More on the Filesystem

Module 9

Overview

- In this module:
- Linking File Names
- Inodes
- Directories
- Symbolic Links

Linking File Names

- To link a file
- ln file1 file2
- After linking
- both file names refer to the same contents
- both names are equivalent
- either can be renamed (mv) or removed (rm)
- This is a ‘hard’ link

An inode

- An inode is
- a file's contents and metadata
  - the 'real' file on disk
- uniquely numbered within a filesystem

A Directory

- ...is simply a 'file' which maps
- a file **name** to...
- an **inode number** (or i-number)

The '.' and '..' Directories

- Each directory, when created, is given two entries:
  - . current directory
  - .. parent directory
- These are just plain directory entries, like any other
  - Automatically 'linked' by the kernel (no other hard links allowed)

Limitations of Hard Links

- Hard links cannot
- link across filesystems
- link directories

Symbolic Links

- A symbolic ‘name’ for another file
- To create:
- ln -s existingname newname
- Benefits
- Can link across devices
- Can link directories

Symbolic Links – Implementation

- Simply a ‘file’ containing the name of another file
- The ‘existingname’ doesn’t need to exist
- Kernel marks symbolic links specially