FSLINT

Published : 2011-04-22
License : None

1. FSLINT

FSlint is a utility to find and clean various forms of unwanted extraneous files on a computer file system. This excess of unnecessary files is referred to as lint. FSlint will help find unwanted or problematic lint in your files or file names. The most common forms of lint FSlint finds are duplicate files, empty directories, and improper names. FSlint has multiple tools to perform a multitude of tasks in both the graphical interface and command line modes.

It is the goal of this guide to walk you through each of the major functions and tools that FSlint provides within the graphical interface. As the graphical interface is simply a front end for the command line, this guide refers to the help page of each of the command line tools as they apply to the graphical tool set described in each chapter. The reader is encouraged to read through the command line help page as it often includes more insight as well as advanced methods used to fine tune the results. Even though you may not need to use the command line tools, the advanced information may provide a better understanding of how to use the graphical interface. Understanding the command line will also help those who wish to use the tools in more advanced settings, such as scripts or remote servers.

The reader is highly encouraged to have a good backup of their system before deleting or modifying any files. You are also encouraged to fully understand the tasks you FSlint to perform. FSlint is a very powerful tool, and it would be unfortunate to make a mistake and lose important data without having a backup.

FSlint can be found in the online repositories of several major Linux releases, and it can be easily installed through the package manager. This guide covers FSlint version 2.43 which is the latest release at the time of this writing. This guide was written and tested on Debian Lenny with further testing done on Ubuntu 10.04. Detailed information on how to install FSlint for a variety of distributions is kept up-to-date on FSlint's homepage.

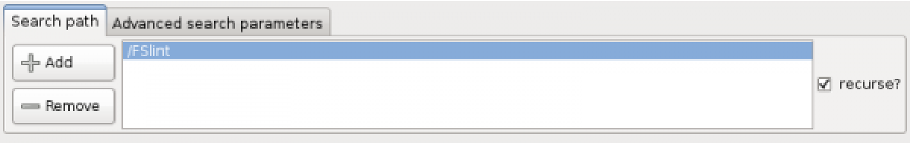
Homepage: http://www.pixelbeat.org/fslint/

2. COMMON INTERFACE ITEMS

As the reader progresses through this manual on FSlint, you will repeatedly see common buttons and interfaces that you should be able to recognize. Not every function uses every interface option, so it is important to understand how these buttons and interfaces work . This will make it easier to understand their importance at various times. There are several tabs and buttons used to access each of the tools available. It is the goal of this guide to explain each of the tabs and tools as they relate to the graphical interface.

FSlint allows for more advanced and powerful filtration through the use of regular expressions and wildcards. An example of a regular expression would be using *.jpg to select all pictures of the JPG file type. You are encouraged to research regular expressions further as they are only briefly covered in this manual.

SEARCH PATH TAB

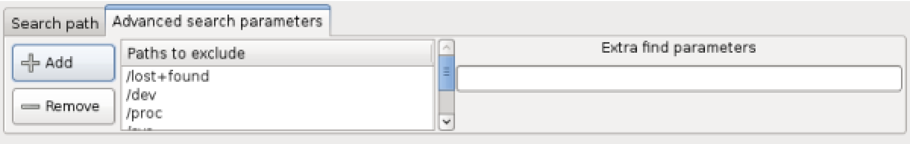


/FSlint

☑ recurse?

The first thing to notice when starting FSlint is the 'Search path' tab. This tab that allows a user to add or remove one or multiple directory paths to be searched. By default FSlint will search the directory from which it is launched. This is typically /home/<USER> if you start FSlint from a menu icon. The check box labelled 'recurse?' on the left hand side of the screen determines the depth of the search for duplicate files. If left unchecked, it will only search the directory listed. When checked, it will descend into every folder below.

ADVANCED SEARCH PARAMETERS TAB



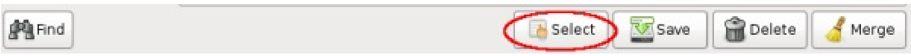
Paths to exclude
/lost+found
/dev
/proc

Extra find parameters

The 'Advanced search parameters' tab allows you to filter out certain file types or directories. 'Paths to exclude' allows for not only paths, but also file types to be excluded from the search. FSlint, by default, will exclude several folders and file types that should not be managed by FSlint. You are free to add or remove items with the 'Add' and 'Remove' buttons.

The 'Extra find parameters' allows for a more refined search using the parameters of the 'find' Linux command. For example, to search for files that only belong to the user Fred the reader could add "-user `id -u fred`". The detailed use of the 'find' command is beyond this guide at this time.

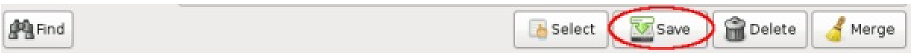
SELECT BUTTON



The 'Select' button will bring up the same options as right-clicking in the results window. This menu allows you to quickly sort through the duplicate files found for further manipulation. 'Select using wildcard' will take a regular expression and apply it to the file names in the results window. When there are already several files selected, 'Unselect using wildcard' will take a regular expression and leave selected only the files that were left unmatched. There are three subsections to the 'Within groups'.

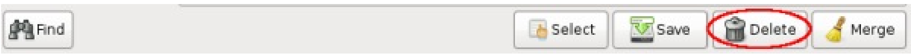
1. "Select all but first" will highlight all files within a group except for the first entry.
2. "Select all but newest" will highlight all files within a group except for the most recent.
3. "Select all but oldest" will highlight all files within a group except for the very oldest.

SAVE BUTTON



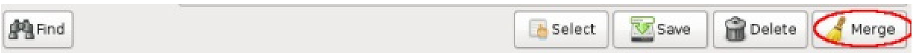
After you have a selection of files, you can save this list with the 'Save' button. This is useful if you plan on doing more advanced tasks like passing the information on to a script for further analysis, or if you just want a record of the files before merging or deleting them. The save button will bring up a menu for you to select the location where you would like to store the output. The saved data will contain a full path for each of the files selected.

DELETE BUTTON



After selecting a file, or many files, you can delete them with the 'Delete' button. A confirmation window will appear each time this is done unless the 'Ask me this in the future?' checkbox is unchecked. **This will permanently delete the selected files.**

MERGE BUTTON

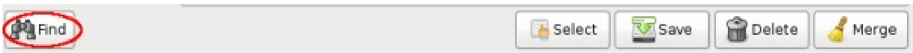


The Merge button will merge all the files within a group into one physical file using hardlinks, or symlinks if spanning across file systems. You can merge all files by ensuring no file is selected (right click -> Unselect all), or you can exclude files from merging by selecting them. Merging leaves the files on the system, but frees up hard drive space by using the command 'ln' to hard link the files.

A hard linked file is a single file on the disk that may appear to reside in two different locations. Suppose there are two folders, test1 and test2. Inside of test1 there are two files A and B. Inside of test2 there are two files B and C. If B takes up a considerable amount of hard drive space but is needed in both folders and is the exact same file, then this option is very useful as it will merge the B file with hardlinks. Only one copy of B will exist on the hard drive, but B will still appear and function as needed in both folders. More information on hard and soft links can be found by reading about the utility 'ln'.

If you are trying to remove excess files and B is not needed in both folders, the delete option would be a better choice. It is left up to you as to which copy of B to remove.

FIND BUTTON



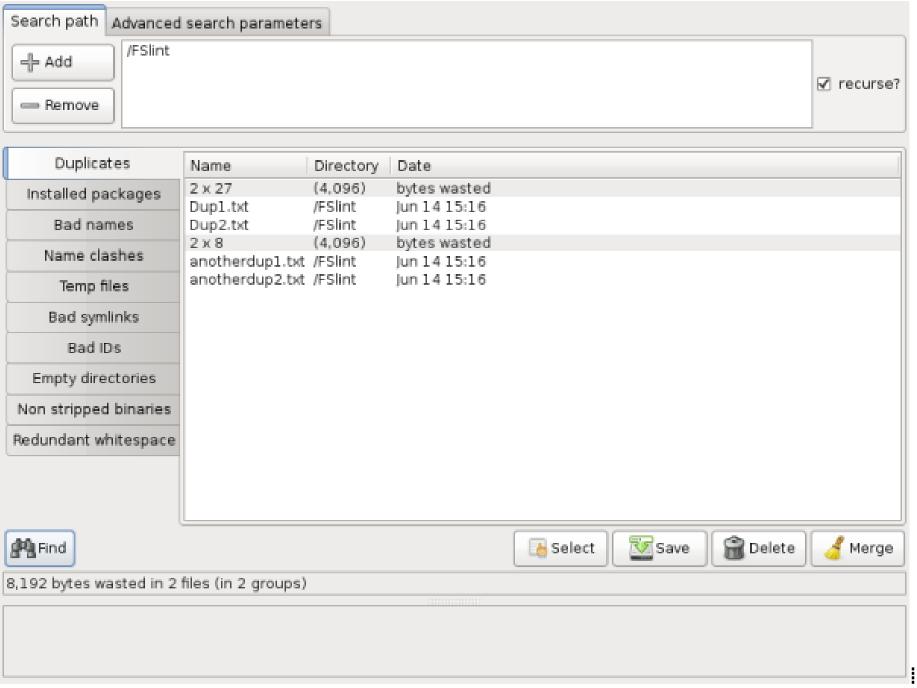
After you have chosen the 'Search path' and selected the type of search you wish to perform, the find button tells FSlint to perform the selected actions.

Using FSLint
3. Duplicates
4. Installed Packages
5. Bad Names
6. Name Clashes
7. Temp Files
8. Bad Symlinks
9. Bad ID's
10. Empty Directories
11. Non Stripped Binaries
12. Redundant Whitespaces

3. DUPLICATES

One of the most commonly used features of FSlint is the ability to find duplicate files. The easiest way to remove lint from a hard drive is to discard any duplicate files that may exist. Often a computer user may not know that they have four, five, or more copies of the exact same song in their music collection under different names or directories. Any file type whether it be music, photos, or work documents can easily be copied and replicated on your computer. As the duplicates are collected, they eat away at the available hard drive space. The first menu option offered by FSlint allows you to find and remove these duplicate files.

GRAPHICAL INTERFACE



/FSlint

☑ recurse?

Name	Directory	Date
2 x 27	(4,096)	bytes wasted
Dup1.txt	/FSlint	Jun 14 15:16
Dup2.txt	/FSlint	Jun 14 15:16
2 x 8	(4,096)	bytes wasted
anotherdup1.txt	/FSlint	Jun 14 15:16
anotherdup2.txt	/FSlint	Jun 14 15:16

8,192 bytes wasted in 2 files (in 2 groups)

The 'Duplicates' tab on the left hand side of the screen is the default tab selected at FSlint start up. The algorithm used to determine if a file is a duplicate of another is very thorough to minimize any possible false positives that may lead to data loss. FSlint scans the files and filters out files of different sizes. Any remaining files of the exact same size are then checked to ensure they are not hard linked. A hard linked file could have been created on a previous search should the user have chosen to 'Merge' the findings. Once FSlint is sure the file is not hard linked, it checks various signatures of the file using md5sum. To guard against md5sum collisions, FSlint will re-check signatures of any remaining files using sha1sum checks.

The 'Duplicates' interface is very simple. After the user has verified the 'Search path' location that they wish to search, they can simply click the 'Find' button on the lower left of the screen. When the process has finished the results of the found duplicate files are displayed in the central portion of the screen. All of the duplicate files will be grouped together under a grey bar giving information such as how many files are in the group and the number of bytes wasted in duplicate files. The files themselves are listed below the grey divider by their name, directory, and last modification date. Listed directly below the 'Find' button is the total number of bytes wasted in the total number of files and total number of groups.

COMMAND LINE INTERFACE

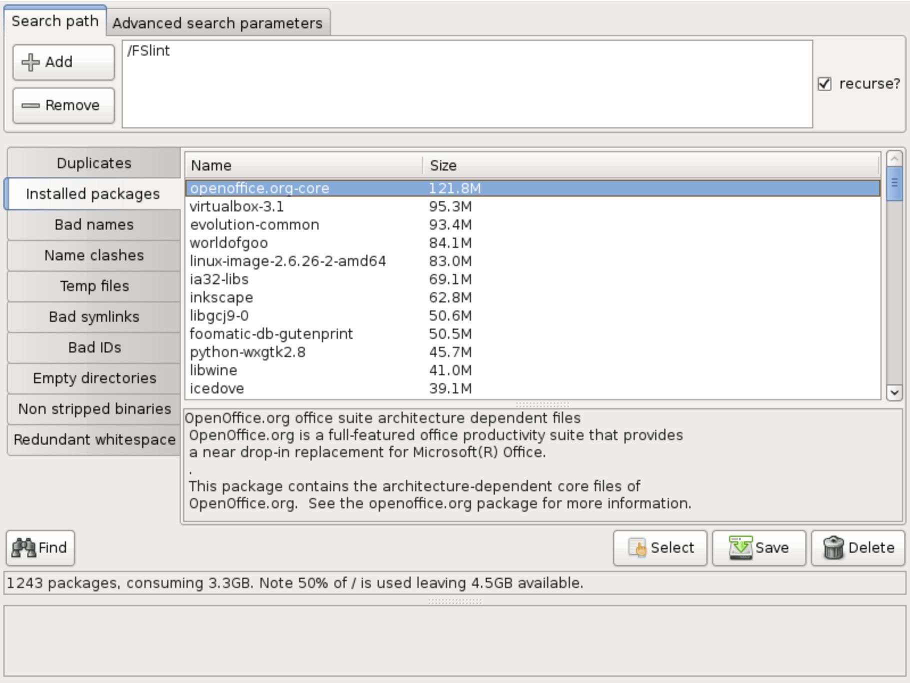
The command line interface to this utility is 'findup'. This utility will be found in the installation directory of FSlint.

```
$/usr/share/fslint/fslint/findup --help
find dUPlicate files.
Usage: findup [[[-t [-m|-d]] | [--summary]] [-r] [-f] paths(s) ...]

If no path(s) specified then the current directory is assumed.

When -m is specified any found duplicates will be merged (using hardlinks).
When -d is specified any found duplicates will be deleted (leaving just 1).
When -t is specfied, only report what -m or -d would do.
When --summary is specified change output format to include file sizes.
You can also pipe this summary format to /usr/share/fslint/fslint/fstool/dupwaste
to get a total of the wastage due to duplicates.

Examples:
search for duplicates in current directory and below
    findup or findup .
search for duplicates in current directory and below listing the files full path
    findup -f
search for duplicates in all linux source directories and merge using hardlinks
    findup -m /usr/src/linux*
same as above but don't look in subdirectories
    findup -r .
search for duplicates in /usr/bin
    findup /usr/bin
search in multiple directories but not their subdirectories
    findup -r /usr/bin /bin /usr/sbin /sbin
search for duplicates in $PATH
    findup `/usr/share/fslint/fslint/supprt/getffp`
search system for duplicate files over 100K in size
    findup / -size +100k
search only my files (that I own and are in my home dir)
    findup ~ -user `id -u`
search system for duplicate files belonging to roger
    findup / -user `id -u roger`
```

4. INSTALLED PACKAGES

Over time any computer user will install countless programs that are not used or needed. Whether the package was installed on accident, to test some functionality, or installed and simply forgotten, it still eats away at hard drive space. Should you find yourself low in hard drive space in the root partition, it may help to locate the installed packages that are taking up the greatest amount of space.

GRAPHICAL INTERFACE



/FSlint

☑ recurse?

Name	Size
openoffice.org-core	121.8M
virtualbox-3.1	95.3M
evolution-common	93.4M
worldofgoo	84.1M
linux-image-2.6.26-2-amd64	83.0M
ia32-libs	69.1M
inkscape	62.8M
libgcj9-0	50.6M
foomatic-db-gutenprint	50.5M
python-wxgtk2.8	45.7M
libwine	41.0M
icedove	39.1M

```
OpenOffice.org office suite architecture dependent files
 OpenOffice.org is a full-featured office productivity suite that provides
 a near drop-in replacement for Microsoft(R) Office.
 .
 This package contains the architecture-dependent core files of
 OpenOffice.org.  See the openoffice.org package for more information.
```

1243 packages, consuming 3.3GB. Note 50% of / is used leaving 4.5GB available.

The 'Installed packages' tool will find all of the packages installed and organize them by the total amount of disk space used. It supports dpkg (e.g. Debian, Ubuntu), rpm (e.g. Red Hat, Fedora), and packman (e.g. Arch) based systems. When a package is selected the description of the package can be found in the grey dialog box below the selection screen.

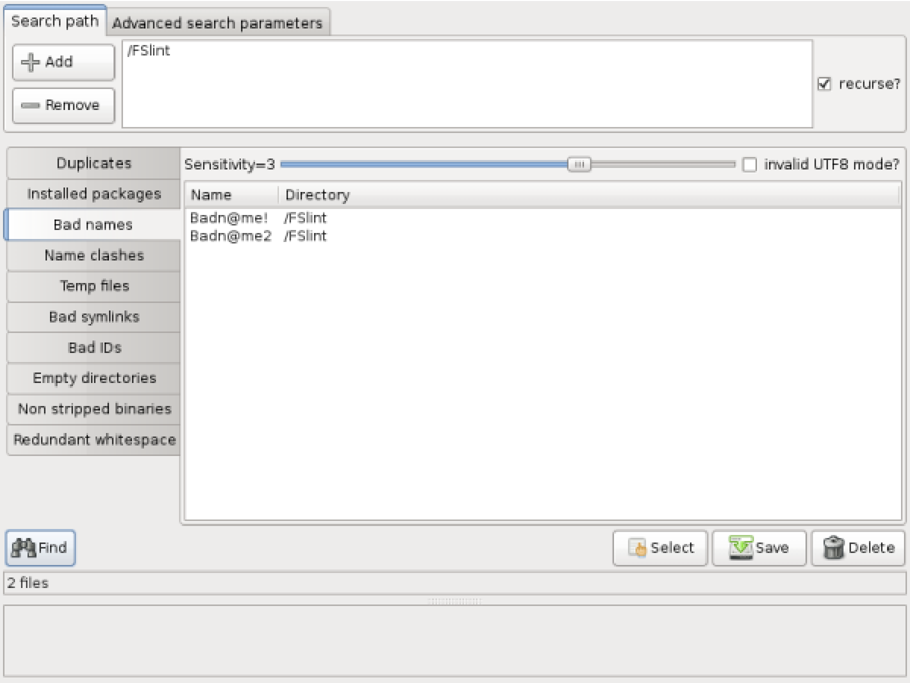
COMMAND LINE INTERFACE

FSlint does not provide a command line tool for finding installed packages. The graphical interface simply calls out to the package management on your computer system for the information. Please consult your package manager for more details.

BAD NAMES

Another form of file lint is bad file names. These files may not eat away at hard drive space, but they may be difficult to use or move. The common user will occasionally encounter bad names when they try to copy their music collection from their computer to their portable device. This can result in odd warnings about a file having an invalid name, and the transfer fails. Typically this is found when the file was titled in a language other than the user's native language. While bad names are a common annoyance, it is not required that the user fix them (though it is recommended).

GRAPHICAL INTERFACE



/FSlint

☑ recurse?

Sensitivity=3

□ invalid UTF8 mode?

Name	Directory
Badn@me!	/FSlint
Badn@me2	/FSlint

2 files

'Bad names' searches out all files and inspects their naming conventions. The slider at the top of the section allows the user to set the sensitivity level. Level one is the least strict while level four is a strict POSIX check. The check box to the right of the slide bar allows the user to select UTF8 checking. The color code is simply the color given by the 'ls' command on the users system. Please consult the OS help pages for more information on UTF8, POSIX, and 'ls'.

COMMAND LINE INTERFACE

There are two utilities provided by FSlint for command line checking. The first is 'findnl' to find bad names. The second is 'findu8' to find names with bad UTF8 encoding.

findnl

```
$ /usr/share/fslint/fslint/findnl --help
 find Name (directory or file) Lint.
 Usage: findnl [-1] [-2] [-3] [-p] [[-r] [-f] paths(s) ...]

 These options are mutually exclusive (i.e. only the last one takes effect).
 -1 is least checking, -3 is most. The default is 2.

 -p is most stringent and applies POSIX.1 filename portability testing.
 I.E. characters are limited to [A-Za-z0-9_.-] and max name length = 14 and
 max path length = 255.
 If no path(s) specified then the current directory is assumed.
```

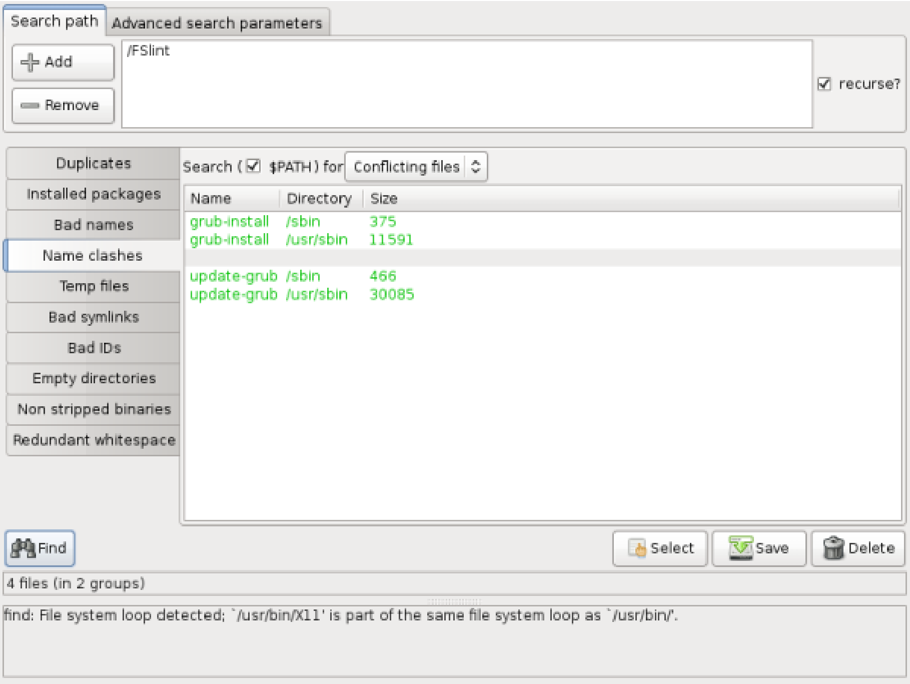
findu8

```
$ /usr/share/fslint/fslint/findu8 --help
 find names with invalid UTF8 encoding.
 Usage: findu8 [[-r] [-f] paths(s) ...]

 If no path(s) specified then the current directory is assumed.
```

6. NAME CLASHES

Another form of file lint involves files that have identical or similar names. This usually does not cause any significant problems for the user other than slight inconvenience. Finding files with name clashes can be of great help when dealing with multiple versions of files. It is not uncommon to find a school or work document in several locations, each at a different revision stage of the document. When all of these are found and brought to your attention, then you may decide to delete all but the final version of the document.

GRAPHICAL INTERFACE



/FSlint

☑ recurse?

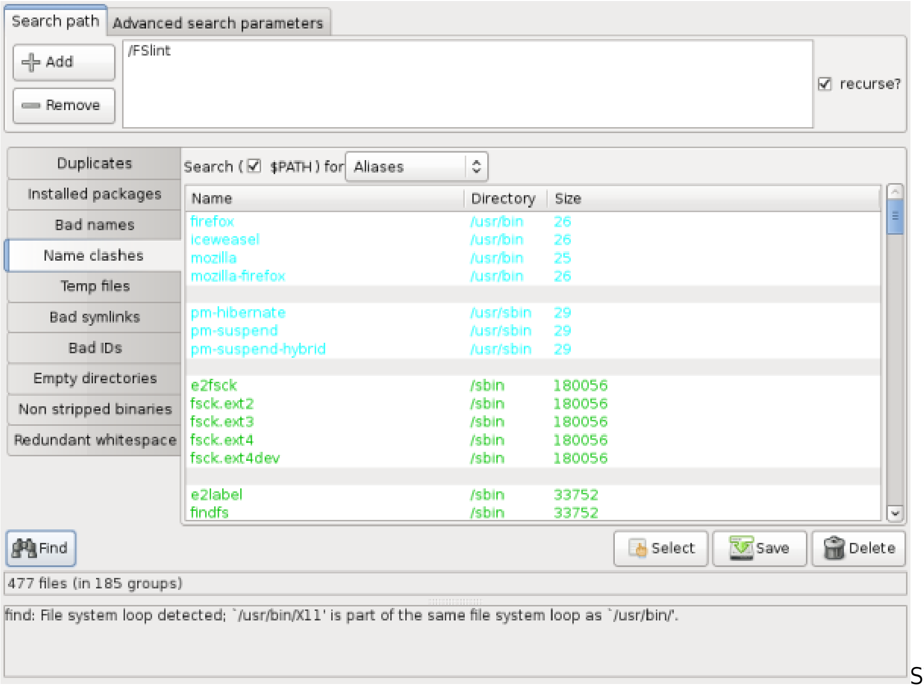
Search (☑ $PATH) for Conflicting files

```
Name          Directory   Size
grub-install  /sbin       375
grub-install  /usr/sbin   11591

update-grub   /sbin       466
update-grub   /usr/sbin   30085
```

4 files (in 2 groups)

```
find: File system loop detected; `/usr/bin/X11' is part of the same file system loop as `/usr/bin/'.
```

When the 'Search $PATH' check box is enabled and the 'Conflicting files' drop box is selected, FSlint will search the user's $PATH for files that share a common name. Each Linux system has $PATH setting. The $PATH allows programs to find each other and the $PATH should only be modified by advanced users. Be extremely careful when searching the system $PATH as required system settings may be listed, and damage to the system can occur if these system files are deleted or modified.



/FSlint

☑ recurse?

Search (☑ $PATH) for Aliases

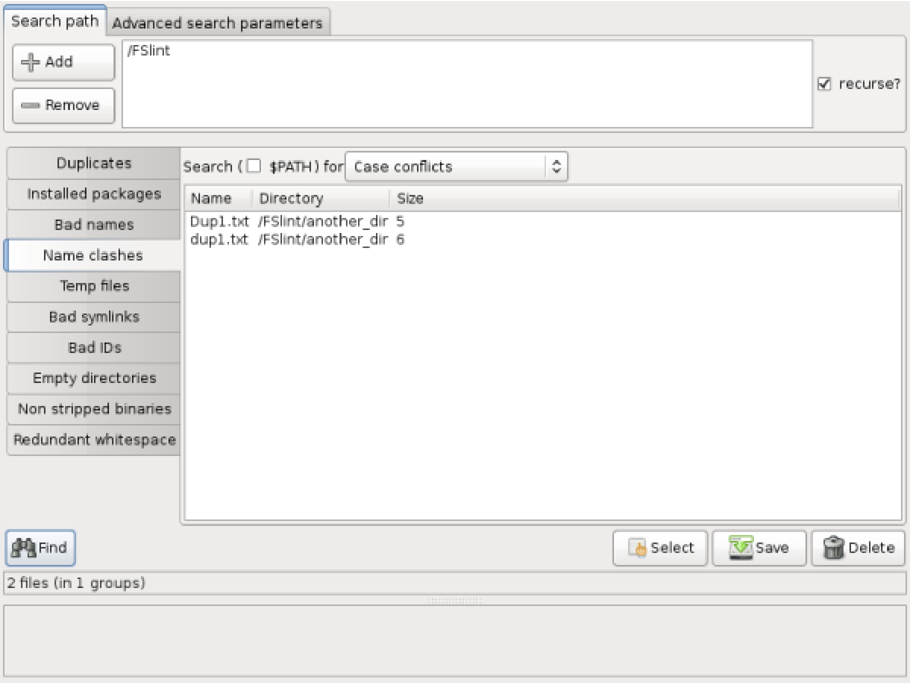
Name	Directory	Size
firefox	/usr/bin	26
iceweasel	/usr/bin	26
mozilla	/usr/bin	25
mozilla-firefox	/usr/bin	26
pm-hibernate	/usr/sbin	29
pm-suspend	/usr/sbin	29
pm-suspend-hybrid	/usr/sbin	29
e2fsck	/sbin	180056
fsck.ext2	/sbin	180056
fsck.ext3	/sbin	180056
fsck.ext4	/sbin	180056
fsck.ext4dev	/sbin	180056
e2label	/sbin	33752
findfs	/sbin	33752

477 files (in 185 groups)

```
find: File system loop detected; `/usr/bin/X11' is part of the same file system loop as `/usr/bin/'.
```

Similarly when the "Aliases" drop box is selected FSlint will search for any aliases that are in the user's $PATH. Aliases can be thought of as shortcuts to various Linux programs. Once again, be aware that damage to the system can occur if extreme care is not taken to ensure the proper items are deleted.

When the "Search $PATH" checkbox is not selected, there are four options in the drop box, and FSlint will search the path given in the "Search path" tab.



/FSlint

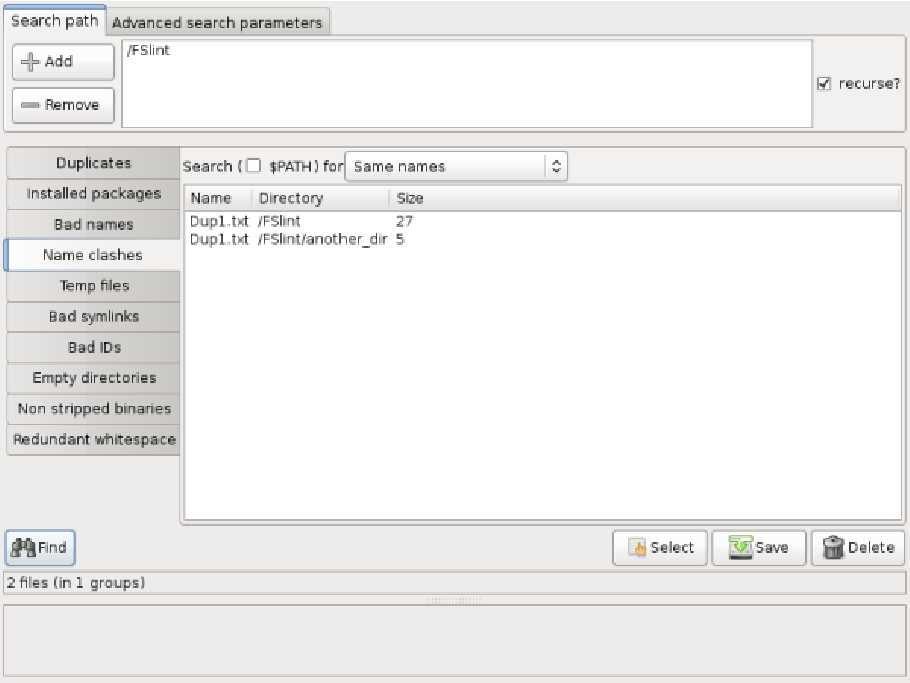
☑ recurse?

Search (□ $PATH) for Case conflicts

Name	Directory	Size
Dup1.txt	/FSlint/another_dir	5
dup1.txt	/FSlint/another_dir	6

2 files (in 1 groups)

The first is "Case conflicts". FSlint will look for files that share the same name differing only in capitalization.



/FSlint

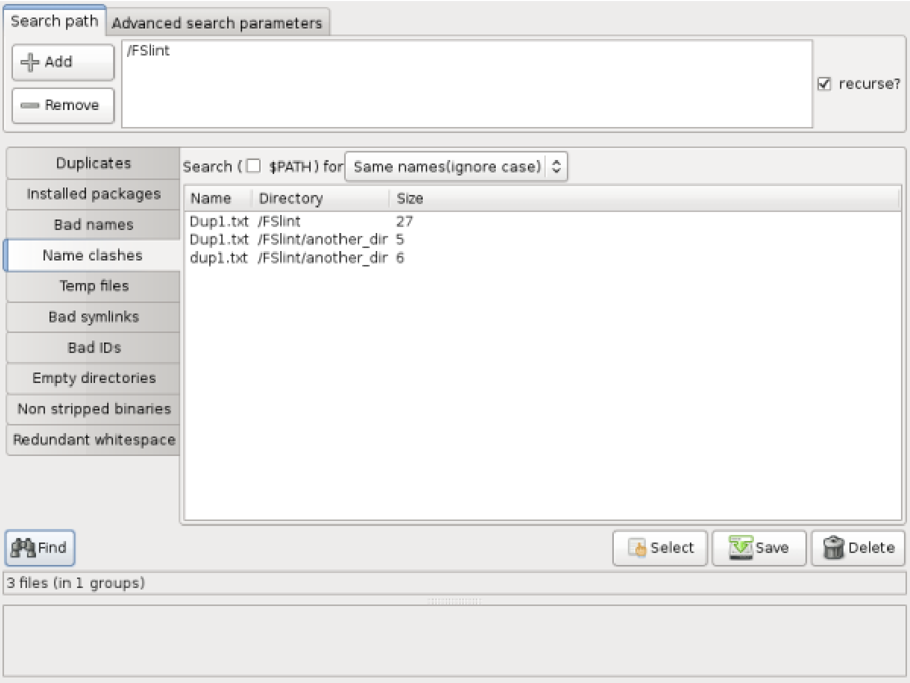
☑ recurse?

Search (□ $PATH) for Same names

Name	Directory	Size
Dup1.txt	/FSlint	27
Dup1.txt	/FSlint/another_dir	5

2 files (in 1 groups)

The second is "Same names". FSlint will look for files that share the exact same name, but are located in different directories within the search parameters.



/FSlint

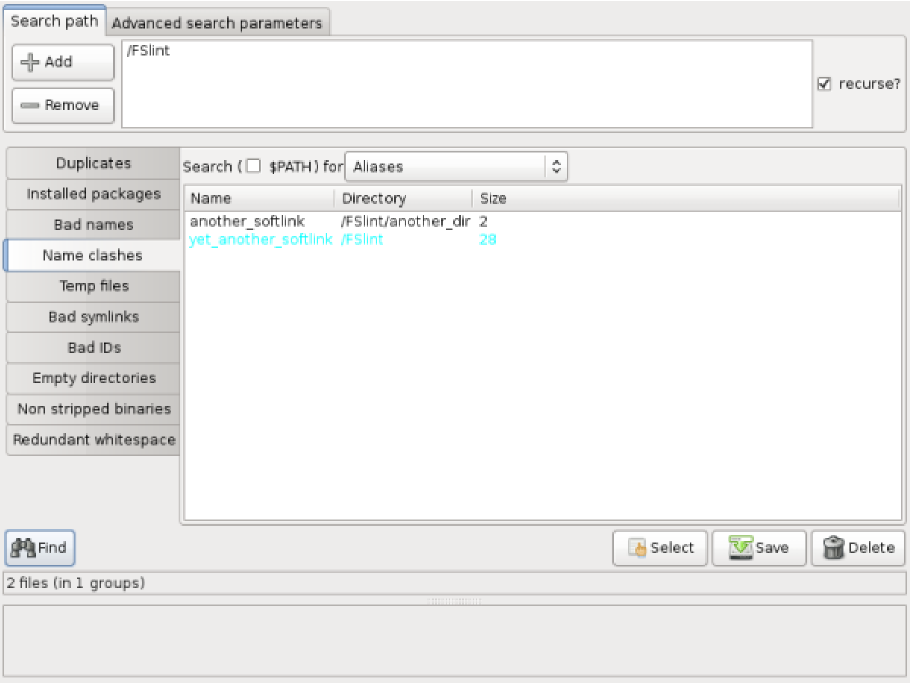
☑ recurse?

Search (□ $PATH) for Same names(ignore case)

Name	Directory	Size
Dup1.txt	/FSlint	27
Dup1.txt	/FSlint/another_dir	5
dup1.txt	/FSlint/another_dir	6

3 files (in 1 groups)

The third is 'Ignore case'. FSlint will look for all files in all directories and sub directories that share the same name regardless of capitalization.



/FSlint

☑ recurse?

Search (□ $PATH) for Aliases

Name	Directory	Size
another_softlink	/FSlint/another_dir	2
yet_another_softlink	/FSlint	28

2 files (in 1 groups)

The last is "Aliases". FSlint will look for all aliases, soft or hard linked.

COMMAND LINE INTERFACE

The command line interface to this utility is 'findsn'. This utility will be found in the installation directory of FSlint.

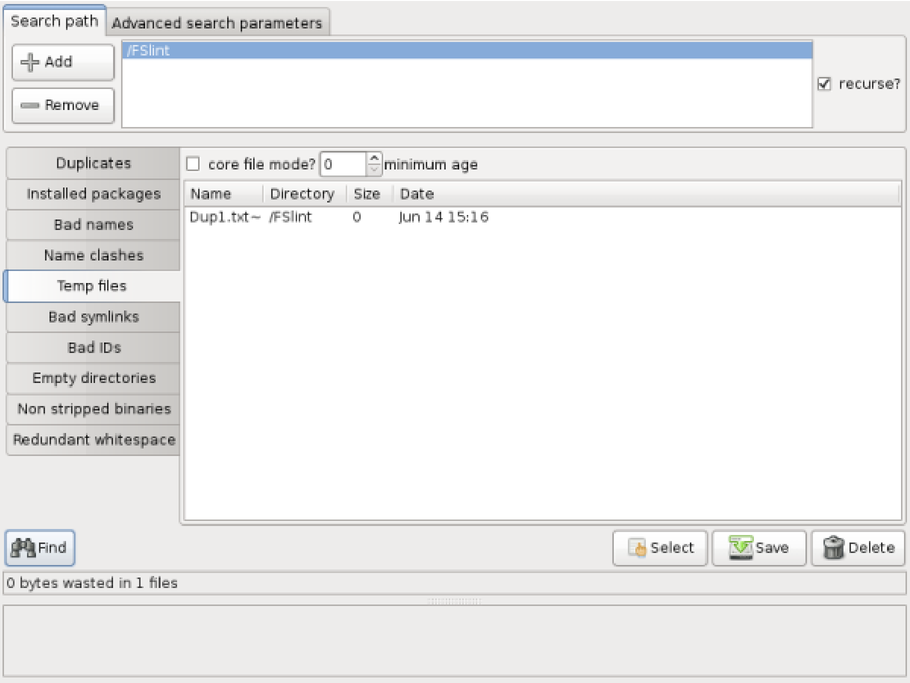
```
$ /usr/share/fslint/fslint/findsn --help

 find (files) with duplicate or conflicting names.
```

```
Usage: findsn [-A -c -C] [[-r] [-f] paths(s) ...]

If no arguments are supplied the $PATH is searched for any redundant
or conflicting files.

-A reports all aliases (soft and hard links) to files.
If no path(s) specified then the $PATH is searched.

If only path(s) specified then they are checked for duplicate named
files. You can qualify this with -C to ignore case in this search.
Qualifying with -c is more restrictive as only files (or directories)
in the same directory whose names differ only in case are reported.
I.E. -c will flag files & directories that will conflict if transfered
to a case insensitive file system. Note if -c or -C specified and
no path(s) specifed the current directory is assumed.
```

7. TEMP FILES

Temporary files can eat away at a your hard disk space without you ever noticing. Temporary files are typically recognized and hidden from the user's sight by the OS. Temporary files can be created when editing files, when running some programs, or when a program wishes to report and save a problem. Removing these temporary files can free up valuable space, and FSlint can find these files so that they can be purged.

GRAPHICAL INTERFACE



/FSlint

☑ recurse?

☐ core file mode? 0 minimum age

Name	Directory	Size	Date
Dup1.txt~	/FSlint	0	Jun 14 15:16

0 bytes wasted in 1 files

It is possible to specify a minimum age to find only files modified in the last X number of days. A minimum age of 0 is all temporary files.

When a program exits or is terminated prematurely, the program can leave behind files known as core files or dump files for debugging purposes. The core files are generally named "core.#####" where # can represent any number 0-9. Many Linux distributions have this feature disabled. To check if core files are enabled on your Linux system, run `ulimit -c` from the command line. A value of 0 is disabled. In FSlint the check box "core file mode?" enables a more thorough search for these core files.

COMMAND LINE INTERFACE

The command line interface to this utility is 'findtf'. This utility will be found in the installation directory of fslint.

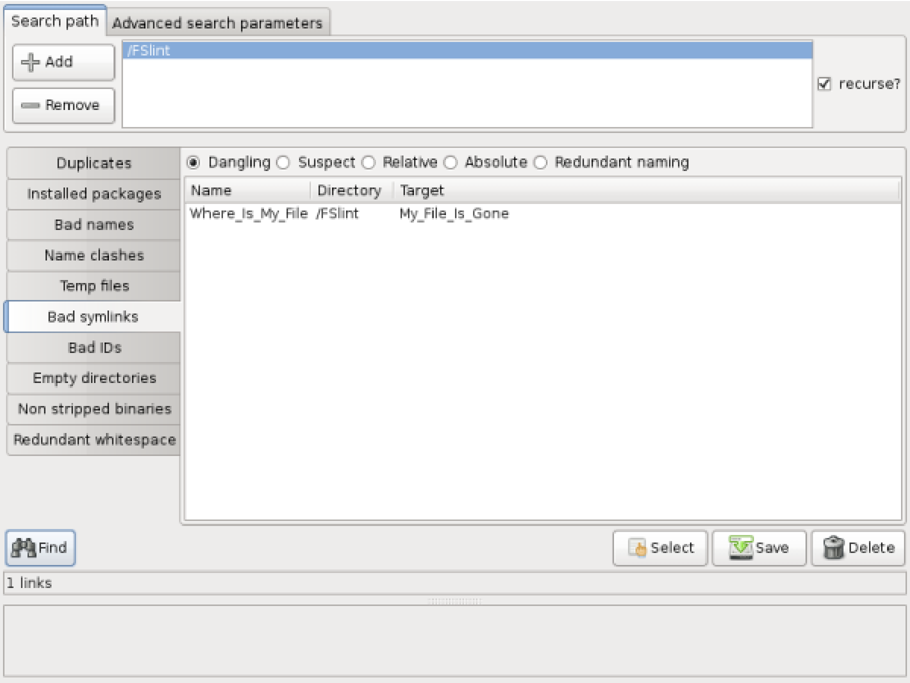
```
$ /usr/share/fslint/fslint/findtf --help
 find Temporary Files.
 Usage: findtf [-c] [--age=days] [[-r] [-f] paths(s) ...]

 If -c specified then a more thorough search is done for
```

```
(and only for) core files, and more information is printed about them.

If --age=days specified then the temporary files must be over that
number of "days" old before being reported.
For e.g. findtf -c --age=4 only reports core files over 4 days old.

If no path(s) specified then the current directory is assumed.
```

8. BAD SYMLINKS

There are two main types of symlinks, hard and soft. Hard symlinks allow a single file to appear as if it resides in multiple locations, and soft symlinks simply point to a location. Should that location change the soft symlink will no longer point to the file, and unless it is updated it will become a bad symlink. Bad symlinks are most commonly found as a result of programs being installed and un-installed. One example is when a user finds that some of their files work as expected in one location but fail when moved or a directory is renamed. This is usually due to problems caused by bad symlinks. Symlinks can be extremely useful, but they can potentially cause great frustration for the user. FSlint will look for symlinks that have some kind of problem. Not all symlinks reported need to be fixed as the circumstances for their inception can be radically different depending on the user and the purpose. FSlint will simply report the symlinks it finds during its search. Specifically, it will look for five different types.

GRAPHICAL INTERFACE



/FSlint

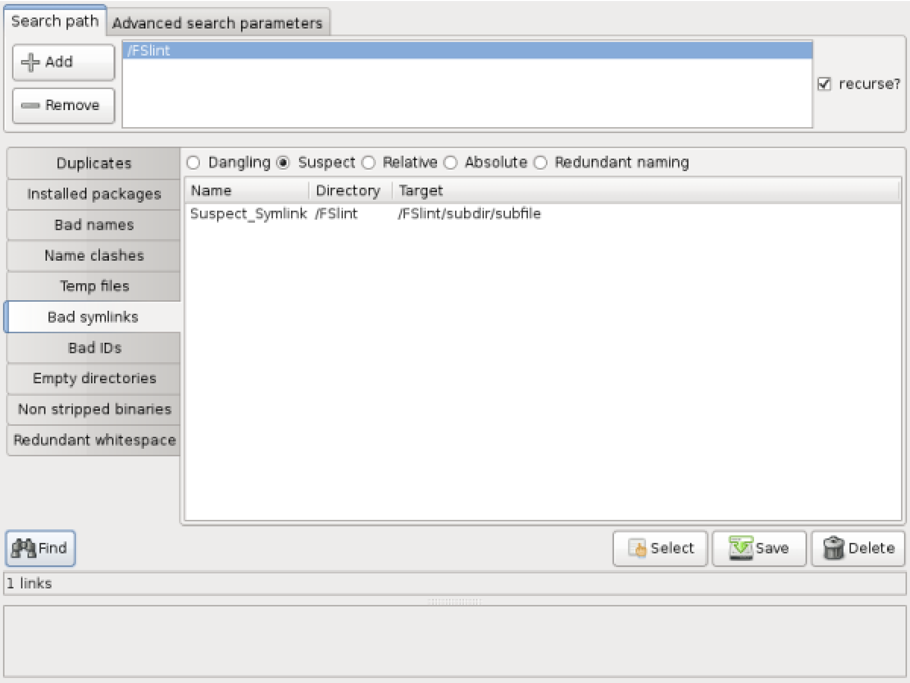
☑ recurse?

● Dangling ○ Suspect ○ Relative ○ Absolute ○ Redundant naming

Name	Directory	Target
Where_Is_My_File	/FSlint	My_File_Is_Gone

1 links

"Dangling" symlinks point to a file that is no longer there. These symlinks do not take up a lot of hard drive space, but they can often cause confusion as they clutter the file system. In this example, the file My_File_Is_Gone has been deleted leaving the symlink Where_Is_My_File dangling.



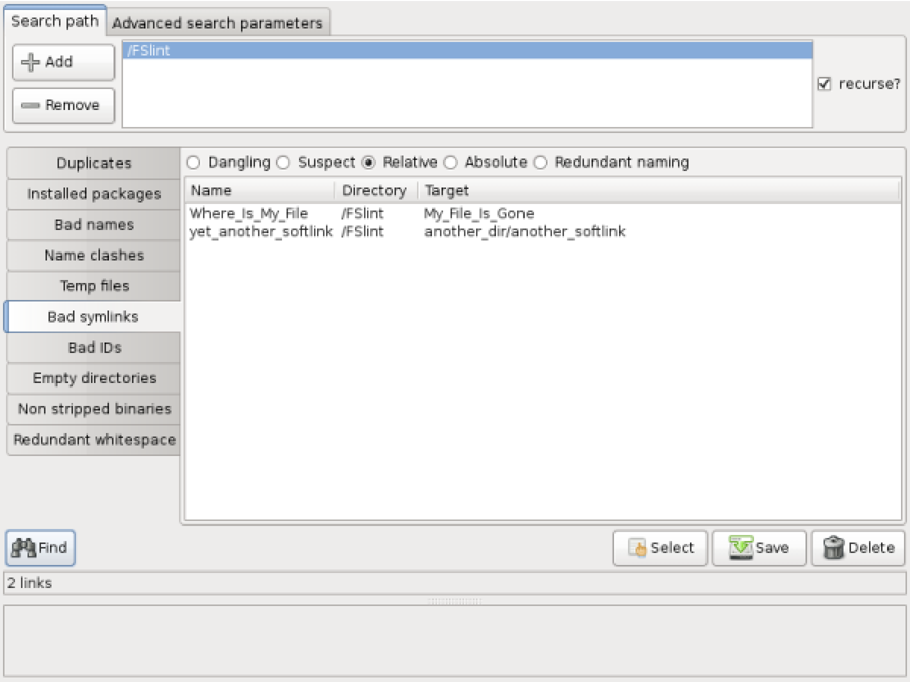
/FSlint ☑ recurse?

○ Dangling ● Suspect ○ Relative ○ Absolute ○ Redundant naming

Name	Directory	Target
Suspect_Symlink	/FSlint	/FSlint/subdir/subfile

1 links

"Suspect" symlinks point to a file below their directory structure. In this example Suspect_Sym link points to a valid file sub file, however, if not treated properly the file could easily be moved breaking the symlink.



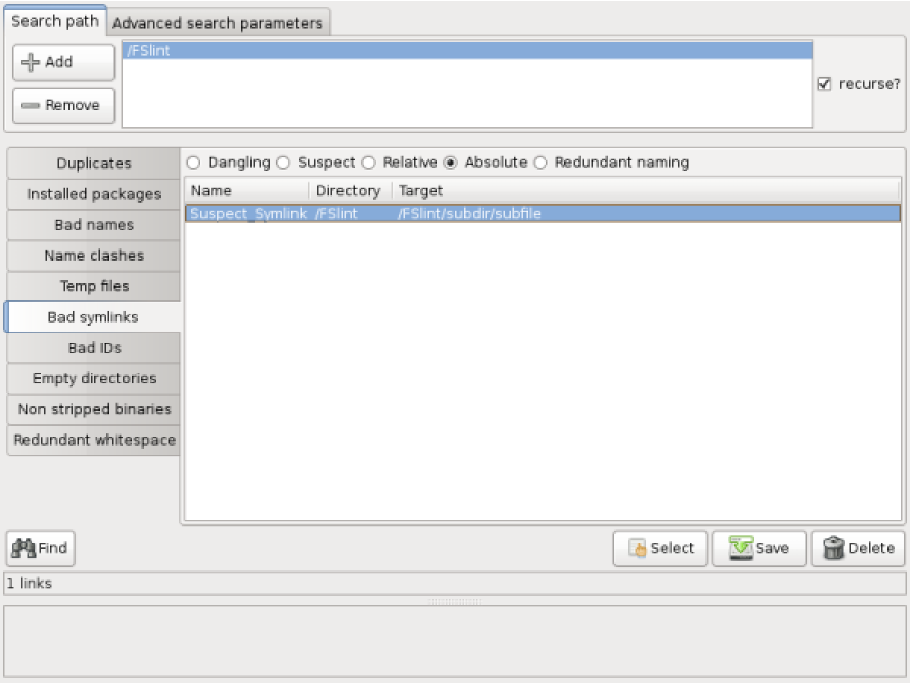
/FSlint ☑ recurse?

○ Dangling ○ Suspect ● Relative ○ Absolute ○ Redundant naming

Name	Directory	Target
Where_Is_My_File	/FSlint	My_File_Is_Gone
yet_another_softlink	/FSlint	another_dir/another_softlink

2 links

"Relative" symlinks point to a path that is determined by the current location. In this example yet_another_softlink will only work if another_softlink exists in a subdirectory of another_dir. Should this symlink be moved to another location, it will cease to work.



/FSlint

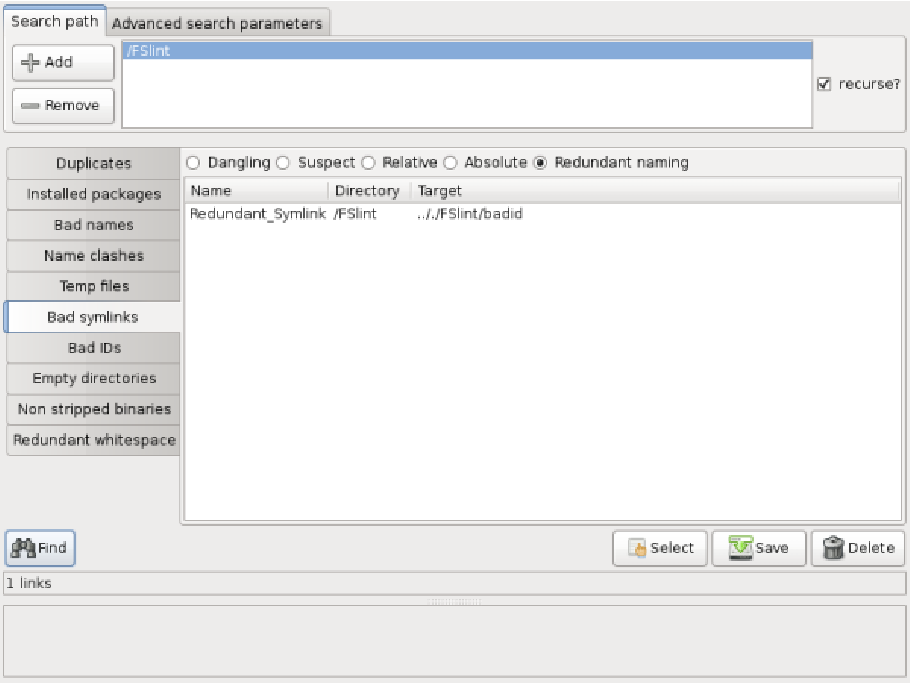
☑ recurse?

○ Dangling ○ Suspect ○ Relative ● Absolute ○ Redundant naming

Name	Directory	Target
Suspect_Symlink	/FSlint	/FSlint/subdir/subfile

1 links

"Absolute" symlinks point to an exact full path location. In this example the path is expressly stated. This is generally a good practice as it allows the symlinks to move at the user's discre tion. However, should the original file be moved or deleted, these symlinks will fail.



/FSlint

☑ recurse?

○ Dangling ○ Suspect ○ Relative ○ Absolute ● Redundant naming

Name	Directory	Target
Redundant_Symlink	/FSlint	../../FSlint/badid

1 links

"Redundant" symlinks collapse and expand pointlessly in their path. In this example, FSlint starts in the /FSlint directory. The link is told to move into the parent directory (with ../), then to search the parent directory (with ./) before descending back into the originating folder, /FSlint, to find the file badid. This tells the link to perform three actions only to arrive in the same directory where it started. It would be better for the link to directly point to the absolute location of the file.

COMMAND LINE INTERFACE

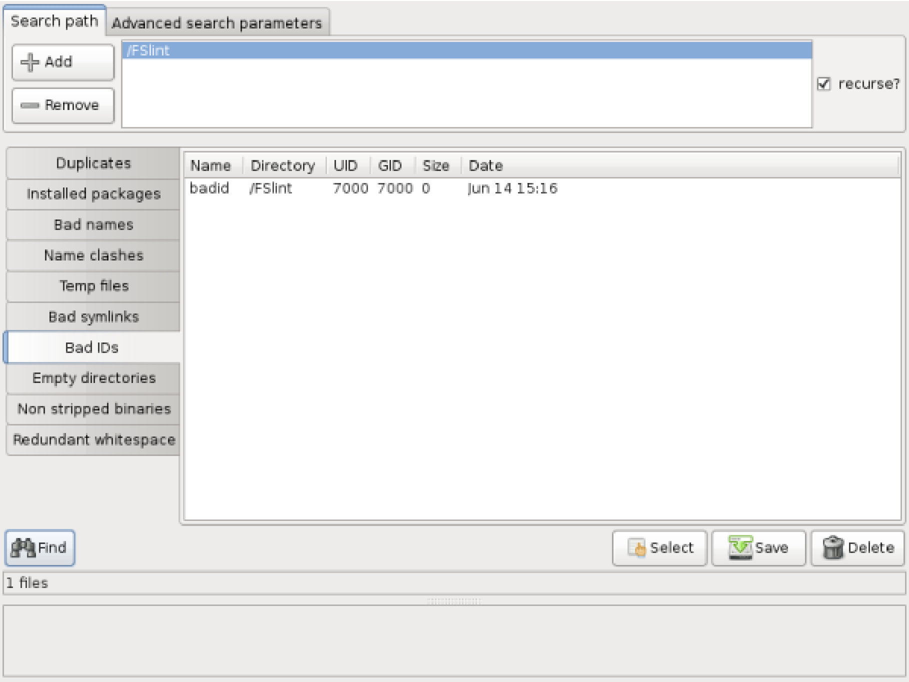
The command line interface to this utility is 'findbl'. This utility will be found in the installation directory of fslint.

```
$ /usr/share/fslint/fslint/findbl --help
 find "Bad" symbolic Links.
 Usage: findbl [-d] [-s] [-l] [-s] [-n] [[-r] [-f] paths(s) ...]

 These options are mutually exclusive (i.e. only the last one takes effect).
 -d Dangling (or stale) links. This is the default mode
 -s Suspect links (absolute links to paths within or below the link's directory)
 -l all reLative links
 -A all Absolute links
 -n reduNdant info in links (/././. ///// /../ etc.)

 If no path(s) specified, then the current directory is assumed.

 e.g. find dangling links in library directories:
 findbl $(getffl)
```

9. BAD ID'S

Every user is assigned an id number on a Linux system. When a user moves files between multiple computers, a file will occasionally end up with an user id that the current system can not map (or associate) to that user. For the desktop user, these can be annoying as access to these files may be denied if they fail to update the user id. The most common forms of bad id's on a desktop stem from changing Linux distributions or extracting a compressed file created by another user on another system. For the server administrators, bad ids's could be a sign that you may have a rogue process or worse, an intruder.

GRAPHICAL INTERFACE



/FSlint

☑ recurse?

Name	Directory	UID	GID	Size	Date
badid	/FSlint	7000	7000	0	Jun 14 15:16

1 files

In this example. FSlint correctly identifies a file with a bad id on this system because no user id nor group id of 7000 exists.

Note: There is a bug which does not allow the graphical interface for bad id's to function properly even though the command line functions as expected. This has been fixed in FSlint >=2.42 released on 2010-04-29. However, since the main repositories for several only have 2.26 the user will have to manually update FSlint or use the command line tools.

COMMAND LINE INTERFACE

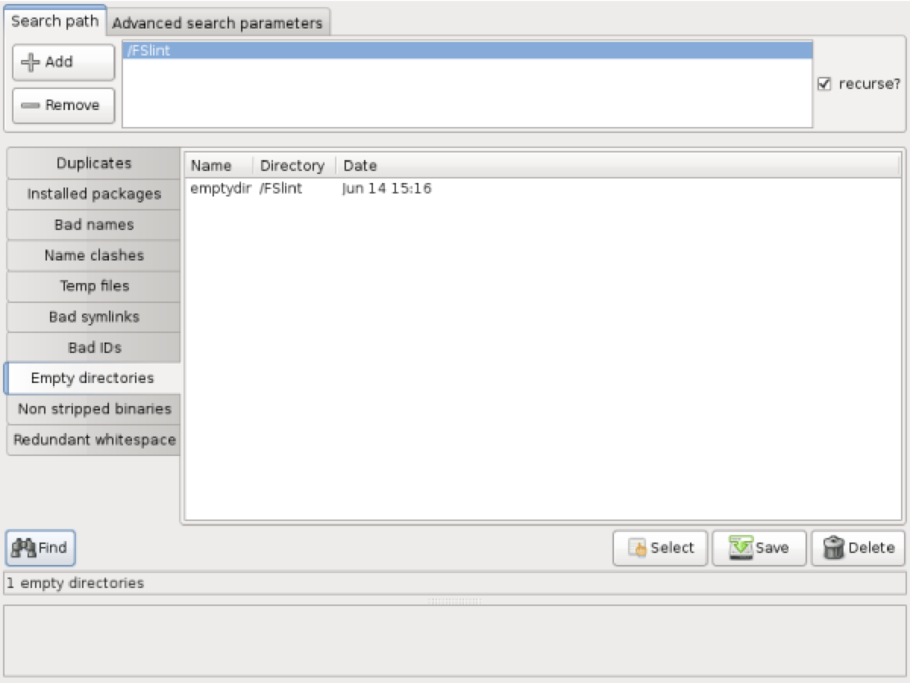
The command line interface to this utility is 'findid'. This utility will be found in the installation directory of fslint.

```
$ /usr/share/fslint/fslint/findid --help
 find unused or bad file ids
 Usage: findid [[-r] [-f] paths(s) ...]

 If no path(s) specified then the current directory is assumed.
```

10. EMPTY DIRECTORIES

Empty directories can clutter a file system and make it difficult for the average user to find information quickly and efficiently. For the more advanced user, filtering out empty directories is a trivial regular expression, but empty directories can be bothersome to the average desktop user. FSlint can find and remove these pesky empty directories.

GRAPHICAL INTERFACE



/FSlint

☑ recurse?

Name	Directory	Date
emptydir	/FSlint	Jun 14 15:16

1 empty directories

FSlint will find any empty directory within the 'Search path.'

COMMAND LINE INTERFACE

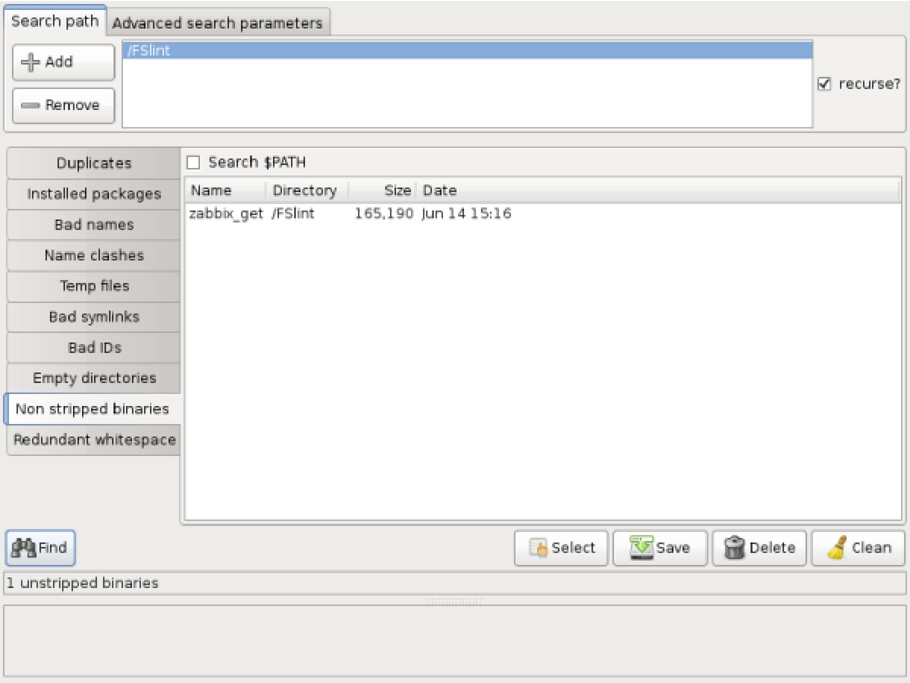
The command line interface to this utility is 'finded'. This utility will be found in the installation directory of fslint.

```
$ /usr/share/fslint/fslint/finded --help
 find Empty Directories.
 Usage: finded [[-r] [-f] paths(s) ...]

 If no path(s) specified then the current directory is assumed.
```

11. NON STRIPPED BINARIES

Non stripped binaries include debugging information and tend to be larger in size. Depending on the user's environment a non stripped binary could be unfavorable. Generally, the average desktop user does not need to be concerned with this feature. Instead, it is more useful to developers and those that prefer to compile their software. Often the extra debug information is not needed, and a considerable amount of drive space can be freed by removing this. Be aware that damage to the system can occur if the user is not careful.

GRAPHICAL INTERFACE



/FSlint

☑ recurse?

□ Search $PATH

Name	Directory	Size	Date
zabbix_get	/FSlint	165,190	Jun 14 15:16

1 unstripped binaries

In this example, the program zabbix_get was purposefully compiled to contain debugging information.

When the 'Search $PATH' checkbox is enabled, FSlint will search the users $PATH for non stripped binaries. This is particularly useful to administrators who wish to keep their executables free from debug information. If any executables are found, they can be stripped of the debug information by highlighting the executable and selecting the 'Clean' button. Be aware that damage to the system can occur if you are not careful.

COMMAND LINE INTERFACE

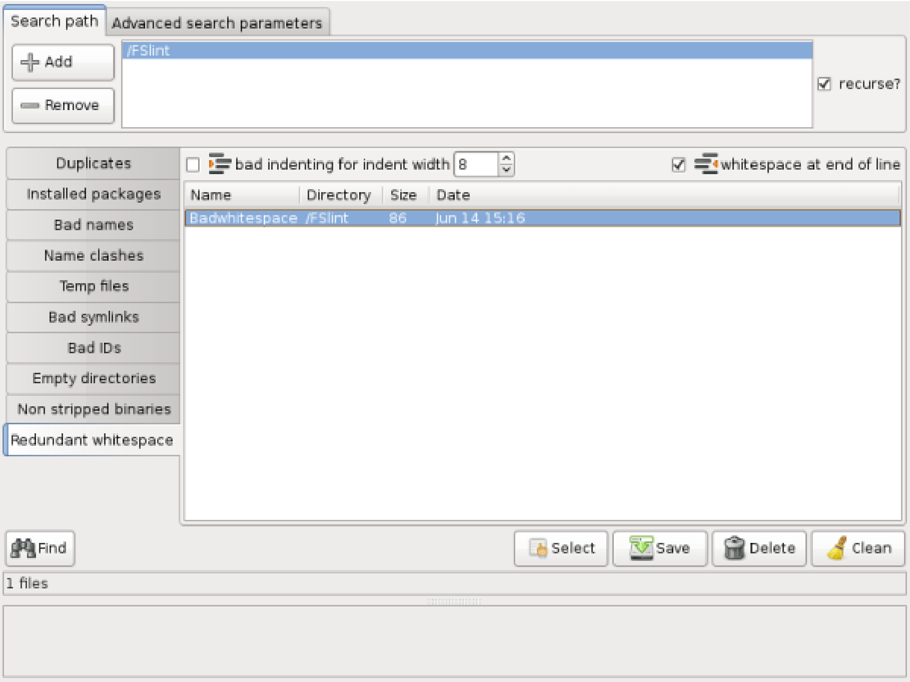
The command line interface to this utility is 'findns'. This utility will be found in the installation directory of fslint.

```
$ /usr/share/fslint/fslint/findns --help
 find NonStripped executables.
 Usage: findns [[-r] [-f] paths(s) ...]

 If no path(s) specified then the PATH is searched.
```

12. REDUNDANT WHITESPACES

FSlint can check text files for a number of whitespace issues, such as unnecessary tabs and spaces. This is a feature that is very useful for programmers and writers who need to be aware of the whitespace within their files. The average user may never need to use this, but knowing about this tool could still be beneficial.

GRAPHICAL INTERFACE



/FSlint

☑ recurse?

☐ bad indenting for indent width 8

☑ whitespace at end of line

Name	Directory	Size	Date
Badwhitespace	/FSlint	86	Jun 14 15:16

1 files

The checkbox for "bad indenting for indent width" can check text files to ensure that the indenting width is uniform and matches the desired number count specified. FSlint can also check for whitespace at the end of a line with the checkbox to the far right labelled "whitespace at end of line". The "Clean" button will attempt to fix the whitespace issues in the selected files.

In this example the file Badwhitespace is as follows:
This_file__
Has_lots___
of_extra__
whitespace__

All spaces have been replaced with underscores for visibility purposes. When FSlint is allowed to clean this file, the following is produced:
This_file
Has_lots
of_extra
whitespace

Further advanced options for cleaning can be completed by the FSlint command line utility.

COMMAND LINE INTERFACE

The command line interface to this utility is 'findrs'. This utility will be found in the installation directory of fslint.

```
$ /usr/share/fslint/fslint/findrs --help
 find Redundant whiteSpace.
 Usage: findrs [-w] [-t[#]] [-c] [[-r] [-f] paths(s) ...]

 -w enables mode to report whitespace at the end of lines.
 This is the default mode if none specified.

 -t enables mode to report erroneous mixing of indenting
 spaces and tabs (on a single line).
 If a number is passed to -t it sets the width of the tabs,
 which allows for more thorough checking.

 If -c specified then the number of lines in each file,
 with problematic whitespace is reported, in addition
 to the file names. Note this will take longer.

 If --view specified then the erroneous whitespace found
 is highlighted using vim.

 If no path(s) specified then the current directory is assumed.
```

Appendix

13. Other Command Line Tools

14. Special Thanks

13. OTHER COMMAND LINE TOOLS

These tools are not yet implemented in the FSlint graphical interface, yet they can be valuable assets for administrators or advanced users.

FIND REDUNDANT LIBRARIES

FSlint provides the utility 'findul' to find any redundant or unused Libraries.

```
$ /usr/share/fslint/fslint/findul --help
 Find possible redundant (Unused) Libraries.
 Usage: findul

 Note BE SURE you know a library is not needed before deleting it.
 For e.g. this tool doesn't recognise libraries that are only referenced
 at runtime (for e.g. plugins).
```

FIND WASTED SPACE IN EXT2 ENTRIES

FSlint provides the utility 'zipdir' to reclaim any wasted space in a EXT2 directory entry. As most newer distributions do not use EXT2 by default for their file system, the average user will not need to concern themselves with this utility.

```
$ /usr/share/fslint/fslint/zipdir --help
 Shrink Directories.
 Usage: zipdir [[-r] paths(s) ...]

 NB make sure that you don't process directories that are
 being referenced by running processes, as this utility
 will move directories from their current locations (for
 a small amount of time). Run only in single user mode
 if you are not sure of the consequences.

 If no path(s) specified then the current directory is assumed.
```

14. SPECIAL THANKS

I would like to give a special thanks to Pádraig Brady the author of FSlint. I have used this tool for quite some time and have gotten a lot out of this tool that he created and gave to the Open Source community. You can find his work at: http://www.pixelbeat.org/fslint/

Thanks also to my wife and everyone on the FLOSS manuals list that helped me proof read and weed out problems.

--Chris Stackpole

LICENSE

AUTHORS

Bad IDs
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Bad Names
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Bad Symlinks
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Common Interface Items
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Credits
© adam hyde 2006, 2007
Modifications:
Chris Stackpole 2010
John Curwood 2010

Duplicates
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Empty Directories
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Installed Packages
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Introduction
© adam hyde 2006, 2007
Modifications:
Chris Stackpole 2010
John Curwood 2010

Name Clashes
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Non Stripped Binaries
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Other Command Line Tools
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Redundant Whitespaces
© Chris Stackpole 2010
Modifications:
John Curwood 2010

Temp Files
© Chris Stackpole 2010
Modifications:
John Curwood 2010

FLOSS MANUALS

Free manuals for free software

GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appro priate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this Li cense and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

> **a)** You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
>
> **b)** You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
>
> **c)** If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written en tirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

> **a)** Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily

used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS