepad.exe". After the preparation of the external USB hard drive, we connected it to one of the PCs where we had an agent installed. This test was done in order to simulate how some potentially malicious software can be secretly loaded into the corporate network. The test file was copied to one of the local partitions on the PC where we had an endpoint security agent. The action of copying is illustrated on Figure 5.

Fig. 5. Copy operation from an external drive to the local drive

Again, we had interesting outcome of the test. The logs didn't show any kind of information that concerned "notepad.exe" and we were able to execute it from local partition. Logs are illustrated on Figure 6.

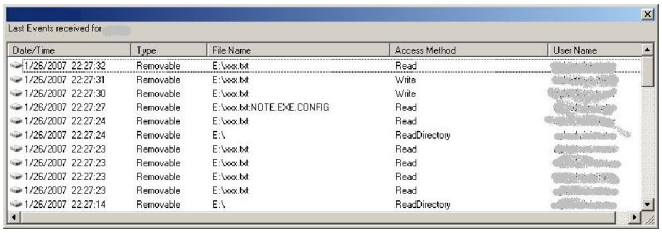

Fig. 6. Logs collected by the agent

4. Conclusion

In this paper we showed how it is possible to bypass the logging functionality that some endpoint security systems provide with their agents. By doing this, we showed how it is possible to steal data from the corporate network or how to put an executable code to the corporate network unnoticed by the protection of the endpoint security systems.

However, we want to express our opinion that endpoint security systems are a good step towards

creating in-depth security in corporate LAN networks. In order to solve problems with subverting the logging functionality we suggest endpoint security systems to add some extra checks for the alternative data streams, because alternative data streams are here to stay for some time.

As for the overall in-depth security concept, at this moment endpoint security systems are not enough to protect the sensitive corporate data. So, some other complementary security measures should also be used to ensure security of the corporate data.

References

[1] E. Dash and T. Zeller Jr.: "MasterCard Says 40 Million Files Put at Risk", New York Times article, June 18, 2005, http://www.nytimes.com/2005/06/18/business/18cards.html?ex=1276747200\&en=519cdb9337e3a3d3\&ei=5088

[2] Conference Report and Text of Gramm-Leach-Bliley Bill, Senate Banking Committee, http://banking.senate.gov/conf/confrpt.htm

[3] Federal Information Security Management Act of 2002, http://csrc.nist.gov/drivers/documents/FISMA-final.pdf

[4] NIST Special Pub 800-12, An Introduction to Computer Security: The NIST Handbook, http://csrc.nist.gov/publications/nistpubs/800-12/

[5] Microsoft Corporation, How To Use NTFS Alternate Data Streams, http://support.microsoft.com/kb/105763

[6] Wikipedia, Hierarchical File System, http://en.wikipedia.org/wiki/Hierarchical_File_System

[7] R. Zadjmool, "Hidden Threat: Alternate Data Streams", WindowSecurity.com, 2004, http://www.windowsecurity.com/articles/Alternate_Data_Streams.html

Nenad Stojanovski is due to receive his MS degree in Computer Science from Institute of Informatics, Faculty of Natural Sciences and Mathematics, at University of Skopje – Macedonia in November 2007. His research interests are network security, software security, mobile phone security and software engineering.

Danilo Gligoroski received his PhD degree in Computer Science from Institute of Informatics, Faculty of Natural Sciences and Mathematics, at University of Skopje – Macedonia in 1997. His research interests are Cryptography, Computer Security, Discrete algorithms and Information Theory and Coding. Currently he is PostDoc at Q2S – Centre for Quantifiable Quality of Service in Communication Systems at Norwegian University of Science and Technology - Trondheim, Norway.

Prof. Svein Johan Knapskog is the head of the "Centre for Quantifiable Quality of Service in Communication Systems – Q2S", at the Norwegian University of Science and Technology, Trondheim, Norway. His research interests are Network Security, Cryptography, and Security Standards.