

Channel 16 - Manual

Joseph Kuan*

November 6, 2001

*Contact: e.xom@ntlworld.com

Contents

1	Introduction	3
2	GUI tutorial	3
2.1	Delete inodes list dialog	3
2.2	Browse inode content dialog	4
2.3	Browsed inodes list dialog	5
3	TroubleShooting	6

1 Introduction

There may be a time when you have accidentally deleted critical file(s) that do not have backup or the backup is too old. In this case, you can undelete the file with debugfs provided by e2fsprogs. However, you need to gather clues like the time when the file was deleted and the size of the file, etc. If you do not have this information stored anywhere, then you may need to search for all recent deleted inodes manually. This is why Channel 16 provides such interface to make the recovery easier.

Channel 16 is a graphical tool for recovering deleted files under ext2 filesystem. It uses the methods provided by e2fsprogs package. This application provides the following features:

- Browsing interactively for each block inside a deleted inode, allowing the user to peep at the blocks before deciding whether it is the right inode.
- Search & Rescue operation (in the next release) - user can type in content of the deleted file and the program will automatically search for each block in all the deleted inodes.

This graphical tool is implemented in Java Swing, then uses JNI to interface with the libext2fs library.

This document gives a general introduction on how to use this application. By using this application, I do not hold any responsibility for losing any critical files nor corrupting the filesystem.

2 GUI tutorial

In the first step before launching the program, the ext2fs partition which contains the deleted file must be remounted to read-only. Please refer to HOWTO/min/Ext2fs-Undeletion for details.

To launch channel16, you must login as root and do the following:

```
channel16 -d device
```

Device is the device file containing the deleted inodes (e.g. /dev/hda1). Diagram 1 shows the front end of channel16.

2.1 Delete inodes list dialog

The user can click on the FileSystem menu, then select "Deleted inodes" menu item.

This operation will probably take a while if the size of the partition is large (A progress bar will be implemented in the future release). The long process only occurs when the filesystem is scanned for the first time and stored in the



Figure 1: Front end of channel 16

memory. The same dialog can be launched instantly in the same application session. Once the scan for deleted inodes is completed, a dialog appears with the inode deleted time, along with the user login name that the inode belongs to and the size of such inode (in bytes).

Figure 3 demonstrates a dialog of all the deleted inodes.

The inodes list is displayed in descending order of delete time, hence the first entry is the most recent deleted inode. When the user selects an inode, the "Delete time:" label is updated corresponding to the delete time of the selected inode. The user login name is shown rather than the user id. If there is no user login name associated with the inode's user id, the user id is displayed in the User column instead. The size column indicates the total size of all the data blocks belonging to the inode.

2.2 Browse inode content dialog

In order to browse the content of an inode, the user can select the inode and then click on the "Browse" button (see screenshot 3). Then a dialog appears for browsing data blocks for that inode. Note that multiples of this dialog can

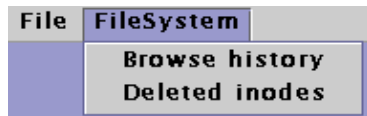


Figure 2: Menu items for filesystem

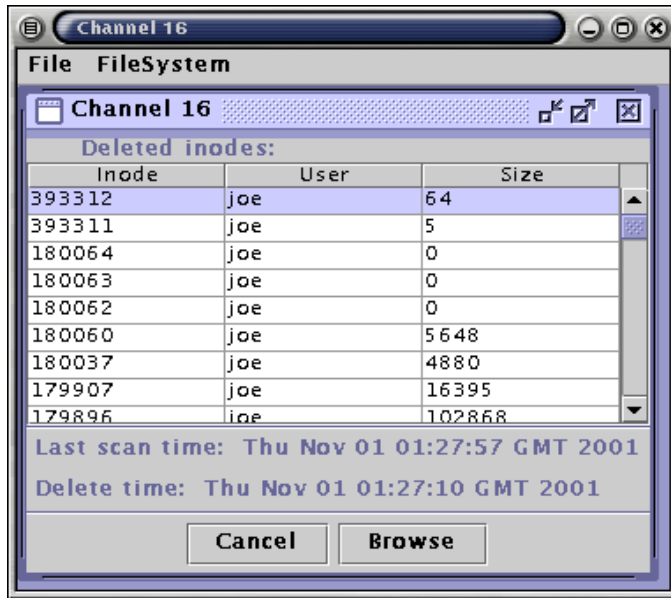


Figure 3: List all the deleted inodes

be opened for different inodes.

The browse inode dialog (see screenshot 4) allows the users to scroll forward and backwards in blocks. At the top, the left and right arrow buttons are to scan the block data forward and backward respectively. The entry inbetween the two buttons are to scan the number of blocks in either directions. The text area displays the content in the blocks. If the data inside the blocks is binary, the area displays hex values of the data. Underneath the text box, it indicates the number of block data currently shown in the text area and also the total number of bytes of data shown (this also corresponds to binary data despite hexadecimal presentation). At the bottom of the dialog, the "All blocks" button gets all the block data belonging to the inode. The "Save as" button launches a file choose dialog to save the deleted inode content into a new file. Note that the save operation only saves the block data currently shown in the text area. In order to save all the data of an inode, the user must scan for all the blocks first which are shown in the text area.

2.3 Browsed inodes list dialog

As inodes are identified as unique numbers, the user may find it difficult to remember which inodes have been looked at during the application session. So the browsed inodes list (see screenshot 5) dialog serves for such purposes that

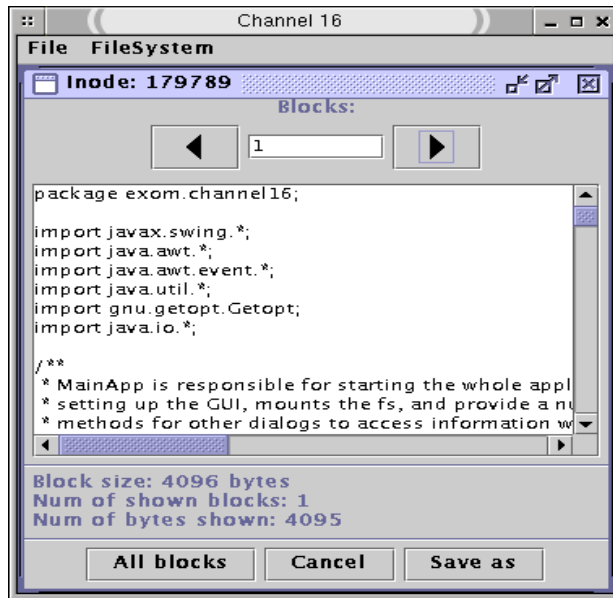


Figure 4: Dialog for scanning data blocks of an inode

records the inode number from an inode content dialog (see figure 4).

The dialog is invoked from the "Browse history" menu item, shown in figure 2.

3 Troubleshooting

To be documented or email me any problems.

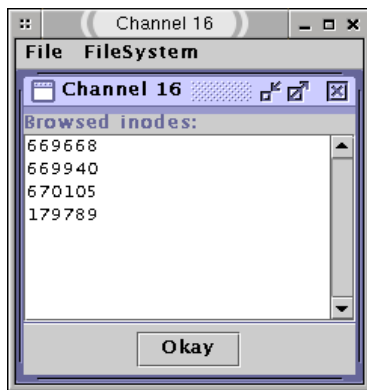


Figure 5: List of all the browsed inodes