8.6.3 File System Maintenance Facts

This lesson covers the commands used to check and maintain file system integrity:

File System Integrity Commands

Use the following commands to maintain file system integrity:

Command	Description	Examples
df	Displays the free space in the partition holding the specified directory. If no directory is given, the space available on all currently mounted file systems is shown. Disk space is shown in 1 K blocks by default. Common options include: • **-h** displays the output in get human readable format (bytes, KB, MB, GB, TB). • **-i** displays inode information. • **-l** limits the list to local file systems.	**df /home** Lists the free space on the partition that holds the **/home** directory.
du	Displays files and file sizes in and below a specified directory. Common options include: • **-c** lists a total amount of space used in the directory. • **-h** display the output in human readable format (bytes, KB, MB, GB, TB). • **-s** lists only the total, not each file. • **-a** evaluates all files, not just directories.	**du -c /home/badam** Lists all files and directories in badam's home directory along with a file size and a total amount of space taken up by the directory. **du -c -s /home/badam** Shows the total amount of space taken up in badam's home directory.
lsof	Displays open files in the file system. **lsof** gives the following information by default: • The command used to access the file • Process ID • Name of the user who is accessing the file	**lsof -u** *user* Lists files opened by processes that the specified user owns.

	• A file descriptor (these are described in the **lsof** man pages) • File node type • Device numbers • File size • Inode address • File path Common options include: • **+D [*directory_name*]** recursively lists files in a directory. • **-c [*command_name*]** lists all files for processes that are executing the specified command. • **-u [*user*]** lists open files owned by the specified user. • **-g [*process_ID*]** lists files opened by a specific process.	
fsck	Checks and optionally repairs one or more Linux file systems. Common options include: • **-s** serializes **fsck** when multiple file systems are checked. • **-t** specifies the type(s) of file system to be checked. • **-a** automatically repairs the file system without any questions. • **-r** prompts for confirmation when errors are found and ask permission to fix the errors (only when **-a** is not specified). Be aware of the following: • The file system must be unmounted before using **fsck**. • When manually running **fsck**, use runlevel 1 (init) or rescue.target (systemd) to ensure that other users do not mount the file system.	**fsck -t ext3 /dev/sdb1** Checks the first partition on the first partition of the second hard drive.
dumpe2fs	Prints super block and block information for an ext2, ext3, or ext4 file system. This includes information for each sector on the partition about sector type, block ranges, inode information, free blocks, and similar information. Command options include: • **-b** prints blocks reserved as bad in the file system. • **-h** prints only super block information. • **-x** prints group information block numbers in hexadecimal format.	**dumpe2fs /dev/sda1** Lists information for the first partition of the first hard drive.
tune2fs	Adjusts tunable file system parameters on ext2, ext3, and ext4 file systems. Some of the adjustable parameters include volume	**tune2fs -o acl /dev/sdb1** Enables access

	label, reserved blocks, inode sizes, and journaling. Tune2fs can also implement access control lists for individual users. Command options include: • **-c** adjust the number of mounts after which the file system will be checked. • **-e remount-ro** remounts the file system as read-only. • -**l** lists the contents of the file system super block. • **-o acl** enables Posix access control lists. • **-j** converts ext2 file systems to ext3 file systems.	control lists on the first partition of the second hard drive. The drive needs to be remounted.
debugfs	An ext2/ext3/ext4 file system debugger. Can be used for information gathering about target partitions, including directory listings with deleted file entries. Also allows file system modification and deleted file recovery. Command options include: • **-w** the file system should be opened in read-write mode. If not included, the file system will be read-only. • **-c** open the file system in catastrophic mode. This ignores inodes and group bitmaps initially. Useful when a file system has significant corruption. • **-f** *cmd_file* will read in commands from the *cmd_file* and execute them. • **-V** print the debugfs version number and exit.	**debugfs -w /dev/sdb1** Opens the file system on sdb1 in read-write mode. **debugfs -c /dev/sda1** Opens the file system on sda1 in catastrophic mode.
iostat	Monitors system I/O device loading by observing the time devices are active in relation to their average transfer rates. The iostat command generates reports that can be used to change system configuration to better balance the input/output load between physical disks. Running iostat without any options displays CPU usage and I/O statistics in the form of how much has been written per second and in total. Command options include: • **-m** displays the results in megabytes (MB) instead of kilobytes (KB). • **-d** only display the statistics for the devices connected on the system. • **-p** *device* display the results for the specified device. • **-x** adds extended statistics, such as avgqu-sz. This statistic shows the number of operations that were either queued or being serviced on a device. If this is not in the single digits (with an occasional double-digit spike) more troubleshooting may be required. • *number* When a number (such as 5) is used iostat will continue displaying statistics for that specified time in seconds. Press Ctrl + c to exit. See the man pages for additional options.	**iostat -m -p sda1** Lists results for the sda1 partition in MBs. **iostat 5** Refreshes the results every 5 seconds.

ioping	This tool generates various I/O patterns and lets you monitor I/O speed and latency in real time. this tool shows disk latency in the same way as ping command shows network latency on Linux or Unix-like system. Command options include: • **-c** *count device* runs for the number of specified *count* requests for the specified device • **-R** *device* shows the disk seek rate for the specified device See the man pages for additional options.	**ioping -c 10 /dev/sda** Performs a latency ping 10 times on the /dev/sda device. **ioping -R /dev/sda** Shows the disk seek rate for the /dev/sda device.
badblocks	A bad sector or block is a section on a disk drive to which data can no longer be written to read from. Included by most Linux distributions, **badblocks** is used to search for bad blocks on a device (usually a disk partition), where the device is the special file corresponding to the device (e.g. /dev/sda). Command options include: • **-b** *block-size* specifies the size of blocks in bytes. The default is 1024. • **-c** *number of blocks* is the number of blocks which are tested at a time. The default is 64. • **-e** *max bad block count* specifies a maximum number of bad blocks before aborting the test. The default is 0, meaning the test will continue until the end of the test range is reached. • **-i** *input_file* reads a list of already existing known bad blocks. Badblocks will skip testing these blocks since they are known to be bad. • **-n** uses non-destructive read-write mode. By default only a non-destructive read-only test is done. This option must not be combined with the -w option, as they are mutually exclusive. • **-o** *output_file* writes the list of bad blocks to the specified file. • **-s** shows the progress of the scan by writing out rough percentage completion of the current badblocks pass over the disk. • **-v** Verbose mode. • **-w** uses write-mode test. With this option, badblocks scans for bad blocks by writing some patterns (0xaa, 0x55, 0xff, 0x00) on every block of the device, reading every block and comparing the contents. This option may not be combined with the -n option, as they are mutually exclusive. • **-X** an internal flag to be used only by **e2fsck** and **mke2fs**. It bypasses the exclusive mode in-use device	**badblocks -v /dev/sda2 > badsectors.txt** Checks for bad blocks using the verbose mode and exports the results into a file named badsectors.txt

	safety check. **Warning** Never use the -w option on a device containing an existing file system. This option erases data! If you want to do write-mode testing on an existing file system, use the -n option instead. It is slower, but it will preserve your data. See the man pages for additional options.	
xfs_metadump	This command is used to copy the metadata (such as (filenames, file sizes) from an XFS files system to a file, but can only be used to copy unmounted file systems, or read-only mounted file systems. Be aware that by default, xfs_metadump obfuscates most file (regular file, directory and symbolic link) names and extended attribute names to allow the dumps to be sent without revealing confidential information. Command options include: • **-a** Copies entire metadata blocks. • **-e** Stops the dump on a read error. • **-g** Shows dump progress. • **-o** Disables obfuscation of file names and extended attributes. See the man pages for additional options.	**xfs_metadump -o /dev/sda3 /xfs/xfs_dump** Copies the file system metadata found on **/dev/sda3** to the **/xfs/xfs_dump** file. The **-o** option disables obfuscation of file names and extended attributes.