Guide to Computer Forensics and Investigations Fourth Edition

Chapter 8

Macintosh and Linux Boot Processes and File Systems

Objectives

- Explain Macintosh file structures and the boot process
- Explain UNIX and Linux disk structures and boot processes
- Describe other disk structures

Understanding the Macintosh File Structure and Boot Process

- Mac OS X version 10.4
  - Darwin core
  - **BSD UNIX** application layer
- **Hierarchical File System (HFS)**
  - Files stored in nested directories (folders)
- **Extended Format File System (HFS+)**
  - Introduced with Mac OS 8.1
  - Supports smaller file sizes on larger volumes, resulting in more efficient disk use

Understanding the Macintosh File Structure and Boot Process (continued)

- **File Manager** utility
  - Reading, writing, and storing data to physical media
- **Finder**
  - Keeps track of files and maintain users' desktops
- In older Mac OSs, a file consists of two parts:
  - **Data fork** and **resource fork**
  - Stores file metadata and application information

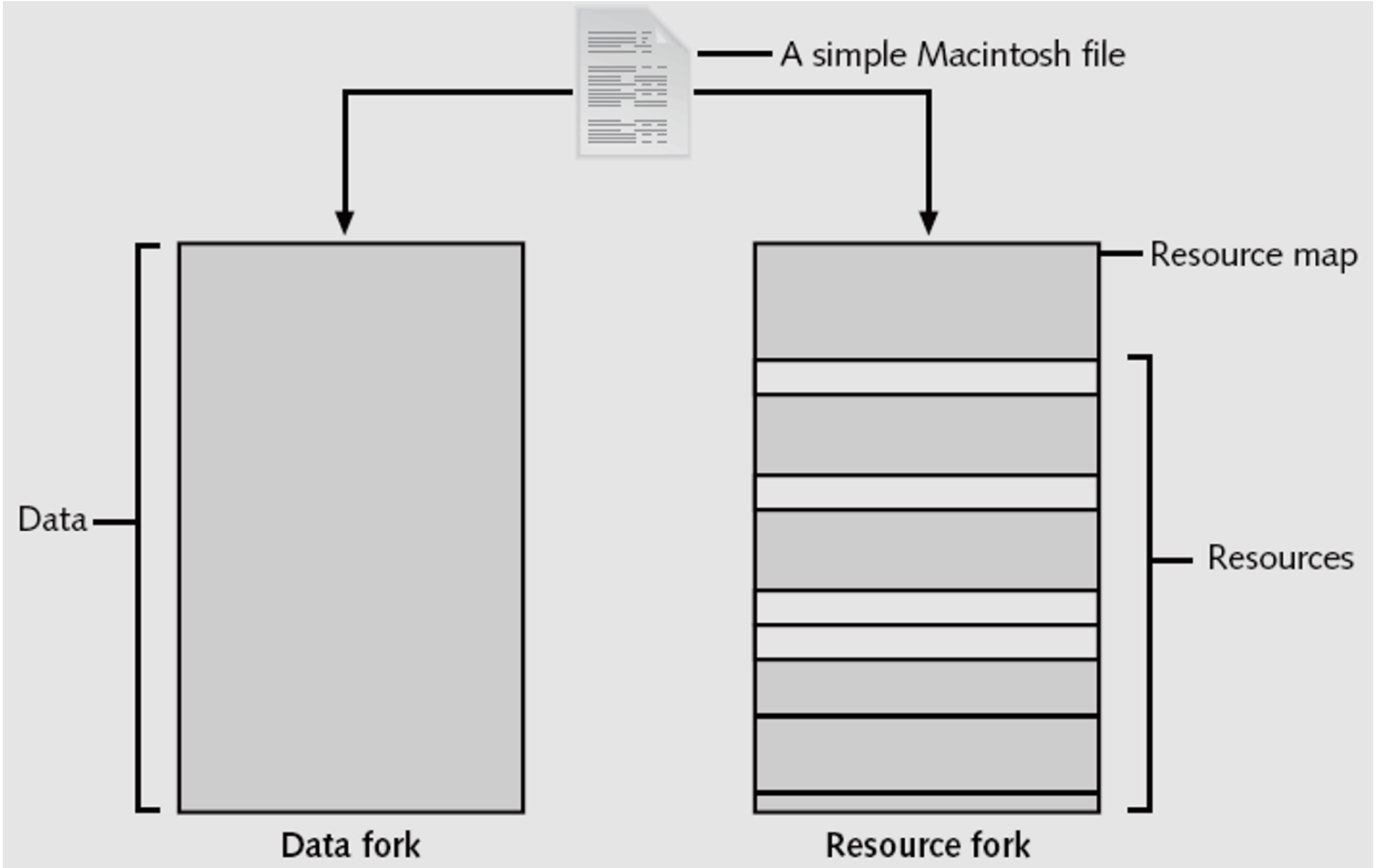

Figure 8-1 The resource fork and data fork in a Mac OS file

Understanding Macintosh OS 9 Volumes

- A volume is any storage medium used to store files
  - Can be all or part of a hard disk
  - On a floppy disk is always the entire disk
- **Allocation** and **logical blocks**
  - Logical blocks cannot exceed 512 bytes
  - Allocation blocks are a set of consecutive logical blocks

Understanding Macintosh OS 9 Volumes (continued)

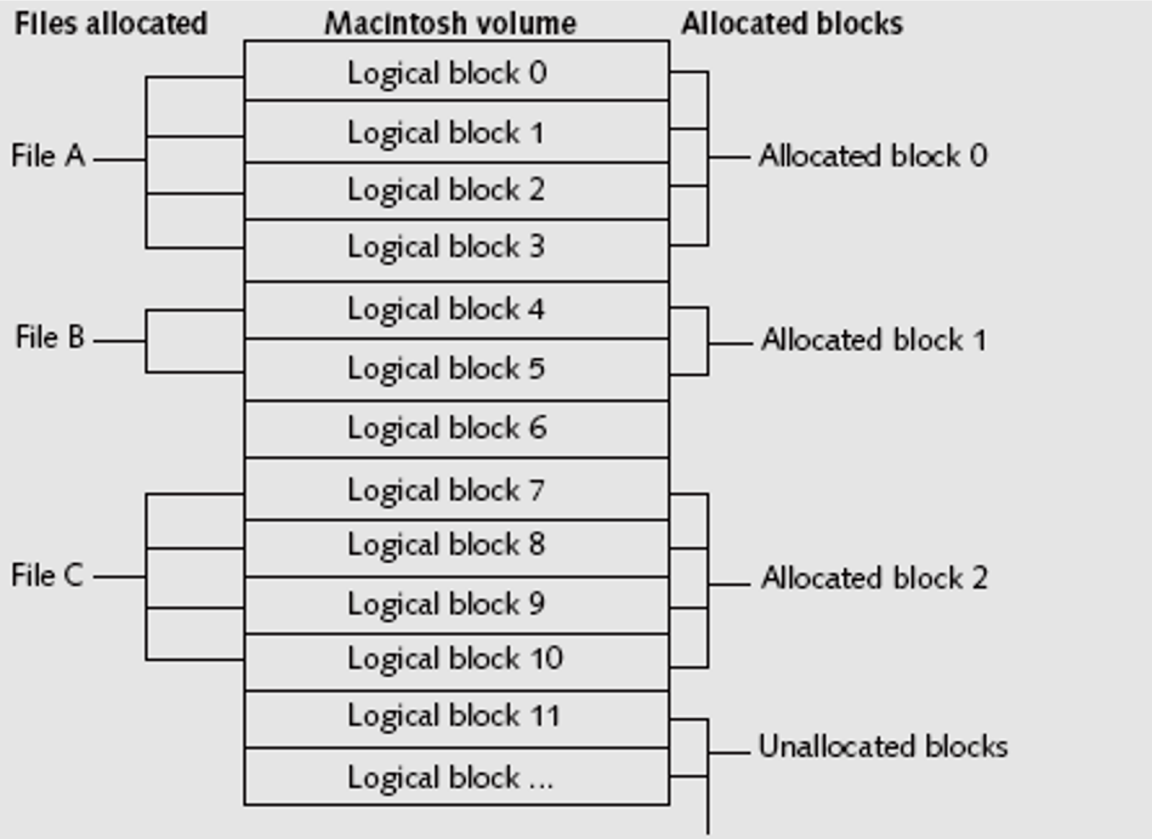

Figure 8-2 Logical and allocation block structures

Understanding Macintosh OS 9 Volumes (continued)

- Two EOF descriptors
  - **Logical EOF**
    - Actual size of the file
  - **Physical EOF**
    - The number of allocation blocks for that file
- **Clumps**
  - Groups of contiguous allocation blocks
  - Reduce fragmentation

Understanding Macintosh OS 9 Volumes (continued)

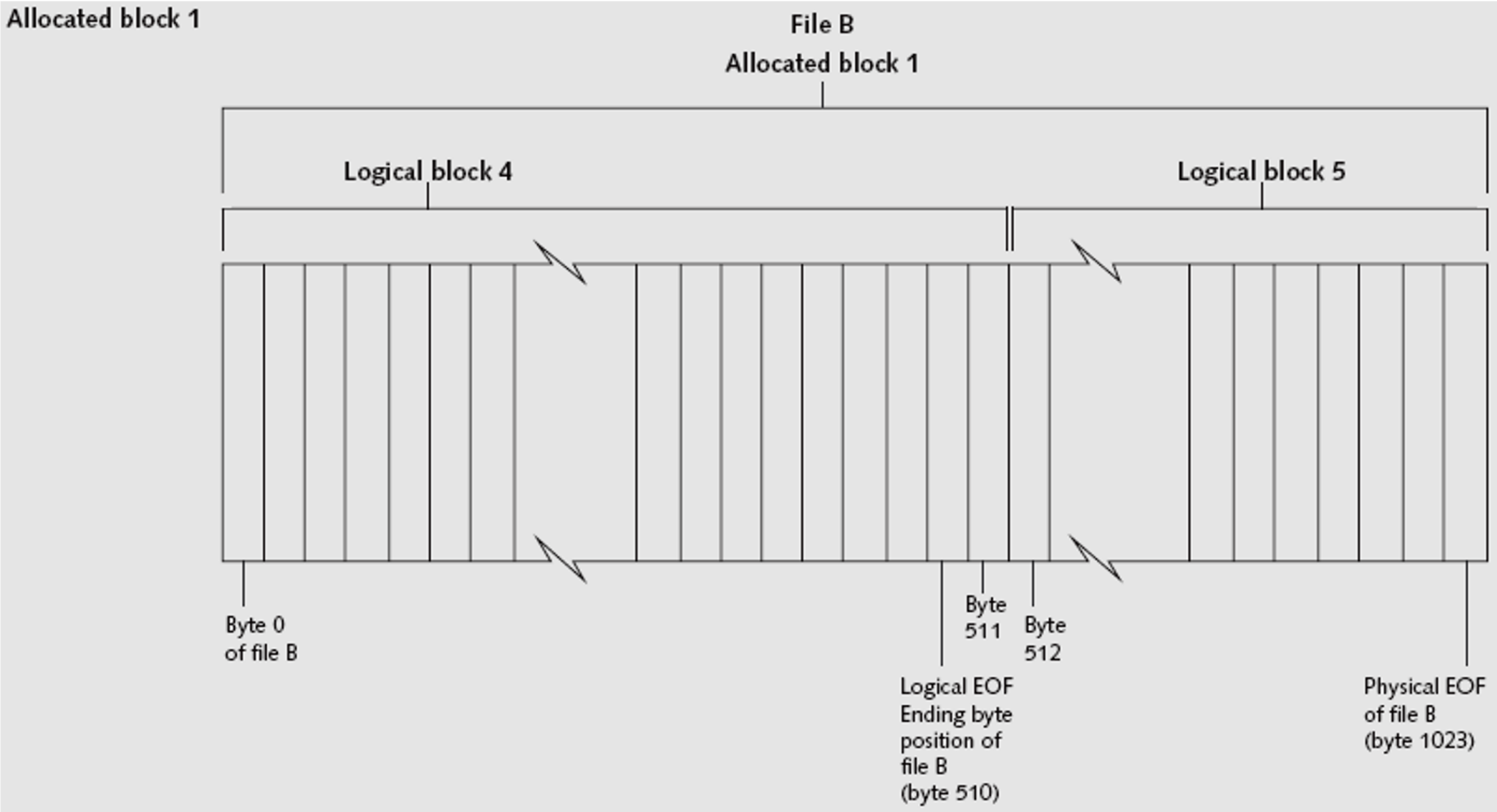

Figure 8-3 Logical EOF and physical EOF

Exploring Macintosh Boot Tasks

- Use Open Firmware
  - Processor- and system-independent firmware
  - Controls microprocessor after hardware initialization
- The boot process for OS 9 is as follows:
  - 1. Power on the computer
  - 2. Hardware self-test and Open Firmware run
  - 3. Macintosh OS starts
  - 4. The startup disk is located
  - 5. System files are opened

Exploring Macintosh Boot Tasks (continued)

- The boot process for OS 9 (continued):
  - 6. System extensions are loaded
  - 7. OS 9 Finder starts
- Tables 8-1 and 8-2 are an overview of how HFS and HFS+ system files handle data

Table 8-1 HFS system files

HFS block position	HFS structure	Purpose of structure
0 1	Boot block	Startup volume containing boot instructions. Also stores system files and Finder information.
2	Master Directory Block (MDB)	Contains volume creation date and time and location of other system files, such as Volume Bitmap. A duplicate of this file called the Alternate MDB is located at the second-to-last block on the volume. Its purpose is to provide information to the OS disk utilities.
3	Volume Bitmap	Tracks used and unused blocks on the volume.
	Catalog	Lists all files and directories on the volume. It's a B*-tree file that uses the extents overflow file to coordinate all file allocations to the volume.
	Extents overflow file	Lists the extra extents, which are the allocated blocks used to store data files. It's a B*-tree file.

Table 8-2 HFS+ system files

HFS+ byte offset (fixed starting position)	HFS+ structure	Purpose of structure
0	Boot blocks	No change from HFS.
1024	Volume Information Block (VIB)	Replaces the MDB used in HFS.
Not fixed	Allocation file	Tracks available free blocks on the volume; replaces the HFS Volume Bitmap.
Not fixed	Extents overflow file	For files with more than eight extents, additional extents are recorded and managed through this B*-tree system file.
Not fixed	Catalog	Similar to an HFS catalog, this improved version allows up to eight extents for each file's forks. It's a B*-tree file.
Not fixed	Attributes file	Stores new file attribute information that isn't available in HFS. The new attributes are inline data attribute records, fork data attribute records, and extension attribute records.
Not fixed	Startup file	New to HFS+, this file can boot non-HFS and HFS+ volumes.
Not fixed	Alternate VIB	Same file as the HFS Alternate MDB.
	Reserved (512 bytes)	Last sector of the volume; used by Apple during manufacturing.

Exploring Macintosh Boot Tasks (continued)

- Older Macintosh OSs use
  - First two logical blocks as boot blocks
  - **Master Directory Block (MDB)** or **Volume Information Block (VIB)**
    - Stores all information about a volume
  - **Volume Control Block (VCB)**
    - Stores information from the MDB when OS mounts
- **Extents overflow file**
  - Stores any file information not in the MDB or a VCB

Exploring Macintosh Boot Tasks (continued)

- **Catalog**
  - Listing of all files and directories on the volume
  - Maintains relationships between files and directories
- Volume Bitmap
  - Tracks used and unused blocks on a volume
- Mac OS 9 uses the **B*-tree** file system for File Manager
  - Actual file data is stored on the **leaf nodes**
  - B*-tree also uses header, index, and map nodes

Using Macintosh Forensic Software

- Tools and vendors
  - BlackBag Technologies
  - SubRosaSoft MacForensicsLab
  - Guidance EnCase
  - X-Ways Forensics
  - ProDiscover Forensic Edition
  - Sleuth Kit and Autopsy

Using Macintosh Forensic Software (continued)

- Macintosh Acquisition Methods
  - Make an image of the drive
  - Static acquisition of the suspect drive is preferable to a live acquisition
  - Removing the drive from a Macintosh Mini's CPU case is difficult
    - Attempting to do so without Apple factory training could damage the computer
  - Use a Macintosh-compatible forensic boot CD to make an image

Using Macintosh Forensic Software (continued)

- Macintosh Acquisition Methods (continued)
  - BlackBag Technologies sells acquisition products specifically designed for OS 9 and earlier
    - As well as OS X
  - MacQuisition is a forensic boot CD that makes an image of a Macintosh drive
  - After making an acquisition, examine the image of the file system
    - The tool you use depends on the image file format

Using Macintosh Forensic Software (continued)

- Macintosh Acquisition Methods (continued)
  - BlackBag Technologies Macintosh Forensic Software and SubRosaSoft MacForensicsLab
    - Can disable/enable **Disk Arbitration**
  - Being able to turn off the mount function in OS X
    - Allows you to connect a suspect drive to a Macintosh without a write-blocking device

Using Macintosh Forensic Software (continued)

- Examining OS 9 Data Structures with BlackBag
  - Activities in this section assume you have a Macintosh running OS X
  - All data acquisitions (image files) must be configured as Disk Images
    - With the correct filename and extensions
  - To keep the correct order of each segment
    - Numbers need to be inserted between the filename and the extension
      - See Table 8-3

Using Macintosh Forensic Software (continued)

Table 8-3 Requirements for renaming Disk Image files

Original filenames for image file and segments	Macintosh Disk Image filenames
GCFI-0S9.001	GCFI-0S9.dmg
GCFI-0S9.002	GCFI-0S9.002.dmgpart
GCFI-0S9.003	GCFI-0S9.003.dmgpart
GCFI-0S9.004	GCFI-0S9.004.dmgpart

Using Macintosh Forensic Software (continued)

- Examining OS 9 Data Structures with BlackBag (continued)
  - Load images as a virtual disk image double-clicking the files in Finder
    - See Figure 8-4
  - OS X loads and displays an icon of the virtual mounted disk with the name “untitled” on the desktop
    - You can rename it with your case name
    - See Figure 8-5

Using Macintosh Forensic Software (continued)

Figure 8-4 OS X Finder showing the renamed raw files as .dmg files

Figure 8-5 Changing the icon name

Using Macintosh Forensic Software (continued)

- Examining OS 9 Data Structures with BlackBag (continued)
  - Start BlackBag from Finder
    - See Figure 8-6
  - BlackBag includes several utilities for conducting a full analysis of evidence, including
    - PDISKInfo, PMAPInfo, DirectoryScan, FileSearch, MacCarver, and FileSpy

Figure 8-6 Starting BlackBag from Finder

Using Macintosh Forensic Software (continued)

- Examining OS 9 Data Structures with BlackBag (continued)
  - Activity 1:
    - Use the BlackBag DirectoryScan utility, which lists all folders and files, visible and hidden, in the image loaded as a .dmg file
    - See Figure 8-8
  - Activity 2:
    - Use the FileSearcher utility to locate files by a specific extension
    - See Figure 8-9

Using Macintosh Forensic Software (continued)

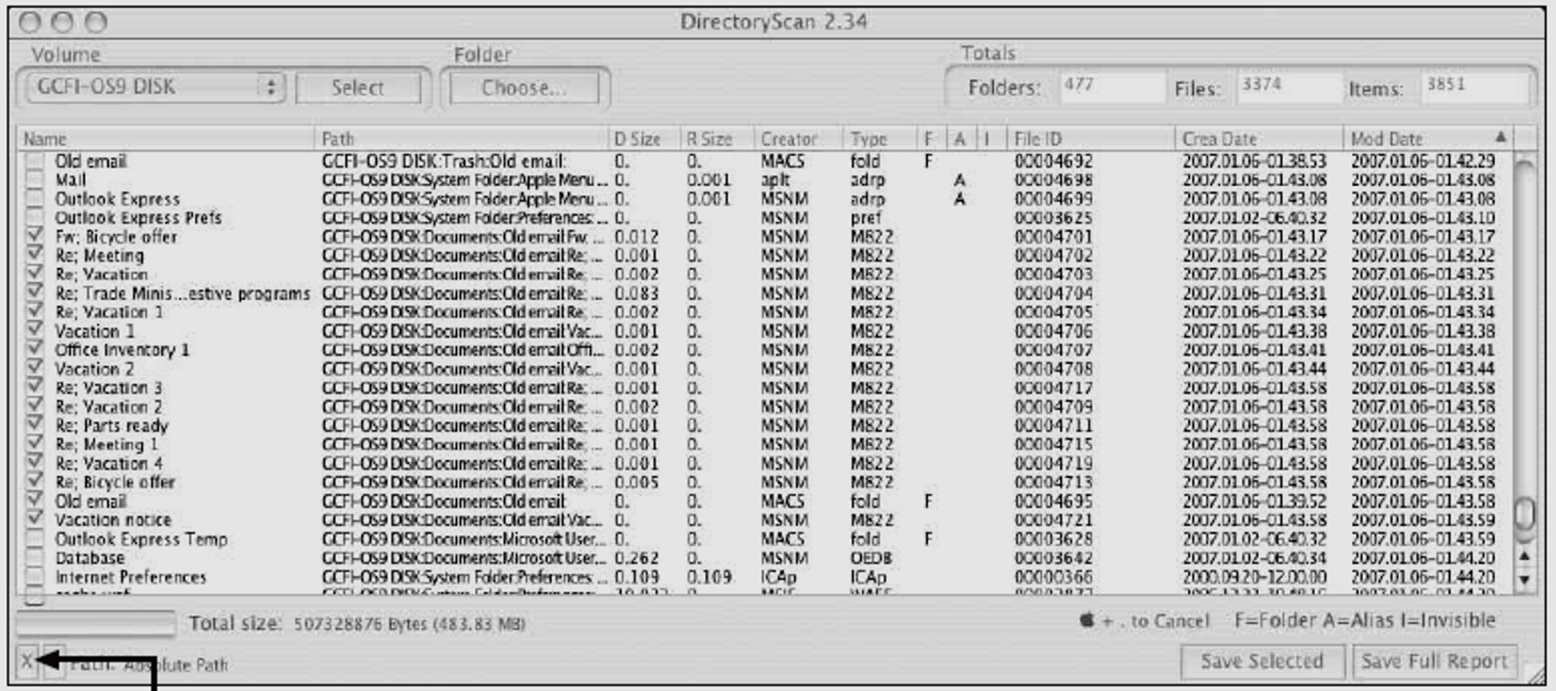

Figure 8-8 Selecting the entire GCFI-OS9 DISK volume in the DirectoryScan window

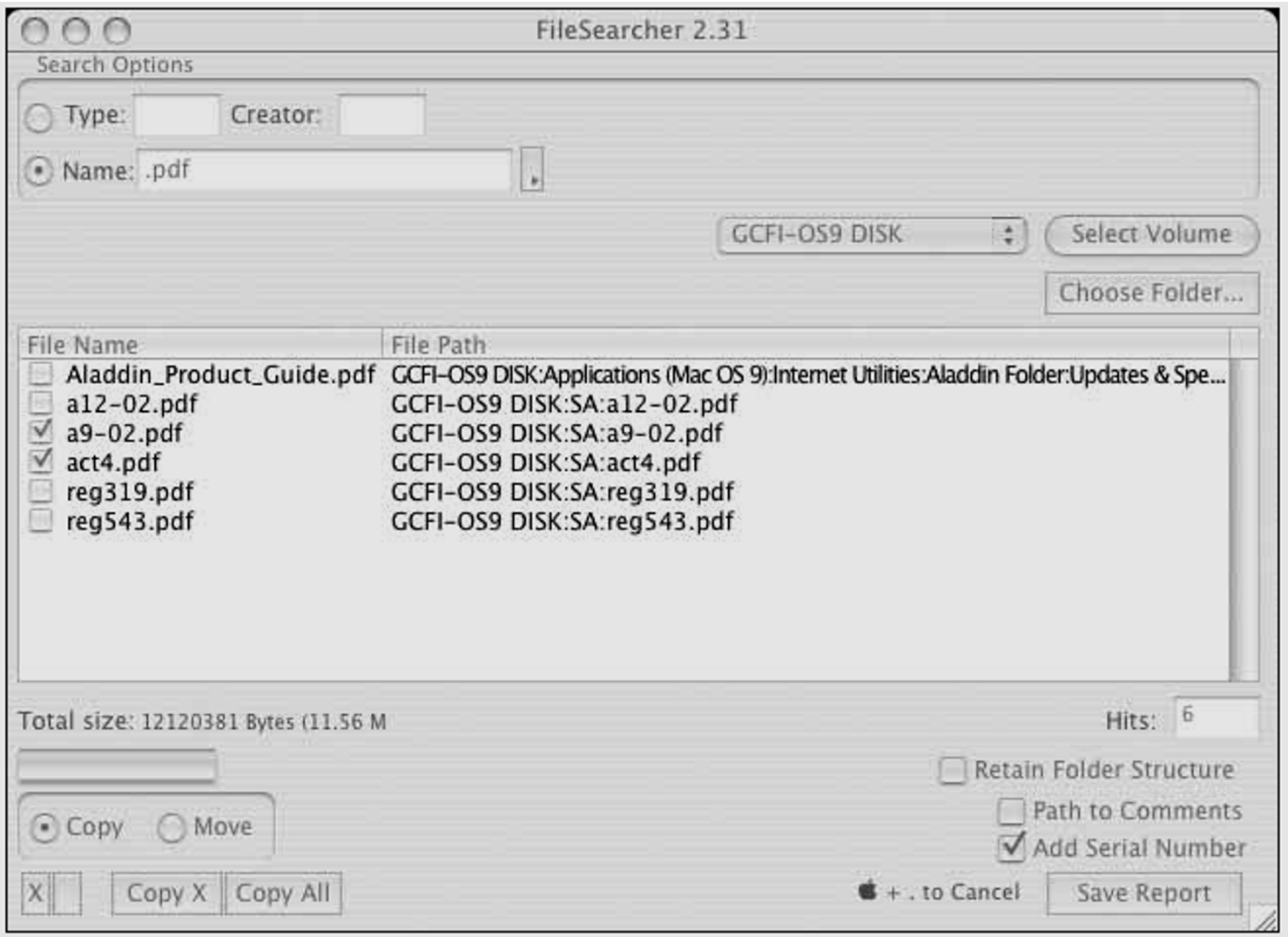

Figure 8-9 FileSearcher listing all .pdf documents in the search results

Examining UNIX and Linux Disk Structures and Boot Processes

- UNIX flavors
  - System V variants, Sun Solaris, IBM AIX, and HP-UX
  - BSD, FreeBSD, OpenBSD, and NetBSD
- Linux distributions
  - Red Hat, Fedora, Ubuntu, and Debian
  - Most consistent UNIX-like OSs
- Linux kernel is regulated under the **GNU General Public License (GPL)** agreement

Examining UNIX and Linux Disk Structures and Boot Processes (continued)

- BSD license is similar to the GPL
  - But makes no requirements for derivative works
- Some useful Linux commands to find information about your Linux system
  - uname –a
  - ls –l
  - ls –ul *filename*
  - netstat -s

Table 8-4 UNIX system files

OS	System files	Purpose
AIX	/etc/exports	Configuration file
	/etc/filesystems	File system table of devices and mount points
	/etc/utmp	Current user's logon information
	/var/adm/wtmp	Logon and logoff history information
	/etc/security/lastlog	User's last logon information
	/var/adm/sulog	Substitute user attempt information
	/etc/group	Group memberships for the local system
	/var/log/syslog	System messages log
	/etc/security/passwd	Master password file for the local system
	/etc/security/failedlogin	Failed logon attempt information
HP-UX	/etc/utmp and /etc/utmpx	Current user's logon information
	/var/adm/wtmp and /var/adm/wtmpx	Logon and logoff history information
	/var/adm/btmp	Failed logon attempt information
	/etc/fstab	File system table of devices and mount points
	/etc/checklist	File system table information (version 9.x)
	/etc/exports	Configuration files
	/etc/passwd	Master password file for the local system
	/etc/group	Group memberships for the local system
	/var/adm/syslog.log	System messages log
	syslog	System log files
	/var/adm/sulog	Substitute user attempt information

Table 8-4 UNIX system files (continued)

OS	System files	Purpose
IRIX	/var/adm/syslog	System log files
	/etc/exports	Configuration files
	/etc/fstab	File system table of devices and mount points
	/var/adm/btmp	Failed logon information
	/var/adm/lastlog	User's last logon information
	/var/adm/wtmp and /var/adm/wtmpx	Logon and logoff history information
	/var/adm/sulog	Substitute user attempt information
	/etc/shadow	Master password file for the local system
	/etc/group	Group memberships for the local system
	/var/adm/utmp and /var/adm/utmpx	Current user's logon information
Linux	/etc/exports	Configuration files
	/etc/fstab	File system table of devices and mount points
	/var/log/lastlog	User's last logon
	/var/log/wtmp	Logon and logoff history information
	/var/run/utmp	Current user's logon information
	/var/log/messages	System messages log
	/etc/shadow	Master password file for the local system
	/etc/group	Group memberships for the local system
Solaris	/etc/passwd	Account information for local system
	/etc/group	Group information for local system
	/var/adm/sulog	Switch user log data
	/var/adm/utmp	Logon information
	/var/adm/wtmp, /var/adm/wtmpx, and /var/adm/lastlog	Logon history information
	/var/adm/loginlog	Failed logon information
	/var/adm/messages	System log files
	/etc/vfstab	Static file system information
	/etc/dfs/dfstab and /etc/vfstab	Configuration files

Examining UNIX and Linux Disk Structures and Boot Processes (continued)

- Linux file systems
  - **Second Extended File System (Ext2fs)**
  - Ext3fs, journaling version of Ext2fs
- Employs **inodes**
  - Contain information about each file or directory
  - Pointer to other inodes or blocks
  - Keep internal link count
    - Deleted inodes have count value 0

UNIX and Linux Overview

- Everything is a file
  - Files are objects with properties and methods
- UNIX consists of four components
- Boot block
  - Block is a disk allocation unit of at least 512 bytes
  - Contains the bootstrap code
  - UNIX/Linux computer has only one boot block, located on the main hard disk

UNIX and Linux Overview (continued)

- Superblock
  - Indicates disk geometry, available space, and location of the first inode
  - Manages the file system
- Inode blocks
  - First data after the superblock
  - Assigned to every file allocation unit
- Data blocks
  - Where directories and files are stored
  - This location is linked directly to inodes

UNIX and Linux Overview (continued)

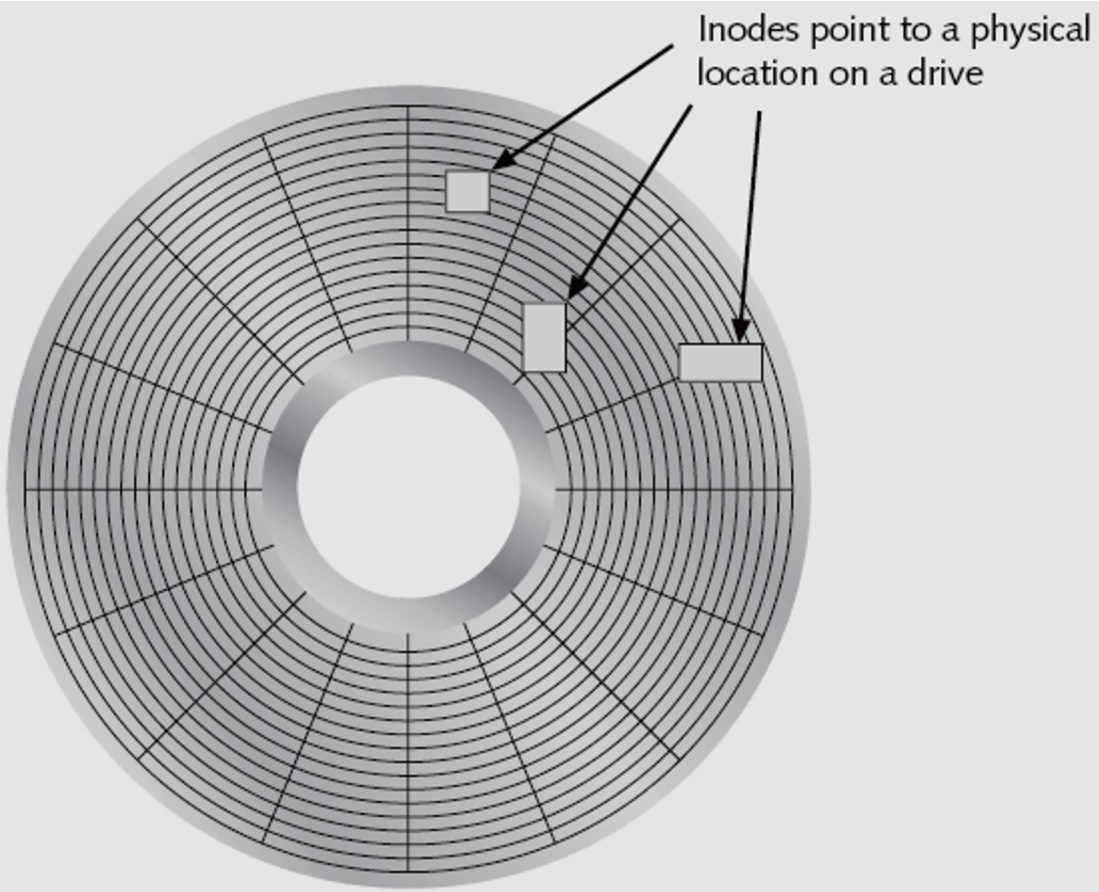

Figure 8-10 Clustering data blocks to save a file in Linux

UNIX and Linux Overview (continued)

- **Bad block inode**
  - Keeps track of disk's bad sectors
  - Commands: badblocks, mke2fs, and e2fsck/
- Linux ls command displays information about files and directories
- **Continuation inode**
  - Provides information about a file or directory
    - Mode and file type, the quantity of links in the file or directory, the file or directory status flag

UNIX and Linux Overview (continued)

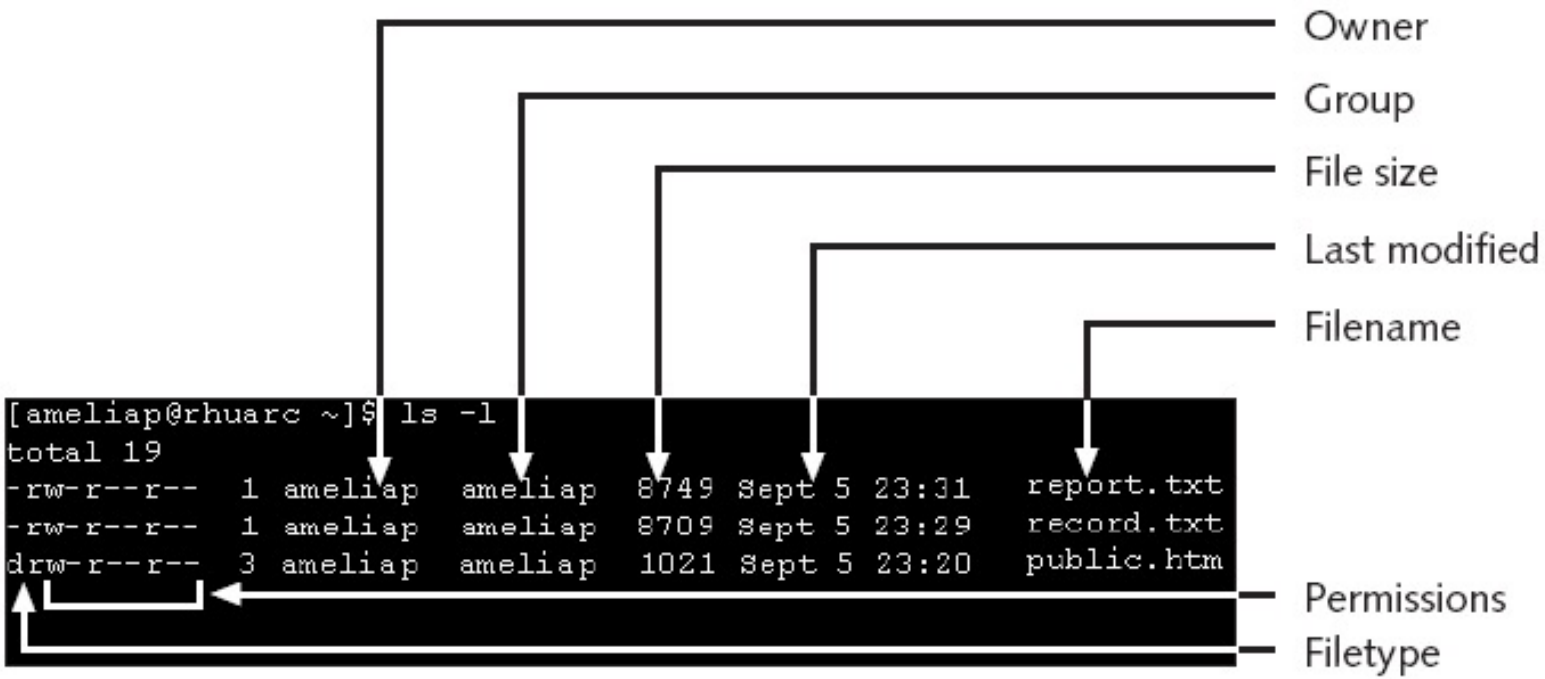

Figure 8-11 Finding information about a file

UNIX and Linux Overview (continued)

Table 8-5 Code values for an inode

Code values	Description
4000	UID on execution—set
2000	GID on execution—set
1000	Sticky bit—set
0400	Read by owner—allowed
0200	Write by owner—allowed
0100	Execution/search by owner—allowed
0040	Read by group—allowed
0020	Write by group—allowed
0010	Execution/search by group—allowed
0004	Read by others—allowed
0002	Write by others—allowed
0001	Execution/search by others—allowed

Understanding Inodes

- Link data stored in data blocks
- Ext2fs and Ext3fs are improvements over Ext
  - Data recovery easier on Ext3fs than on Ext2fs
- First inode has 13 pointers
  - Pointers 1 to 10 are direct pointers to data storage blocks
  - Pointer 11 is an **indirect pointer**
  - Pointer 12 is a **double-indirect pointer**
  - Pointer 13 is a **triple-indirect pointer**

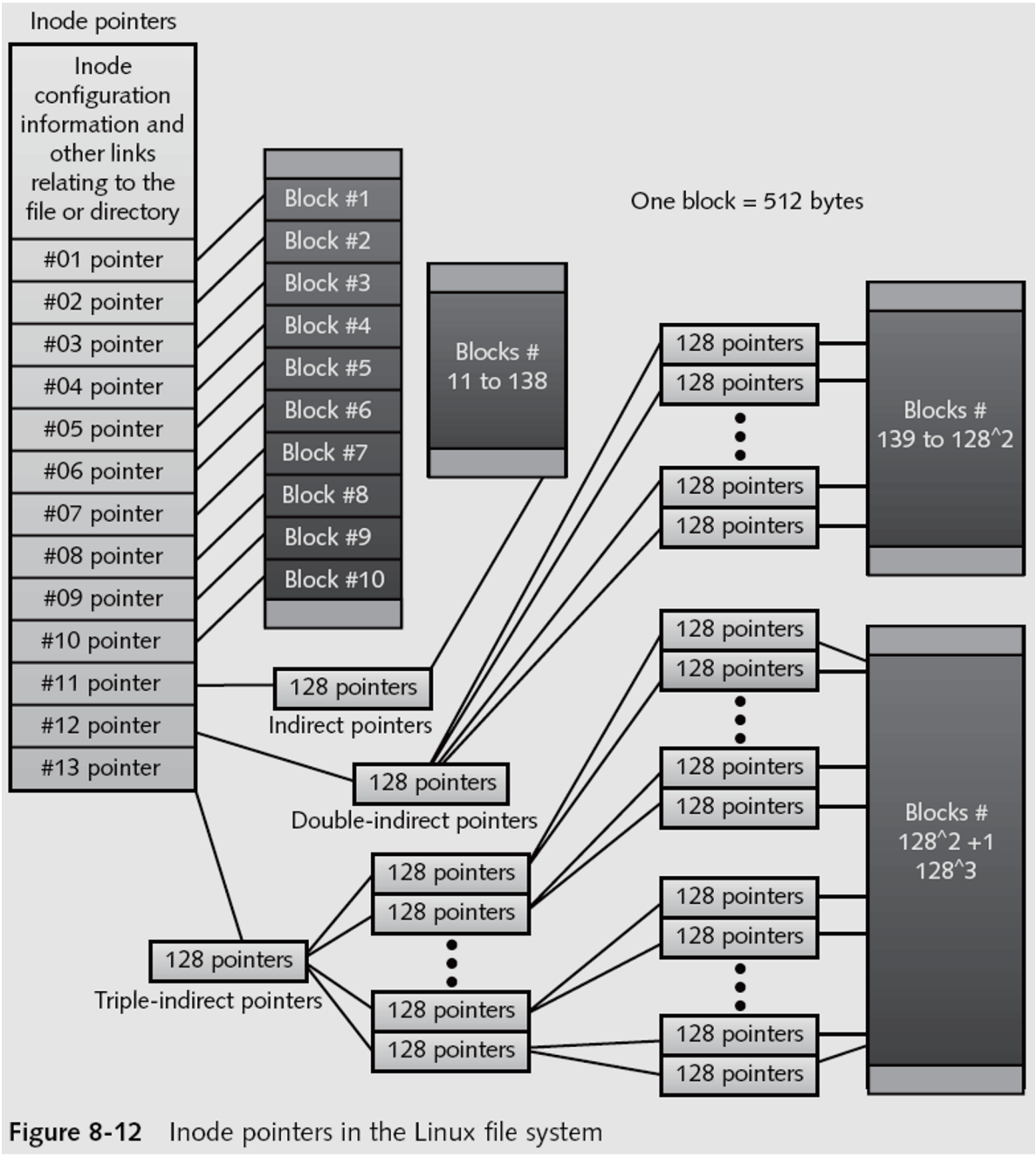

Figure 8-12 Inode pointers in the Linux file system

Understanding Inodes (continued)

Table 8-6 UNIX and Linux shell commands

Shell command	Associated options	Purpose
cat *file* more *file*		Displays the contents of a file (similar to the MS-DOS Type command)
dd	Refer to man pages for available options	Copies a disk drive by blocks, which is the same as creating an image of a disk drive
df bdf (HP-UX)	-k (Solaris)	Displays partition information for local or NFS mounted partitions
find	Refer to man pages for available options	Locates files matching a specific attribute, such as name, last modification time, or owner
netstat	-a	Identifies other systems connected via the network to a UNIX or Linux system
ps	ax (BSD) -ef (System V)	Displays the status of OS processes
uname	-a	Displays the name of the system

Understanding UNIX and Linux Boot Processes

- Instruction code in firmware is loaded into RAM
- Instruction code then:
  - Checks the hardware
  - Load the boot program
- Boot program
  - Loads kernel
  - Transfers control to kernel
- Kernel's first task is to identify all devices

Understanding UNIX and Linux Boot Processes (continued)

- Kernel
  - Boots system on single-user mode
  - Runs startup scripts
  - Changes to multiuser mode
  - Identifies root directory, swap, and dump files
  - Sets hostname and time zone
  - Runs consistency checks on the file system and mounts partitions
  - Starts services and sets up the NIC
  - Establishes user and system accounting and quotas

Understanding Linux Loader and GRUB

- Linux Loader (LILO)
  - Old boot manager
  - Can start two or more OSs
  - Uses configuration file Lilo.conf
- Grand Unified Boot Loader (GRUB)
  - More powerful than LILO
  - As LILO, it resides on MBR
  - Command line or menu driven

Understanding UNIX and Linux Drives and Partition Schemes

- Labeled as path starting at root (/) directory
  - Primary master disk (/dev/had)
    - First partition is /dev/hda1
    - Second partition is /dev/hda2
  - Primary slave or secondary master or slave (/dev/hdb)
    - First partition is /dev/hdb2
  - SCSI controllers
    - /dev/sda with first partition /dev/sda1
    - Linux treats SATA, USB, and FireWire devices the same way as SCSI devices

Examining UNIX and Linux Disk Structures

- Most commercial computer forensics tools can analyze UNIX UFS and UFS2
  - And Linux Ext2, Ext3, ReiserFS, and Reiser4 file systems
- Freeware tools include Sleuth Kit and its Web browser interface, Autopsy Browser
- Foremost
  - A freeware carving tool that can read many image file formats
  - Configuration file: foremost.conf

Examining UNIX and Linux Disk Structures (continued)

- **Tarball**
  - A data file containing one or more files or whole directories and their contents
- Installing Sleuth Kit and Autopsy
  - Requires downloading and installing the most recent updates of these tools
  - Download the most current source code from *www.sleuthkit.org*
  - To run Sleuth Kit and Autopsy Browser, you need to have root privileges

Examining UNIX and Linux Disk Structures (continued)

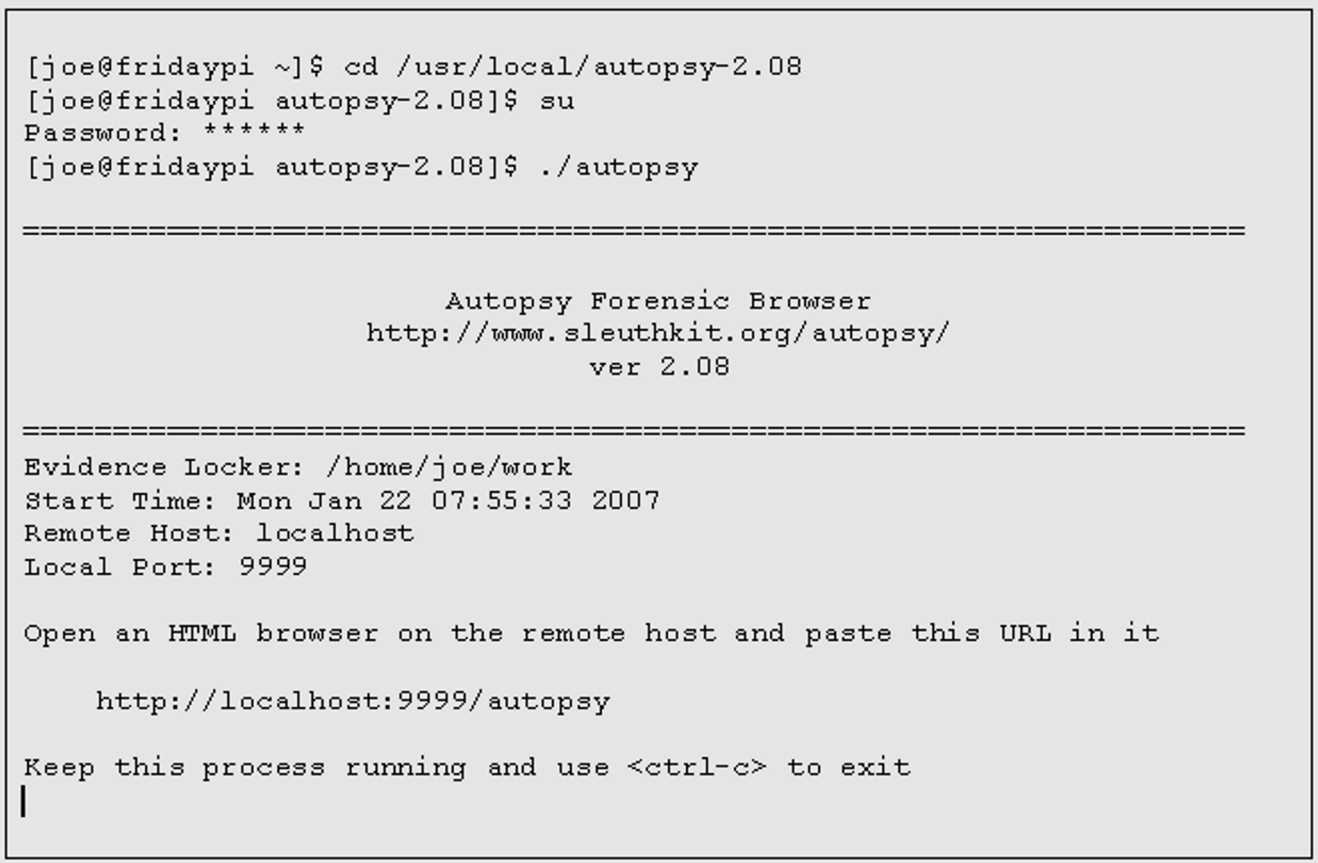

Figure 8-13 Starting Autopsy from a Linux terminal window

Figure 8-14 The Autopsy main window

Examining UNIX and Linux Disk Structures (continued)

- Examining a case with Sleuth Kit and Autopsy
  - Use Sleuth Kit and Autopsy Browser to analyze a Linux Ext2 and Ext3 file system
    - See Figures 8-15 through 8-18

Examining UNIX and Linux Disk Structures (continued)

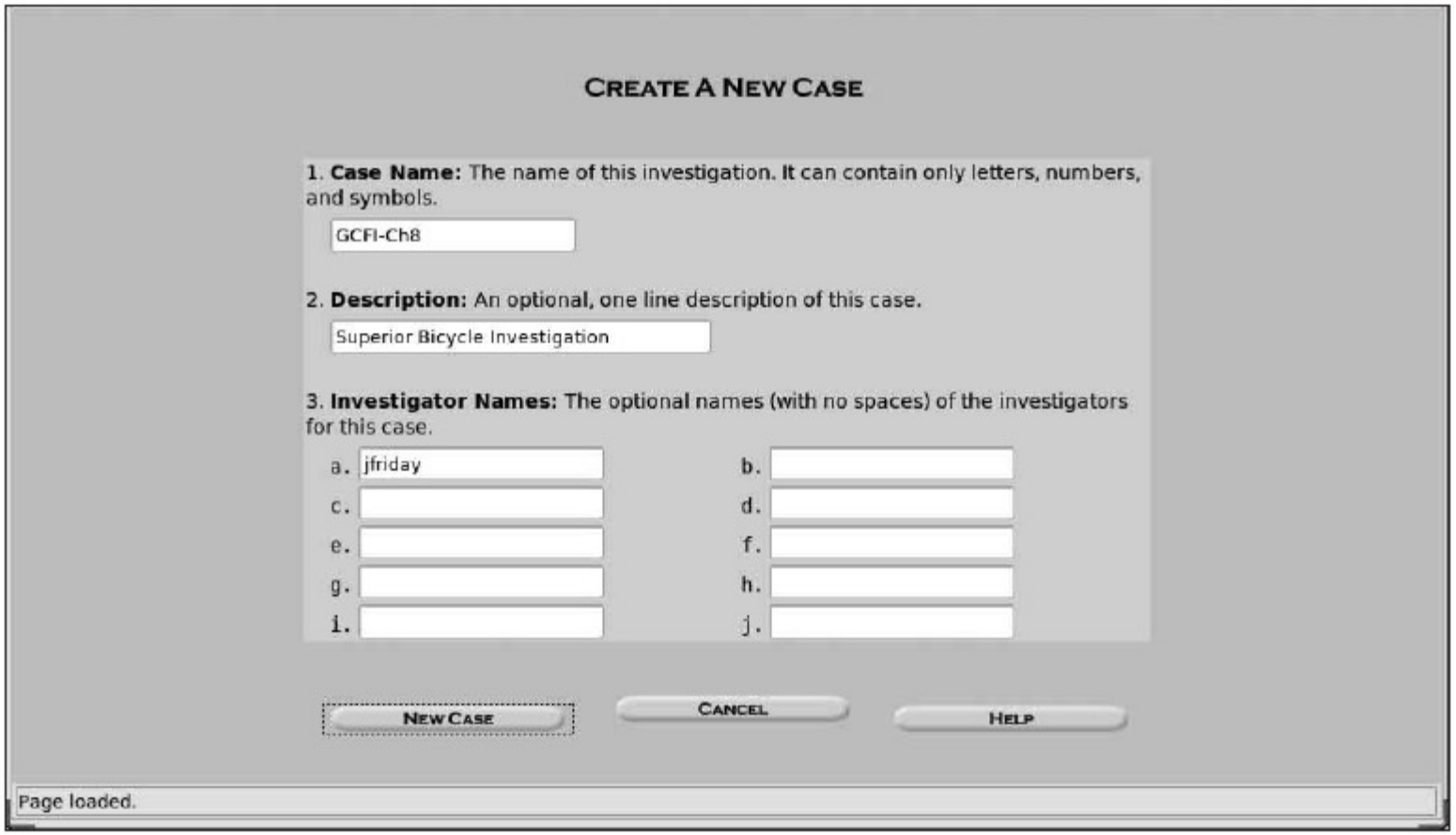

Figure 8-15 The Create A New Case dialog box

Examining UNIX and Linux Disk Structures (continued)

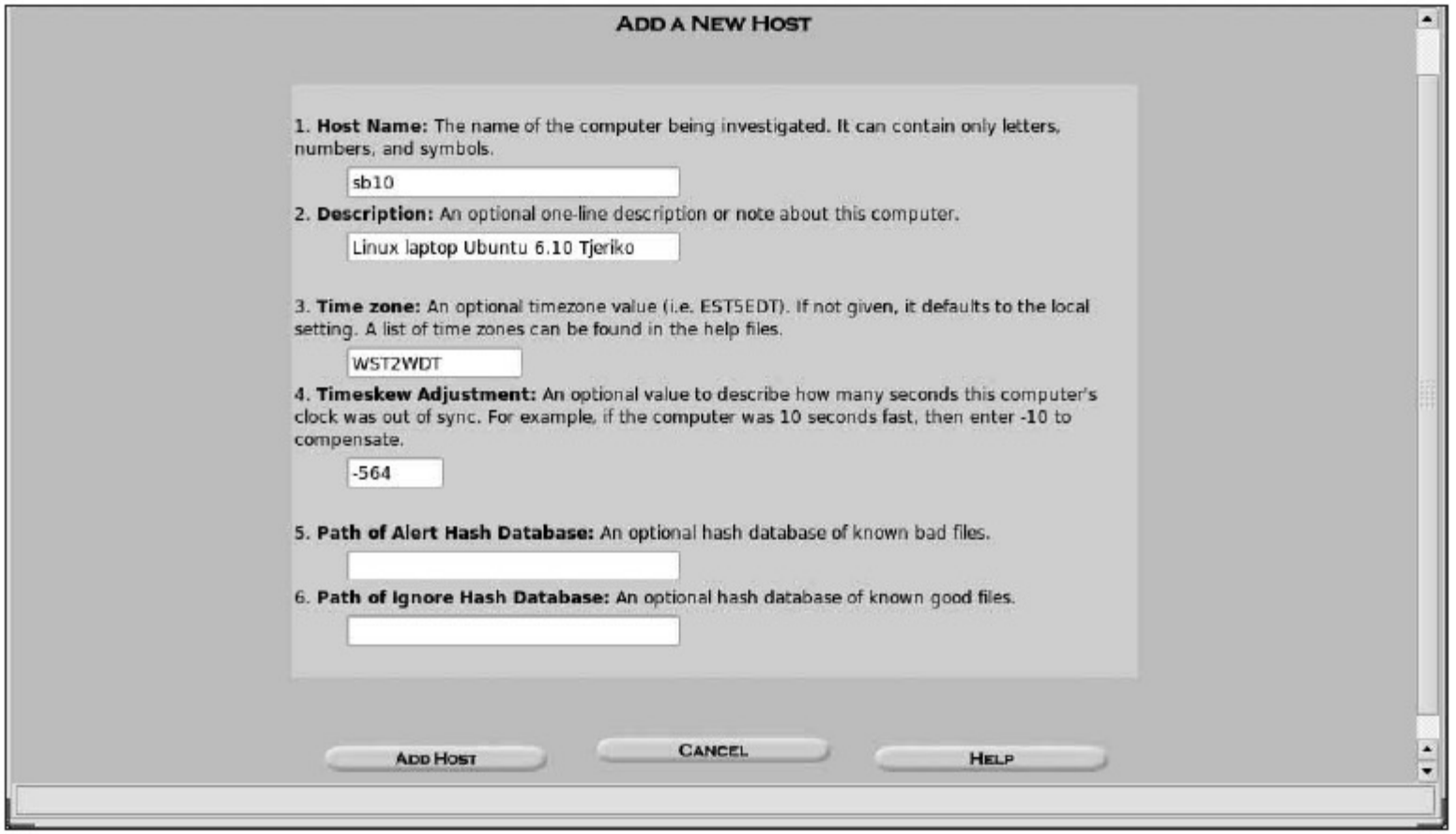

Figure 8-16 The Add A New Host dialog box

Examining UNIX and Linux Disk Structures (continued)

Figure 8-17 The Keyword Search dialog box

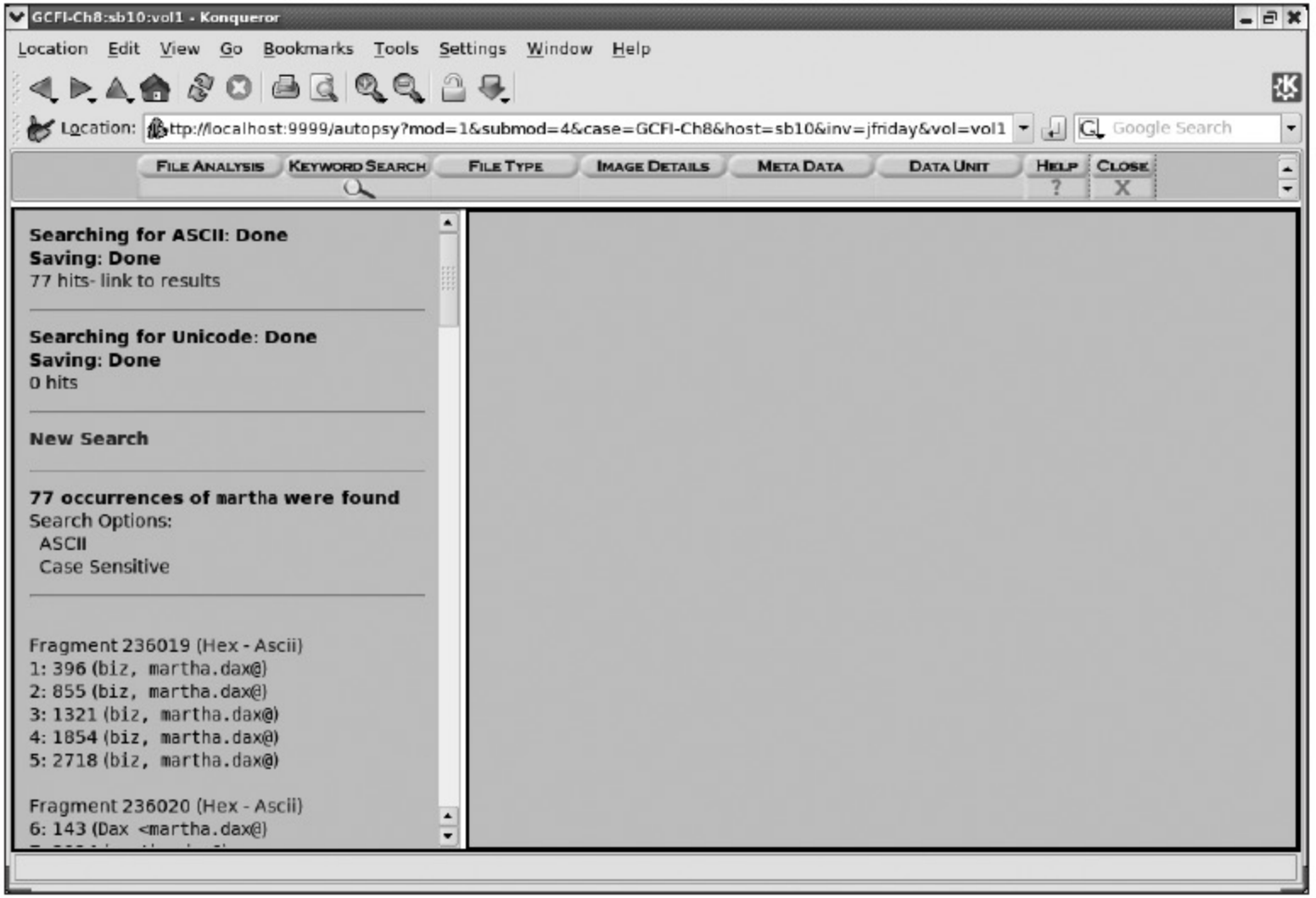

Figure 8-18 Summary of search results

Examining UNIX and Linux Disk Structures (continued)

- Examining a case with Sleuth Kit and Autopsy (continued)
  - Use the File Activity Time Lines function
    - Identifies what files were active at a specific time
    - See Figures 8-19 and 8-20

Examining UNIX and Linux Disk Structures (continued)

Figure 8-19 The Select a volume to analyze or add a new image file dialog box

Examining UNIX and Linux Disk Structures (continued)

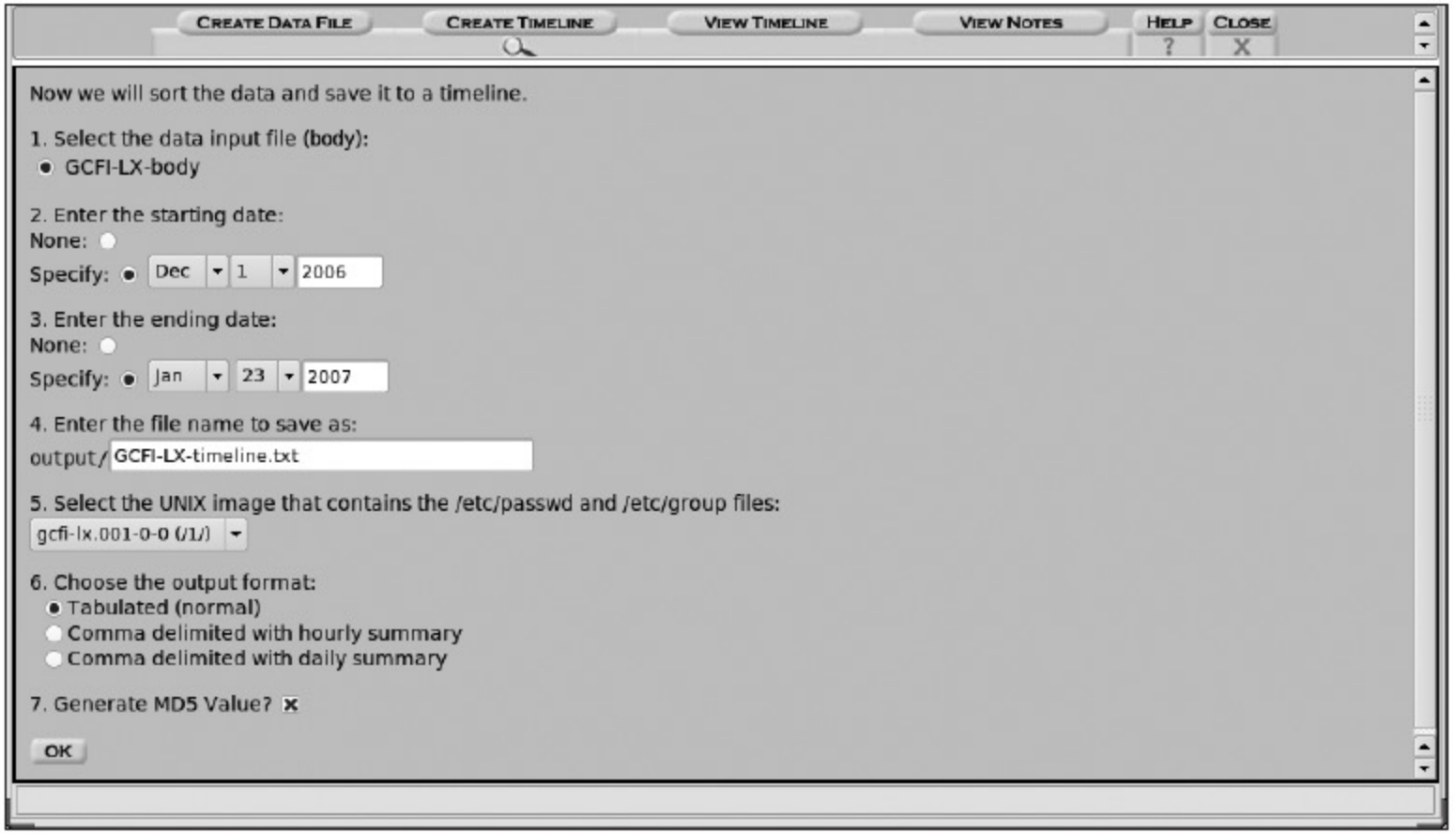

Figure 8-20 Entering timeline options

Understanding Other Disk Structures

- SCSI disks
- IDE/EIDE disks
- SATA drives

Examining CD Data Structures

- Laser burns flat areas (lands)
- Lower areas are called pits
- Transitions
  - From lands to pits have binary value 1 (on)
  - No transition has binary value 0 (off)
- **International Organization of Standards (ISO)**
  - ISO 9660 for CD, CD-R and CD-RW
  - ISO 13346 for DVDs

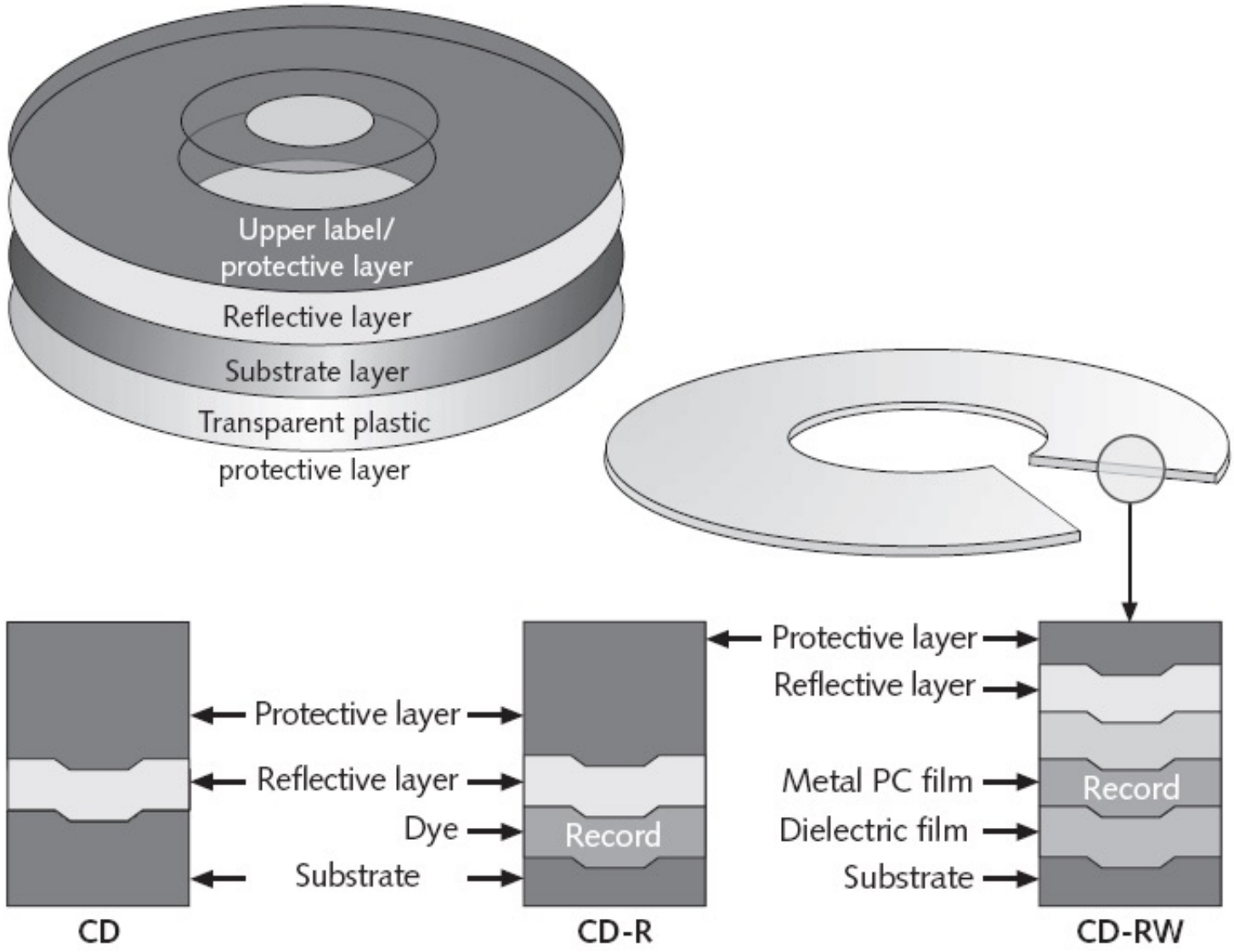

Figure 8-21 Physical makeup of a CD

Examining CD Data Structures (continued)

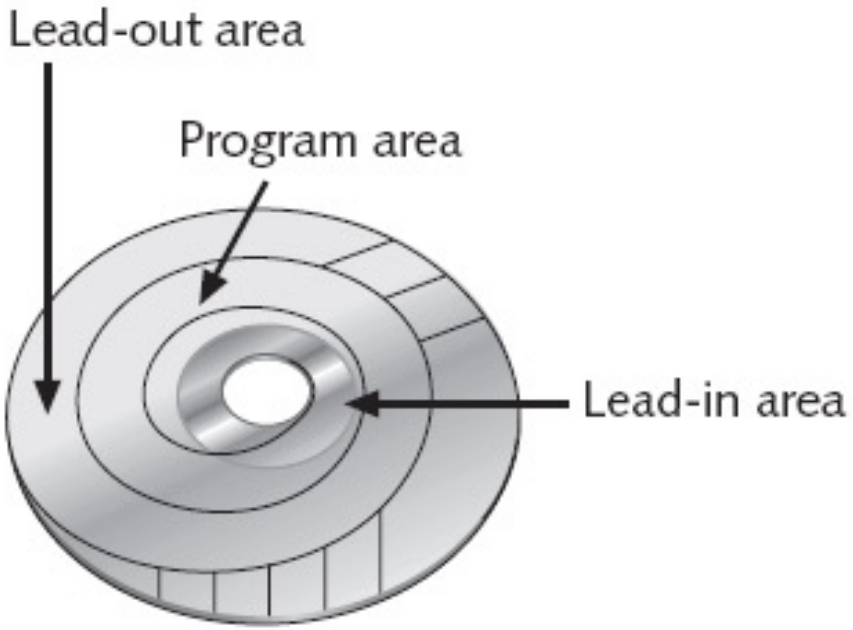

Figure 8-22 Logical layout of a CD

Examining CD Data Structures (continued)

- Frame is the unit storage
  - Contains 24 17-bits symbols
- Frames are combined into blocks
- Blocks are combined into sectors
  - 2352 bytes for CD-DA
  - 2048 bytes for CD
- **Constant Linear Velocity** (< 12X)
- **Constant Angular Velocity** (>= 12X)

Examining CD Data Structures (continued)

- DVD disk file structures use a Universal Disk Format (UDF)
  - Called Micro-UDF (M-UDF)
- For backward compatibility, some DVDs have integrated ISO 9660
  - To allow compatibility with current OSs

Examining SCSI Disks

- **Small Computer System Interface (SCSI)**
  - Provides a common bus communication device
- During investigation
  - Check if the device is internal or external
  - Check if card, cables, adapters, terminators, and drivers are available
- **Advance SCSI Programming Interface (ASPI)**
  - Provides several software drivers for communication between the OS and SCSI component

Examining SCSI Disks (continued)

- Might need to adjust settings
  - Port numbers and terminators
- Newer SCSI devices typically use an integrated self-terminator
- One problem with older SCSI drives is identifying which jumper group terminates and assigns a port number

Examining IDE/EIDE and SATA Devices

- Most forensic disk examinations involve EIDE and SATA drives
- ATA drives from ATA-33 to ATA-133
  - Standard 40-pin ribbon or shielded cable
  - 40-pin/80-wire cable for ATA-66, 100, and 133
- CMOS identifies proper disk settings using:
  - Logical block addressing (LBA)
  - Enhanced CHS configurations
- Can be a problem during an investigation

Examining IDE/EIDE and SATA Devices (continued)

- Solutions
  - Use disk imaging tools
  - Use an old PC
  - Cards and adapters
    - ISA SCSI card
    - A-Card IDE adapter
    - SCSI-to-IDE adapter
    - EISA FireWire card
    - FireWire-to-EIDE adapter

Examining IDE/EIDE and SATA Devices (continued)

- Examining the IDE host protected area
  - ATAPI-5 AT introduced in 1998 reserved and protected areas on IDE devices
    - Protected Area Run Time Interface Extension Service (PARTIES)
  - Data stored by diagnostic and restore programs
  - Tools
    - X-Ways Replica
  - HPA is also referred to as a BIOS Engineering Extension Record (BEER) data structure

Examining IDE/EIDE and SATA Devices (continued)

- Exploring hidden partitions
  - Suspects try to conceal evidence by hiding disk partitions
  - Norton Disk Edit can change the disk partition table
    - Leaving no indication that the deactivated partition exists
  - Use imaging tools that can access unpartitioned areas of a drive

Summary

- Macintosh uses HFS
  - Hierarchical structure
- Mac OS file structure
  - Data fork and resource fork
- Volume refers to any storage media
  - Allocation and logical blocks
- Ext2fs uses inodes
  - Ext3fs: journaling version of Ext2fs

Summary (continued)

- Linux file structure
  - Metadata and data
- CD and DVDs are optical media
  - ISO 9660 and 13346
- Other device technologies
  - SCSI
  - IDE/EIDE
  - SATA