Ambiguous File System Partitions

By:
Janine Schneider (Friedrich-Alexander-Universität Erlangen-Nürnberg), Maximilian Eichhorn (Friedrich-Alexander-Universität Erlangen-Nürnberg), and Felix Freiling (Friedrich-Alexander-Universität Erlangen-Nürnberg)

From the proceedings of

The Digital Forensic Research Conference

DFRWS USA 2022

July 11-14, 2022

Ambiguous File System Partitions

Janine Schneider, Maximilian Eichhorn and Felix Freiling

July 13, 2022

IT Security Infrastructures Lab
Department of Computer Science
Friedrich-Alexander University Erlangen-Nürnberg (FAU)

Introduction

File System

Partition

Storage

Left: Western Digital WD2500JD-00HBC0 20051228, 2007, https://commons.wikimedia.org/wiki/File:Western_Digital_WD2500JD-00HBC0_20051228.jpg, https://commons.wikimedia.org/wiki/Commons:GNU_Free_Documentation_License,_version_1.2

Ambiguous File System Partitions

- Incorporate a fully functional file system within another file system such that both file systems can be used
- Partitions in which two fully functional file systems co-exist and for which it is not totally clear which file system is being primarily used

Contribution

- We examine several different file systems regarding their potential for creating ambiguous file system partitions
- We develop a taxonomy of ambiguous file system partitions, which can be used to classify different combination approaches and evaluate their quality
- We created four ambiguous file system partition examples
- We show that the apparently simple forensic analysis of file systems can be non-trivial and that forensic tools are not prepared for ambiguous file system partitions

What for?

- Forensic tools rely on the correctness and existence of data structures
- Many forensic tools assume that basic data structures are trustworthy
- This must not necessary be the case

- Forensic tools are a critical point in any investigation, and they have to fulfill the highest quality criteria (Wundram et al.)
- Therefore, advanced forensic tool testing is very important
- Find limitations of current forensic tools

Taxonomy

Taxonomy

Placement:

a) Overlapping: starting sector x < starting sector y, starting sector y < ending sector x and ending sector x < ending sector y

b) Subset: starting sector x < starting sector y < ending sector y < ending sector x

c) Identical: starting sector x = starting sector y and ending sector x = ending sector y

Taxonomy

Stability:

a) Mutual disrespect: file system x does not respect y and file system y does not respect x

b) Directed respect: file system x respects y but y does not respect x

c) Mutual respect: file system x respects y and y respects x

Taxonomy

Accessibility:

a) Unmodified: both file systems do not need special parameters to be found and used

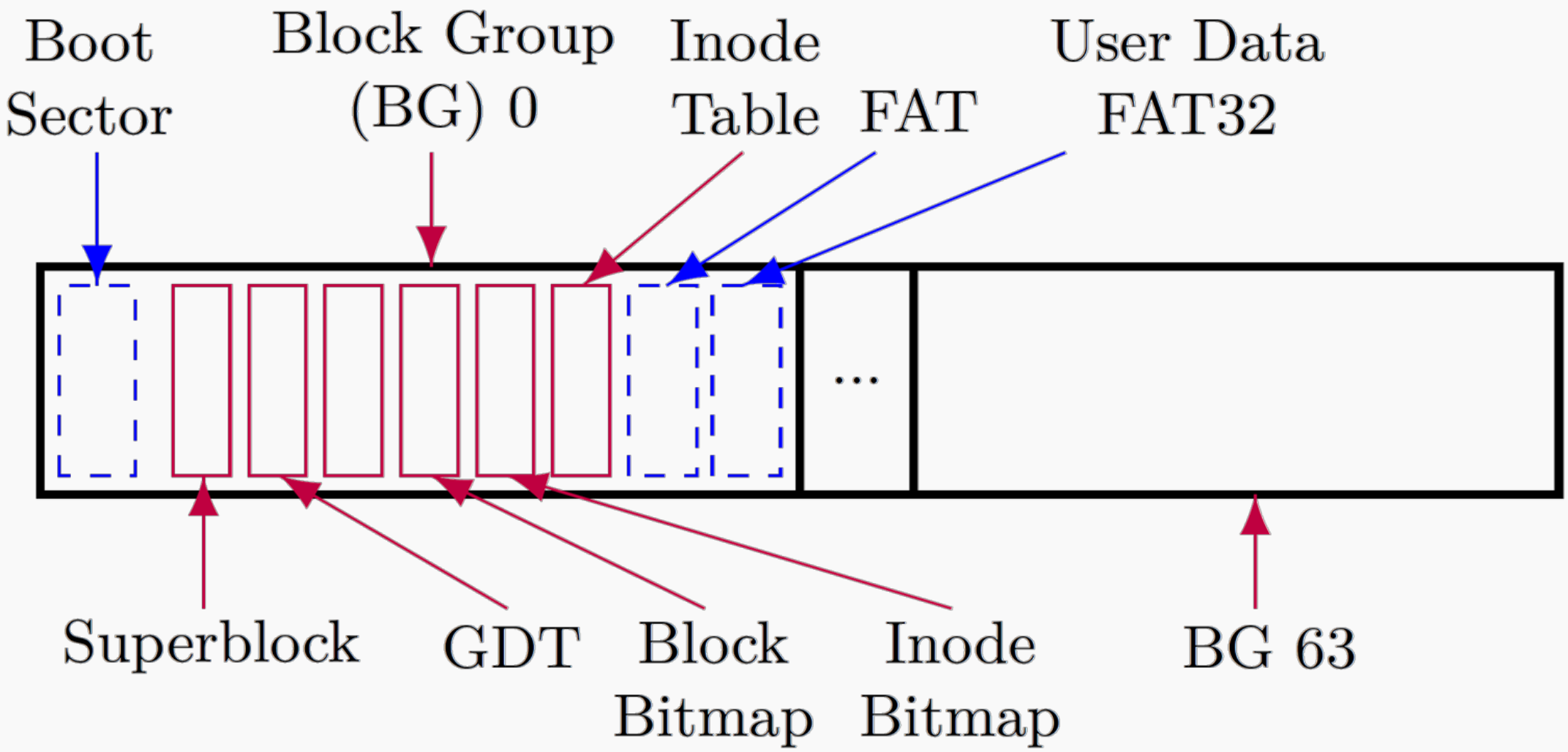
b) Modified: at least one file system needs special knowledge from the outside to be found and use.

Ambiguous File System Partition Examples

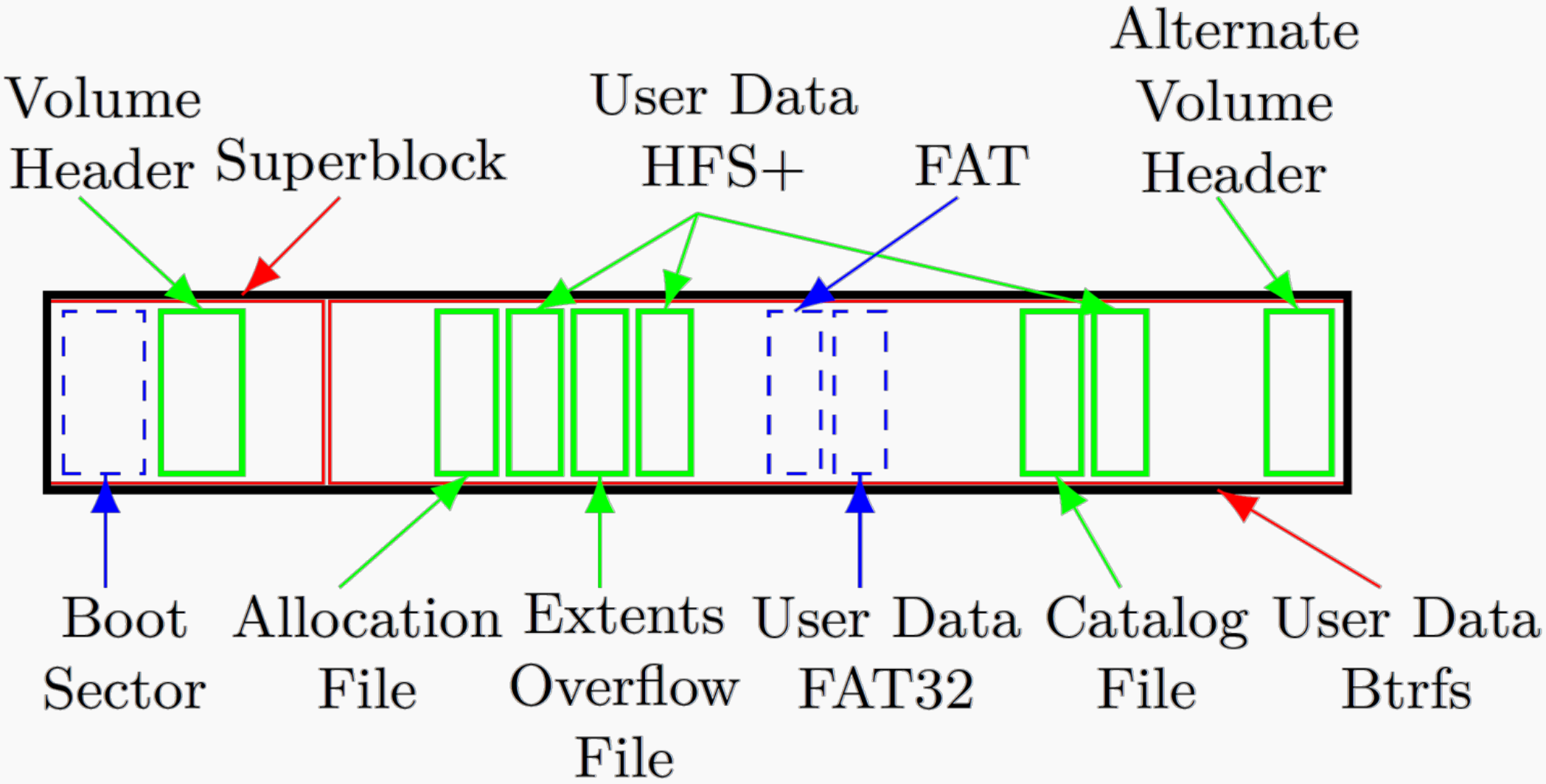
Examples

Host file system	Guest file system(s)	Placement	Stability	Accessibility
NTFS	FAT32	partially identical	mutual respect	modified
Ext3	FAT32	overlapping	mutual respect	unmodified
HFS+	FAT32	overlapping	mutual respect	unmodified
Btrfs	HFS+, FAT32	overlapping	directed respect	unmodified

Ext3 & FAT32


Boot Sector
Block Group (BG) 0
Inode Table
FAT
User Data FAT32
...
Superblock
GDT
Block Bitmap
Inode Bitmap
BG 63

Btrfs & HFS+ & FAT32



Evaluation

Evaluation

Tool	Disk image	Recognized FS	Files found	Notes
TestDisk	Ext3 & FAT32	Ext3	n/a	Indication of FAT and damaged fragments
	HFS+ & FAT32	HFS+	n/a	HFS+ not supported, Indication of FAT
	Btrfs & HFS+ & FAT32	HFS+	n/a	Indication of Btrfs and FAT
The Sleuth Kit	Ext3 & FAT32	ExtX or FAT	Depending on selected file system	Does not commit to a file system
	HFS+ & FAT32	HFS or FAT	Depending on selected file system	Does not commit to a file system
	Btrfs & HFS+ & FAT32	HFS or FAT	Depending on selected file system	Btrfs not supported, Does not commit to a file system
Autopsy	Ext3 & FAT32	ExtX or FAT	Depending on selected file system	Does not commit to a file system
	HFS+ & FAT32	HFS or FAT	Depending on selected file system	Does not commit to a file system
	Btrfs & HFS+ & FAT32	HFS or FAT	Depending on selected file system	Does not commit to a file system
X-Ways Forensics	Ext3 & FAT32	Ext3	Only host files	
	HFS+ & FAT32	HFS+	Only host files	
	Btrfs & HFS+ & FAT32	HFS+	Only host files	
Magnet AXIOM	Ext3 & FAT32	Ext3	Only host files	
	HFS+ & FAT32	HFS+	Only host files	
	Btrfs & HFS+ & FAT32	HFS+	Only host files	

Recognizing Ambiguous File System Partitions

Recognizing Ambiguous File System Partitions

- Check data structures for irregularities (deviations from what can be considered as ordinary)
- Check the position and size of certain data structures
- Check number of damaged sectors or blocks
- Looking for unusual file system labels
- Check “empty” space
- Check size of allocated areas

Conclusion and Future Work

Conclusion

- Ambiguous file system partitions remind us that there is no such thing as a "clear file system interpretation"
- Forensic tools should clearly communicate ambiguities to the user

Future Work

- Other file systems combinations
- Achieved more difficult combination (identical placement, mutual respect and unmodified accessibility)
- Automatically create ambiguous file system partition examples
- Automated storage assignment to allow seamless use of ambiguous file system partitions

Thank you!