PURDUE UNIVERSITY®

CS 50011: Introduction to Systems II

Lecture 1: Unix Fundamentals

Prof. Jeff Turkstra

General information

- Course Blackboard page for lab submission and grades
- Most information on the course website: http://courses.cs.purdue.edu/cs50011:start
  - Under “Course Content”
  - …and “Syllabus”

Grades

- 50% from Module 1
- 25% Laboratory Exercises (~4)
- 25% “Final” Exam
  - Only covers module 2 material
  - Wednesday, August 2
  - 1:00pm-3:00pm, LWSN B134

Introduction

- Module 2 will be a whirlwind coverage of:
  - *NIX
  - Computer architecture
  - x86 Assembly
  - Virtual memory
  - Processes
  - Networking
  - Sockets
  - Databases/SQL
  - HTTP

About me

- BSCmpE, MSECE, and PhD from Purdue University
  - Focused on operating systems and distributed systems
  - Instructor, ECE 2005-2008
  - Software Engineer with HUBzero/RCAC
  - Microfluidic Innovations, LLC and other startups
  - Started past January with CS
- Current academic activity
  - CS 307, CS 180, CS 250, CS 50011
  - Metachory
- Enjoys
  - Linux, skiing, piano/saxophone, flying, HAM, etc

Slides

- Some slides are based on Silberschatz, Galvin and Gagne's Operating Systems Concepts

Lecture 01

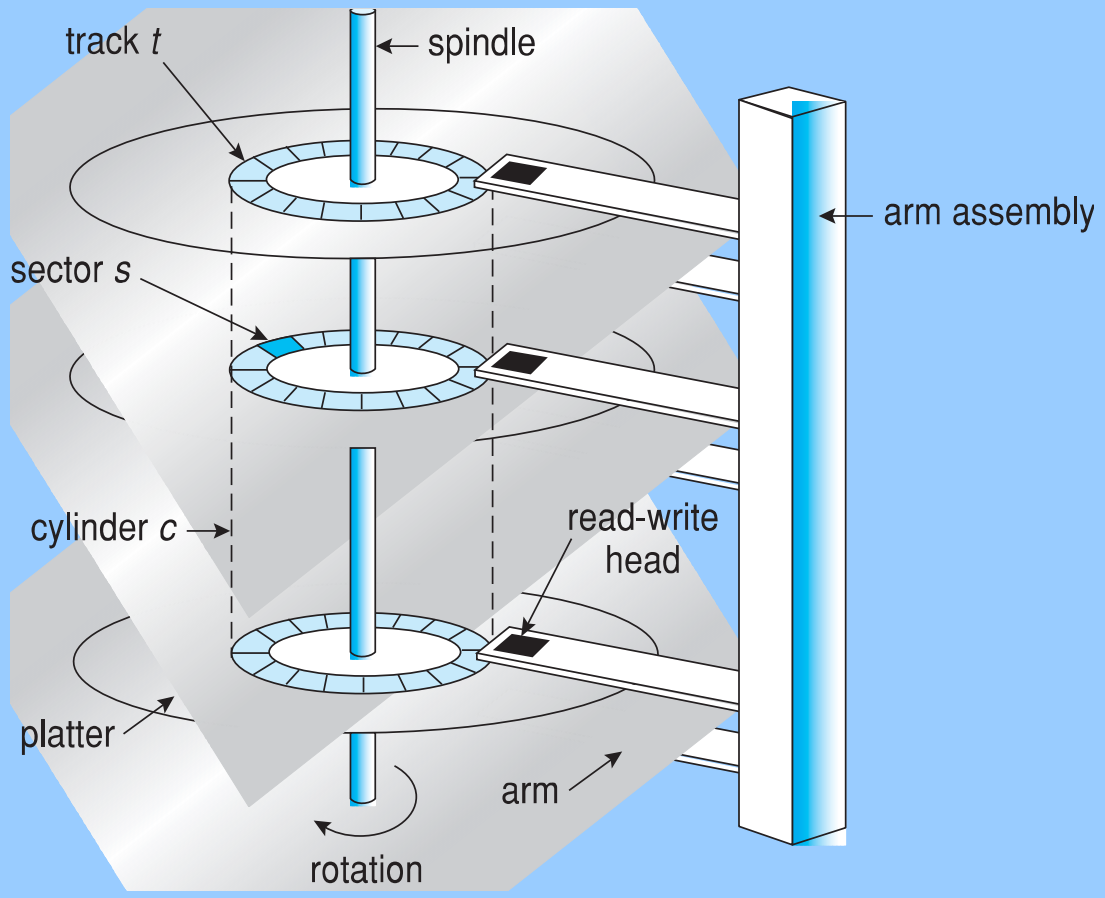
- File systems
- Access control
- More utilities

Starting at the bottom

- Block device
  - Hard disk
  - SSD
  - Tapes
  - More
- At least an order of magnitude (or more) slower than main memory
  - Fastest SSDs ~550MB/sec
  - DDR4 ~16,155MB/s
  - Latency worse

Hard drives

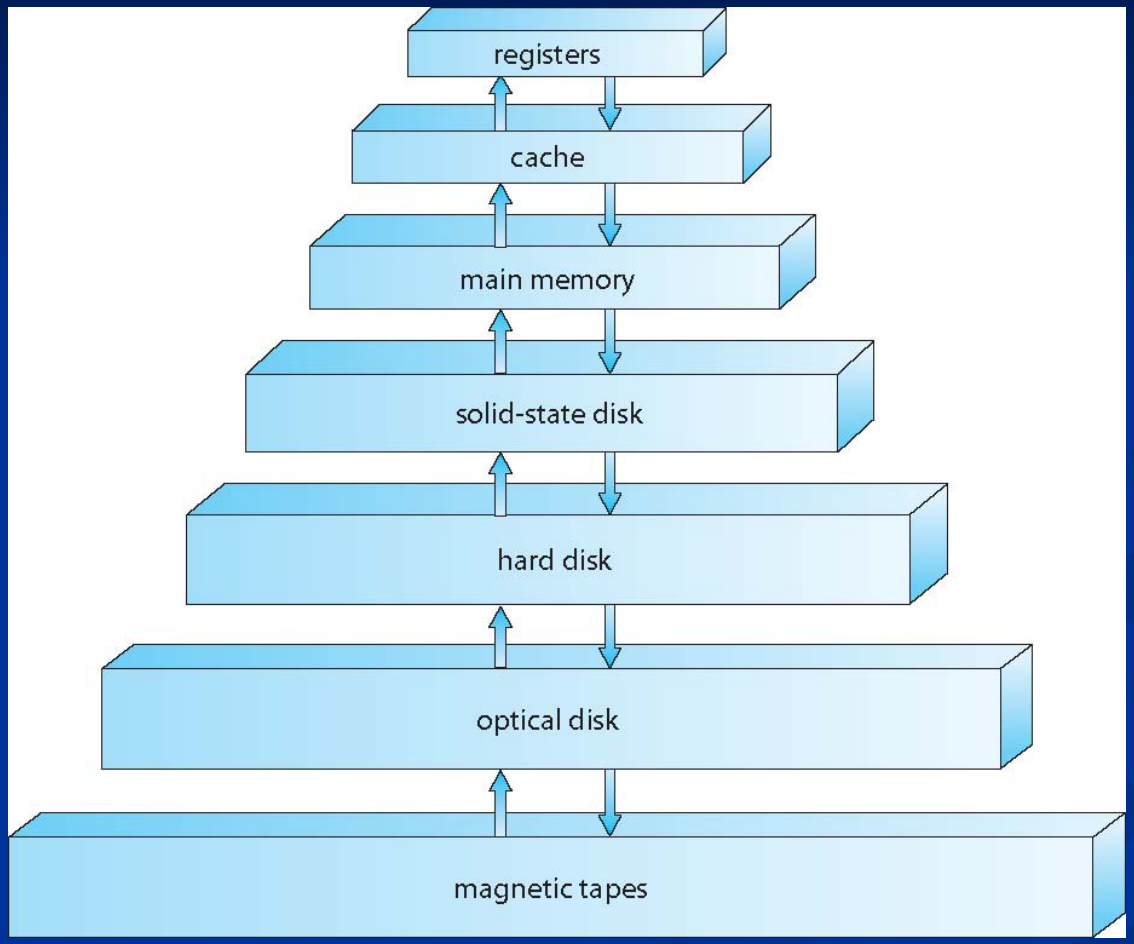
- Mechanical
  - Spinning platters
  - Moving heads
- Modern
  - Lie about sector size
  - On-board cache
  - ECC (Reed-Solomon)
  - Controller handles physical sector remaps



SSDs

- Solid state
  - No moving parts
  - Wear leveling
- 4-5 times faster than HDs
- Hybrid drives

Storage hierarchy



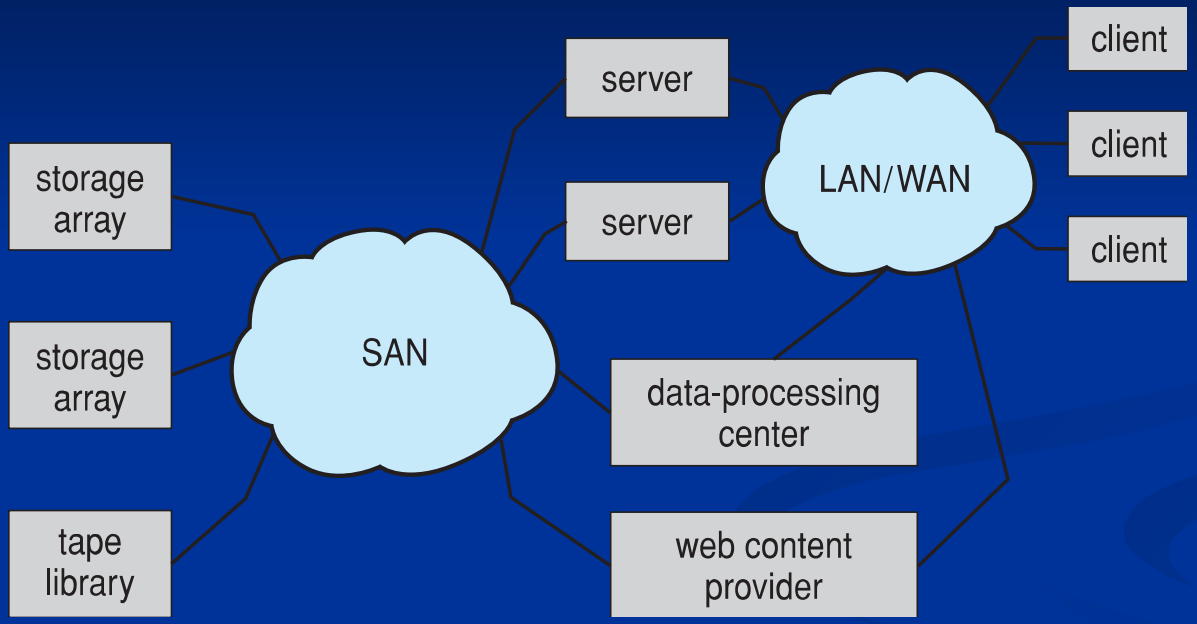
Disk structure

- Large one-dimensional arrays of logical blocks
  - Smallest unit of transfer
- Blocks mapped onto sectors
  - Sector 0 first sector, first track, outermost cylinder
  - Non-constant number of sectors per track
    - Constant angular velocity
- Bad sectors

Disk attachment

- Host-attached: SCSI, SATA, etc
- Fibre Channel
  - Often basis of a Storage Area Network (SAN)
- Network attached storage (NAS)

Storage area network



- Common in large storage environments
- Multiple hosts attached to multiple storage arrays

SAN

- One or more storage arrays
  - Connected to one or more Fibre Channel switches
- Hosts attach to switches as well
- Storage made available via LUN Masking

Network attached storage

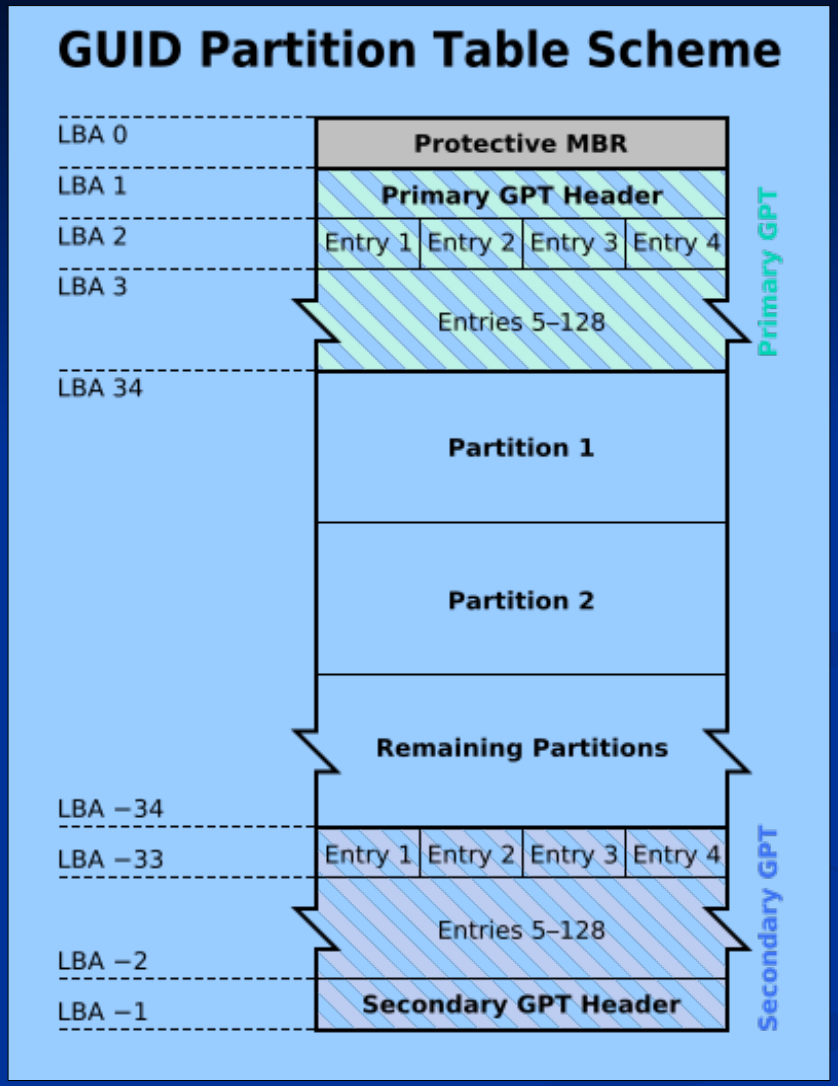
- NAS, storage made available over network
- Remotely attaching file systems
- NFS, CIFS, Samba
- Remote procedure calls (RPCs) between hosts
- iSCSI
  - Uses IP network to carry SCSI protocol

Formatting

- Low-level or physical formatting
  - Divides disks into sectors
  - Each sector holds header information, data, and error correction code (ECC)
  - Usually 4096 bytes now
    - Used to be 512 bytes
    - Many disks can mimic 512 byte sectors
      - There's a cost if misaligned
- Logical formatting

Partitions

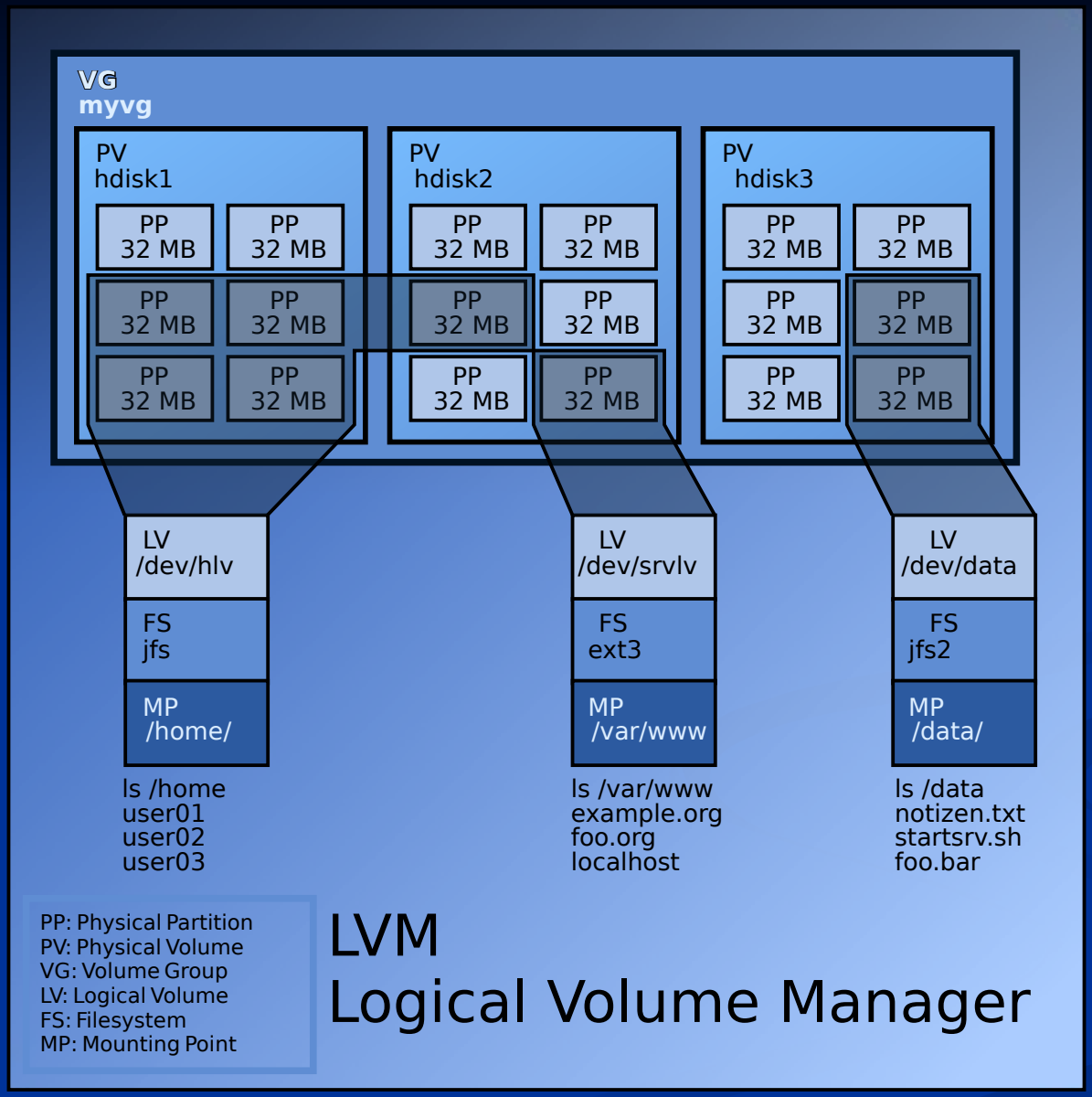
- MBR – Master Boot Record
  - IBM PC DOS 2.0, 1983
  - Limit of 2TiB disk and partition size
  - Four primary partitions
  - Extended partitions
- GPT – GUID Partition Table
  - Part of UEFI
  - Relaxes above limitations
  - 128 partitions for Windoze
  - CRC ECC
  - Protective MBR



Partition formats

- Regular FS (0x83)
- Swap (0x82)
- LVM Physical Disk (0x8e)
- Linux raid autodetect (0xfd)
- Often ignored

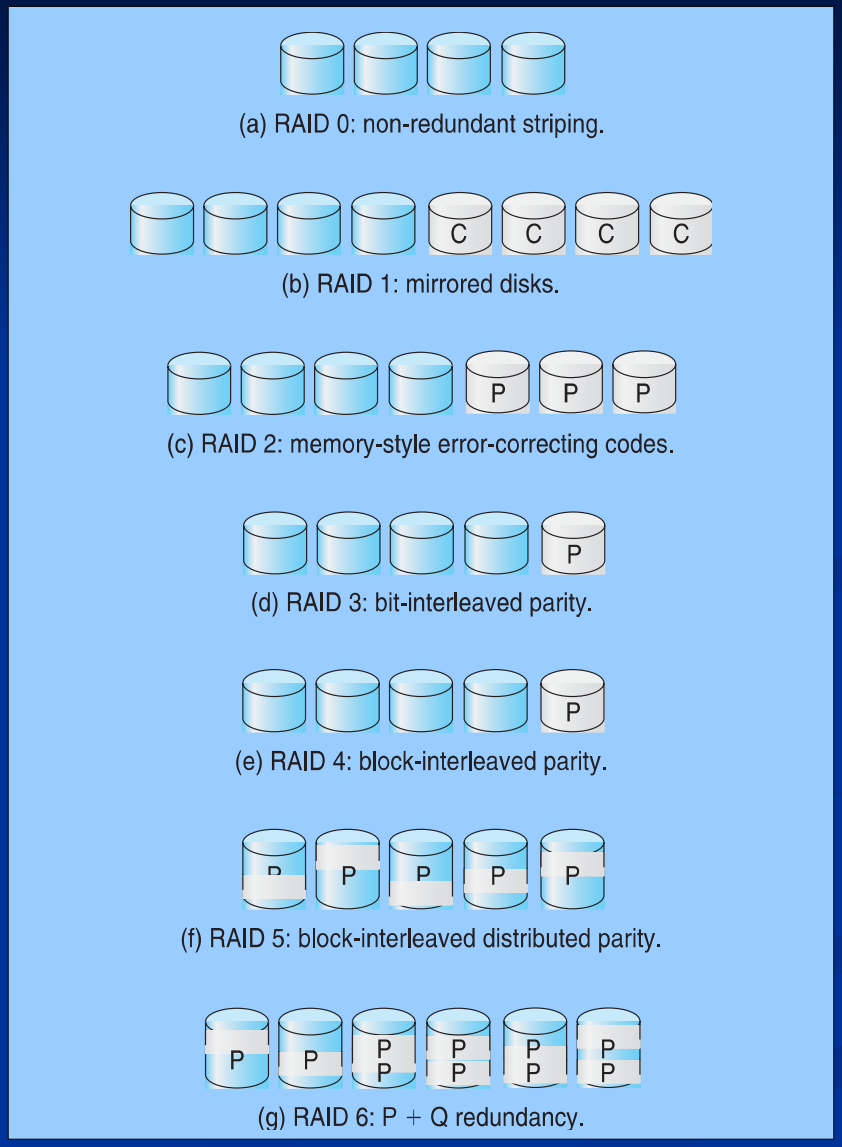
- fdisk/gdisk demo



md (multiple device)

- Virtual devices created from one or more independent underlying devices
  - RAID-0: Block level striping
  - RAID-1: Mirrored
  - RAID-4: RAID-0 + parity
  - RAID-5: Distributed parity
  - RAID-6: RAID-5, except two parity segments
  - RAID 10: RAID-0 striped over RAID-1

RAID levels



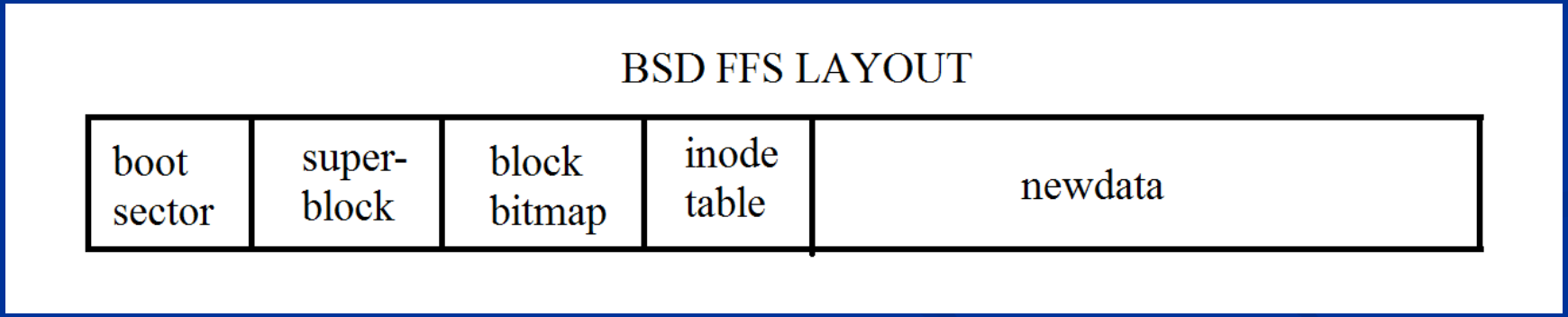
RAID

- Redundant array of disks
- RAID is not a backup
- Fault tolerant
  - Hot spares

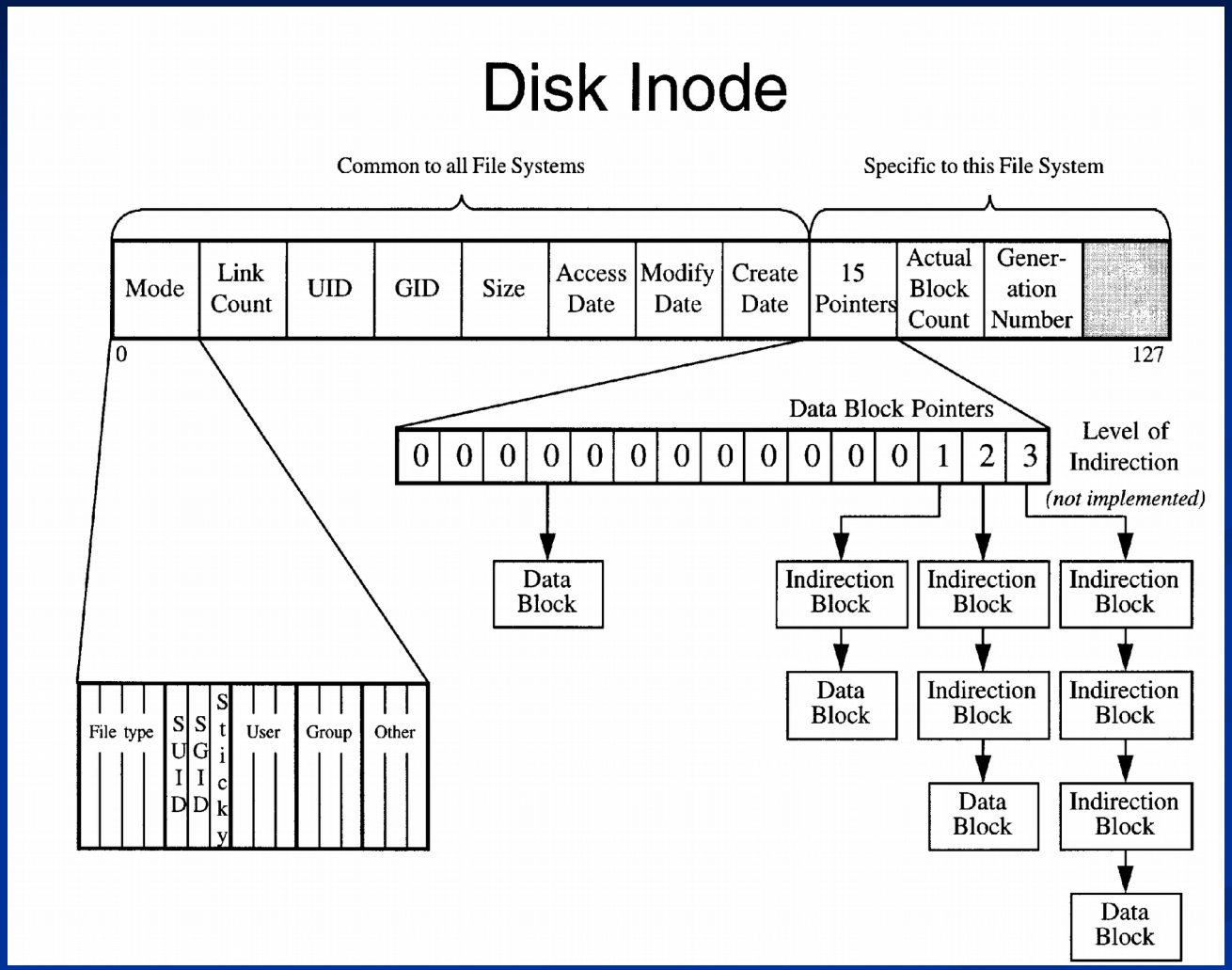
dm-crypt and LUKS

- dm-crypt
  - Encrypted block devices
- LUKS
  - Linux Unified Key Setup
  - Standardizes partition headers and data formats
- cryptsetup
  - Convenient interface to create encrypted block devices using the LUKS extension

BSD File System



inode



* http://web.cs.ucla.edu/classes/spring14/cs111/scribe/12b/

inode

- File size
- User ID (uid)
- Group ID (gid)
- Mode (rwx, special flags)
- Timestamps (ctime, atime, mtime)
- Link count
- Pointers to data blocks
- Many dictated by POSIX

Why do we care?

- Sometimes information security involves forensics
  - Knowing that there may be unwiped flash cells due to wear-leveling
  - Exploring the free blocks on a disk
  - FAT – put a NULL for the first character to delete the file
    - Exceptionally easy to “undelete”
    - Still relevant!

Linux file systems

- Actual file system varies
  - ext2/3/4
  - XFS
  - btrfs
  - ZFS
  - ...and others

- Userland view is generally the same
  - Ownership
  - Permissions
  - Date and time information
  - Number of links
  - File size
  - Extended attributes
  - Directory hierarchy
- Kernel VFS layer

Exploring

- fdisk/gdisk
- mkfs
- tune2fs -l
- cryptsetup luksDump
- ls -ia
- stat

The UNIX mantra

- “On a UNIX system, everything is a file; if something is not a file, it is a process.”
- No difference between a file and a directory
  - Directory is just a file containing names of other files

Types of files

- Directories
  - Lists of other files
- Special files
  - Mechanisms for input/output
  - Often in /dev
- Links
  - Symbolic links
  - Hard links

Types cont.

- (Domain) Sockets
  - Inter-process networking protected by file system's access control
- Named pipes
  - Similar to sockets, without the networking semantics
- Regular files

ls -l

Symbol	Meaning
-	Regular file
d	Directory
l	Symbolic link
c	Special file
s	Socket
p	Named pipe
b	Block device

- Or, maybe ls -F

File permissions

- Read: access the contents of a file
  - For directories, list the file names in a directory
- Write: modify a file
  - For directories, create/delete/rename
- Execute or search: execute a file
  - Not necessarily read its contents, though
  - Must be readable for interpreted files (eg, shell scripts, python, etc)
  - Directories: access a file given its explicit path. Cannot list files without the read bit

Classes

- User: the file owner
- Group: members of the group that owns the file
- Other: anyone that does not fall into the first two classes

Setting the mode

- chmod
  - Symbolic: ugoa[+-]rwxXst
  - Numeric:
    read = 4 (0b100)
    write = 2 (0b010)
    execute = 1 (0b001)
    - Eg, chmod 0711 myfile
  - 4000 for setuid
  - 2000 for setgid
  - 1000 for sticky

setuid/setgid bits

- setuid: when executed, file runs as the user/owner
  - Specifically, the process' effective uid is the owner's
- setgid: same idea, but with gid
  - Except for directories: files created within a setgid directory inherit its group

sticky bit

- Applies to directories only
  - Well, almost
- Users cannot rename/move/delete files owned by other users
  - Even if they have write permission to the directory
  - Doesn’t apply to directory owner
- Why?

Examples

- ls -l
- chmod 4700 /usr/bin/vim
  - Or chmod u+s
- chmod 2700 /usr/bin/vim
  - Or chmod g+s
- chmod 1755 /tmp
  - Or chmod +t

Extended attributes

- Extension to the normal attributes associated with every inode in the system
- name:value pairs associated with files
- Eg, setfacl, getfacl
- -rwxr-xr-x+
- setfacl -x to remove
- getfattr

Examples

- setfacl -m u:apache:r /some/path
  getfacl /some/path
  ls -l

Discretionary Access Control

- User dictated
- Eg, classic file permissions
- POSIX Access Control Lists (ACLs)

Mandatory Access Control

- ...or MAC.
- Policy-based access control

SELinux

- Security-Enhanced Linux
- Implements MAC
- Set of kernel modifications and userland tools
  - Originally from the NSA
- Added to mainline kernel as of 2.6
- Originally included in RedHat
  - CentOS and Scientific Linux
  - Fedora *by default*
- Now Debian, Ubuntu, openSUSE, etc optionally

How?

- ls -Z
- chcon
- restorecon
- Etc

  chcon -R -t httpd_user_content_t
  setsebool -P httpd_can_network_connect on
  setsebool -P httpd_can_sendmail on

Sample policy

Building a policy

- checkmodule -M -m -o modname.mod modname.te
  semodule_package -o modname.pp -m modname.mod
  semodule -i modname.pp

Questions?