Utilizing Slack Space for Hidden Partitions

by

Chuck Easttom

Abstract

Ghost or phantom partitions have been used for some time to hide data. Often they are used to store recovery information in a manner not easily discoverable or altered by the end user. However traditional forensics techniques can easily discover these phantom partitions. This paper outlines a methodology for using the slack space in the primary file system to create an entire secondary phantom file system. This would create a phantom partition that is interwoven with the primary file system and not readily detectable by conventional forensics methodologies.

A ghost partition is a partition that is hidden from normal view of the operating system (Dickson, Harsany, Arnold, & Marler, 1999). These are often used to store sensitive information, to make it more difficult to locate that information. However it is usually possible to detect a ghost partition by comparing the size of the hard drive according to the manufacturer's specifications to the size of the hard drive reported by the operating system. A significant difference would indicate the presence of a ghost partition. Often these phantom partitions are accessible via a small executable on the public partition (Breeden, 2002). That executable is usually password protected.

What is being suggested in this paper is a new approach to phantom partitions. Rather than having a normal partition that simply is not displayed to the user, a truly hidden partition is created. One that would not be readily detected by comparing the hard drive size reported by the operating system to the hard drive size designated by the hard drive manufacturer.

In this method the first step is to create a secondary file systems file table. In FAT and FAT32 the file table is the File Allocation Table. In NTFS this is the Master File Table (MFT). The secondary file table would simply note slack space available in the main file system. Gregg, Varsalone, and Wright (2007) describe slack space as follows:

> "Because clusters are a fixed size, the data stored in a cluster will use the entire space, regardless of whether it needs the entire cluster. For example, if you allocated a cluster size of 4,096 bytes and saved a 10-byte file to the disk, the entire 4KB cluster would be used even though 4,086 bytes of space is wasted. This wasted space is called slack space or file slack. " (p. 93).

Then data would be saved in the slack space of clusters. Slack space has been effectively used on a small scale to hide individual elements of data (Gregg, Varsalone, & Wright, 2007). The use of slack space to store data is shown in figure 1-1.

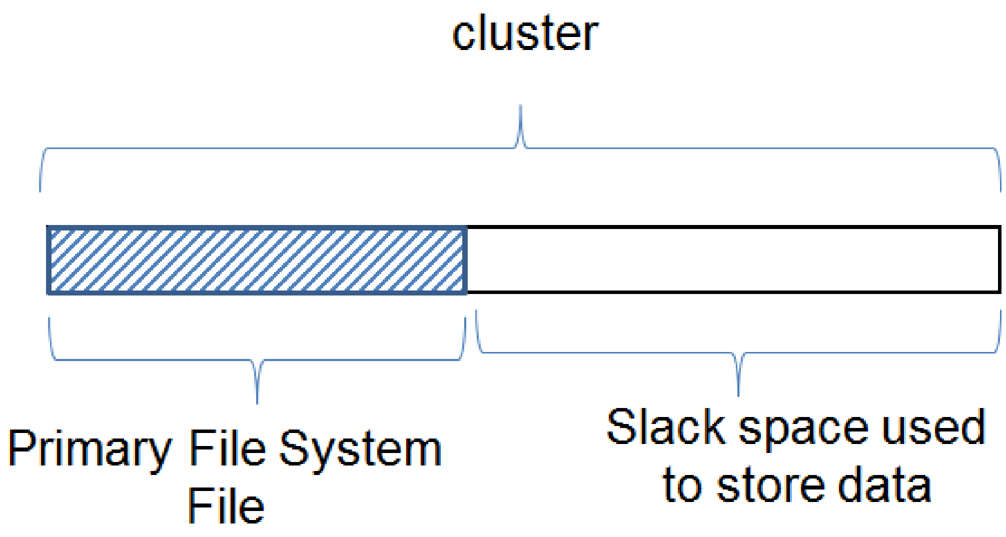
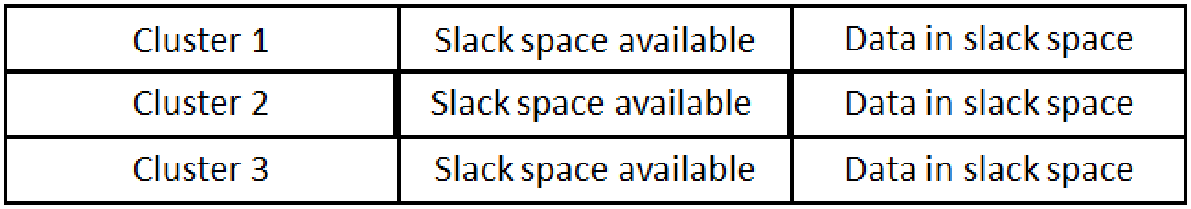


Figure 1-1 Slack space used to store data

What is proposed in this paper is a large scale phantom file system that stores data in the slack space left by the primary file system. This would lead to a secondary file system that was undetectable by conventional forensics methodologies. That secondary file system would record the slack space available by cluster. This is shown in figure 1-2.



Cluster 1	Slack space available	Data in slack space
Cluster 2	Slack space available	Data in slack space
Cluster 3	Slack space available	Data in slack space

Figure 1-2 Phantom File System

Access to this phantom file system could be accommodated in many ways. The most obvious modality would be to utilize the method frequently employed with traditional phantom partitions. That would be an executable on the primary file system, that provides access to this file system. By launching this executable and entering the correct password, the user would be given access to the phantom file system. One method for accomplish this would be for the executable to launch a shell or command window that utilized the phantom file system rather than the primary file system.

The applications for this concept are numerous. The hiding of data is a common way to secure it. Technologies like steganography use the least significant bits of an image, sound file, or video file in order to hide data (Artz, 2001). By having a secondary file system that was completely hidden the user could hide sensitive data in such a way that an attacker would not even be aware that the data was present. It would also protect that data from traditional spyware.

Security could be further enhanced by also utilizing steganography and/or encryption with the data that is stored on the secondary file system. This means that one could store data that was encrypted and/or hidden via steganography on the phantom file system. This would make locating that data very difficult. It would also be possible to essentially encrypt the entire phantom partition. This would lead to a phantom drive that was encrypted, making the data on it virtually impossible to retrieve without the appropriate passwords.

Clearly there are a number of applications and possible implementations of this technology. The goal of this paper is simply to introduce the reader to the concept.

References

Artz, D. (2001). Digital steganography: hiding data within data. *Internet Computing*, *IEEE*, *5* (3). Retrieved from http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?&arnumber=935180

Breeden, J. (2002). *Phantom makes data invisible*. Retrieved from http://gcn.com/articles/2002/11/01/phantom-makes-data-invisible.aspx.

Dickson, C, Harsany, S., Arnold, M., Marler, A. (1999). Ghost Partition. United States Patent 5,974,567

Gregg, M, Varsalone, J., Wright, C (2007). *The Official CHFI Exam Study Guide*. Burlington, MA: Syngress Press