

• NFS allows access to other systems' disks.

## Linux Filesystems — Implementation Overview

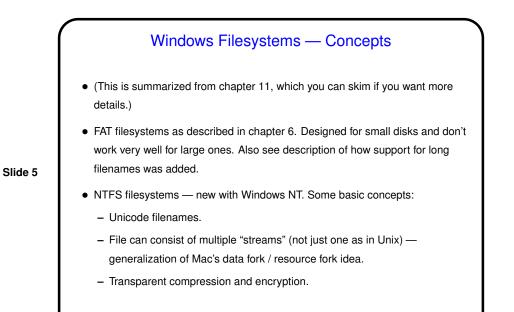
- Originally, MINIX filesystem only similar to early Unix.
- Later, VFS (virtual filesystem) added as intermediate layer to support many kinds.

Slide 4

• ext2/ext3 filesystems (ext3 is ext2 with addition of "journal" file and support for journaling): Similar to FFS, but with a single blocksize, and "block groups" rather than "cylinder groups". Block group also includes bitmap for free space. Attempts to allocate all space for file within block group (may account for less fragmentation).

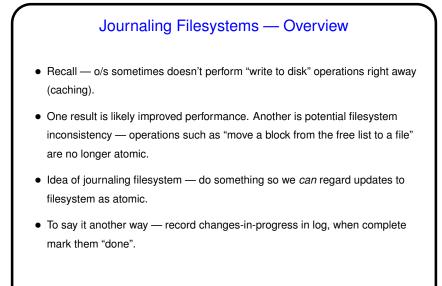
Superblock has bit that says whether filesystem is "clean" (no fsck needed at boot time).

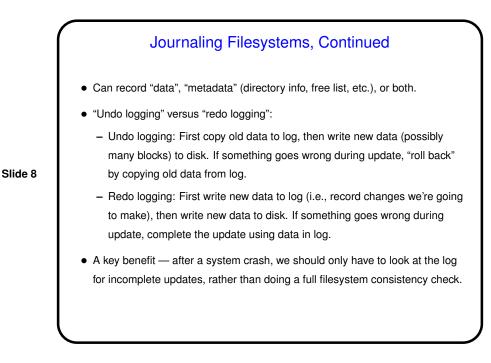
• /proc filesystem represents much system info.

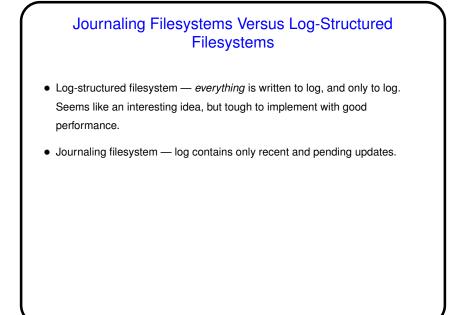


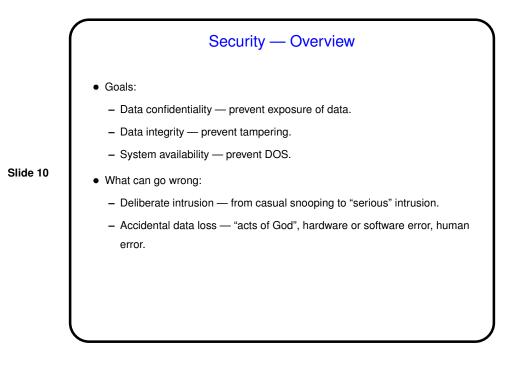
## Windows Filesystems - NTFS Implementation

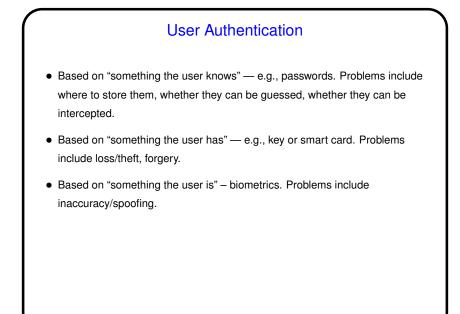
- MFT (master file table) analogous to i-nodes. One or more entries per file/directory, plus some for system. Boot block points to start.
- For small files, data is right in MFT record. Otherwise contains list of contiguous sequences of bytes.

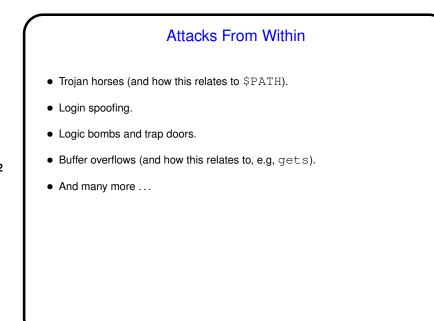


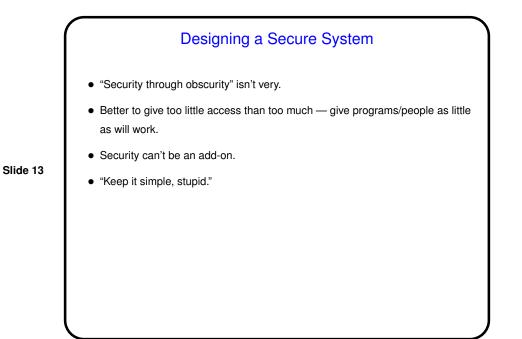












## Attacks From Outside Can categorize as viruses (programs that reproduce themselves when run) and worms (self-replicating) — similar ideas, though. Many, many ways such code can get invoked — when legit programs are run, at boot time, when file is opened by some applications ("macro viruses"), etc. Also many ways it can spread — once upon a time floppies were vector of choice, now networks or e-mail. Common factors: Executable content from untrustworthy source. Human factors. "Monoculture" makes it easier! Virus scanners can check all executables for known viruses (exact or fuzzy matches), but hard/impossible to do this perfectly. Better to try to avoid viruses — some nice advice on p. 633.

