

# CSIS0230A Principles of Operating Systems(Class A)

## Tutorial 13

### Examining an ext2 filesystem

In this tutorial, we will use a utility called `lde` (Linux Disk Editor) to examine the contents of a disk partition, and tries to recover a deleted file in it. To be on the safe side, we will not use a real partition, but will instead use a regular file as a block device to build a filesystem.

#### 1. Preparation

Login as `root`, and create and populate a filesystem on the file `test.img` as follows:

1. Create a file `test.img` of size 40MiB, containing all zeros. This can be done by using the `dd` command, copying from the device file `/dev/zero` with a block size of 1M and a count of 40. See the man page of `dd` for details, in particular the `if`, `of`, `bs` and `count` options.
2. Setup a loopback block device `/dev/loop0` to use the file. This can be done using the command `losetup /dev/loop0 test.img`.
3. Use `mke2fs /dev/loop0` to create a filesystem there.
4. Create a directory `testmnt`, and mount the newly created filesystem to it using `mount /dev/loop0 testmnt`. Type `mount` to make sure that the filesystem is mounted.
5. Change into the `testmnt` directory, and type `tar xzvf ~/t13test.tgz` to extract `t13test.tgz` to your filesystem. Type `sync` to force all data to be written out.
6. Delete the file `lde/changelog` in the `testmnt` directory.
7. `cd` out of the `testmnt` directory and `umount` it.

#### 2. Starting and using lde

`Lde` (<http://lde.sourceforge.net>) is designed to recover deleted files, but it works well for investigating the filesystem as well. It is already installed in computers of our lab. To start `lde`, type `lde` followed by the name of the device that holds the filesystem (i.e., `/dev/loop0`). But due to a bug in the program, it does not work well under the `xterm` terminal type. So before running `lde`, type `TERM=linux`.

After a beginning message (which you should bypass by pressing a key), the program greets you with the **superblock view**, which display some summary information. Don't confuse it with the actual **superblock**, which contains all information needed by the filesystem to operate.

Other than the superblock view, `lde` has three other primary views: inode view, block view, and recovery view. You can switch among them by typing `s`, `i`, `b` and `r` respectively, and can exit the program from a primary view by typing `q` (be careful **not** to use it unless you want to quit!). The inode view and the block view display **unrelated** parts of the disk: the program maintains a current inode number and a current block number, which are shown in the title bar.

**The block view** shows a **hex-dump of a portion of the disk**. This way you can read arbitrary information within the filesystem. Within this view, you can move the cursor around by using page-up, page-down and the cursor keys, and when the cursor is move across a block boundary, the current block number is updated. If you want to jump to a particular block, you can type `#`, which allow you to type in an arbitrary block number (add `$` before the number if you want to input in hexadecimal).

**The inode view** should you the information stored within an inode. Arrow keys will bring you around the fields of the view. If you want to view another inode, you can use page-up and page-down, or you can type # and enter an inode number.

**The recovery view** can be used for recovering files. It displays a “fake inode” maintained by the program, to contain direct block and indirect block numbers like real inodes. In the recovery view, you can dump all these blocks to a file by typing `x`. In the inode or block view, if you find a block that you want to save, you can type a key corresponding to the characters displayed in the top-right corner of the title bar. This copies the block number to the fake inode. If you are in the inode view, you can copy all indices in the inode to the fake inode by typing `R`.

**The directory popup** shows a directory in an easy-to-read format. If you are in the inode or block view and the current block is a directory, you can show the directory popup by typing `d`. You can navigate the directories by using up and down arrows to select a directory and typing enter to switch to it, or you can type `q` to get back to the previous primary view.

**Shortcuts:** in the block view, inode view and directory popup, if the cursor is currently at a number, you can assign it to the current inode number and jump to the inode view by the `I` key, and you can assign it to the current block number and jump to the block view by the `B` key.

### 3. Your tasks

After familiarizing yourselves with `lde`, do the followings:

1. Read the superblock in block view, and compare it against the layout in the tutorial reading, to verify the information shown in the superblock view. Note that a “zone” in the superblock view means a data block.
2. Show the group descriptors of group 0 (i.e., first group) in the block view. By using the tutorial notes, decipher the group descriptor information.
3. Determine which block is used for the block bitmap of group 0, and show it in the block view. Determine which data blocks are used.
4. Repeat step (2) for the inode bitmap of group 0.
5. Find the inode table of group 0 in the block view. Where is the end of that inode table? (Hint: you need to know the number of inodes in each group, which can be found in the superblock, i.e., block 1.)
6. Go to the inode view and visit inode 2, the reserved inode for the root directory; and view its data in the block view. By using the layout shown in the tutorial reading, understand the data there. Verify your answer with the directory popup.
7. Find the inode for the the file `lde-2.6/UNERASE`. What blocks are used for its content? Using the block view, show the content of the first few blocks and the indirect block.
8. There is a symbolic link `/lde`, pointing to `/lde-2.6`. Show the inode of the symbolic link. How is the link target stored? (Hint: look at the block numbers in the inode.)
9. There is a hard link in the `/lde-2.6` directory. Can you find it? How it differs from the symbolic link?
10. Find the inode for the deleted `/lde-2.6/changelog` file. Try to recover it into a new file of the `/root` directory.