NAME

sccs-admin, admin – create and administer SCCS history files

SYNOPSIS

/usr/ccs/bin/admin [**–bhnz**] [**–a** *username* | *groupid*] . . . [**–d** *flag*] . . .
[**–e** *username* | *groupid*] . . . [**–f** *flag* [*value*]] . . . [**–i** [*filename*]]
[**–m** *mr-list*] [**–r***release*] [**–t** [*description-file*]] [**–y**[*comment*]] *s.filename* . . .

AVAILABILITY

SUNWsprot

DESCRIPTION

admin creates or modifies the flags and other parameters of SCCS history files. Filenames of SCCS history files begin with the '**s.**' prefix, and are referred to as **s.**files, or ''history'' files.

The named **s.**file is created if it does not exist already. Its parameters are initialized or modified according to the options you specify. Parameters not specified are given default values when the file is initialized, otherwise they remain unchanged.

If a directory name is used in place of the *s.filename* argument, the **admin** command applies to all **s.**files in that directory. Unreadable **s.**files produce an error. The use of '–' as the *s.filename* argument indicates that the names of files are to be read from the standard input, one **s.**file per line.

OPTIONS

–b Force encoding of binary data. Files that contain ASCII NUL or other control characters, or that do not end with a NEWLINE, are recognized as binary data files. The contents of such files are stored in the history file in encoded form. See **uuencode**(1C) for details about the encoding. This option is normally used in conjunction with **–i** to force **admin** to encode initial versions not recognized as containing binary data.

–h Check the structure of an existing **s.**file (see **sccsfile**(4)), and compare a newly computed check-sum with one stored in the first line of that file. **–h** inhibits writing on the file; and so nullifies the effect of any other options.

–n Create a new SCCS history file.

–z Recompute the file check-sum and store it in the first line of the **s.**file. Caution: it is important to verify the contents of the history file (see **sccs-val**(1), and the **print** subcommand in **sccs**(1)), since using **–z** on a truly corrupted file may prevent detection of the error.

–a *username* | *groupid* Add a user name, or a numerical group ID, to the list of users who may check deltas in or out. If the list is empty, any user is allowed to do so.

–d *flag* Delete the indicated *flag* from the SCCS file. The **–d** option may be specified only for existing s.files. See **–f** for the list of recognized flags.

–e *username* | *groupid* Erase a user name or group ID from the list of users allowed to make deltas.

–f *flag* [*value*] Set the indicated *flag* to the (optional) *value* specified. The following flags are recognized:

b Enable branch deltas. When **b** is set, branches can be created using the **–b** option of the SCCS **get** command (see **sccs-get**(1)).

c*ceil* Set a ceiling on the releases that can be checked out. *ceil* is a number less than or equal to 9999. If **c** is not set, the ceiling is 9999.

f*floor* Set a floor on the releases that can be checked out. The floor is a number greater than 0 but less than 9999. If **f** is not set, the floor is 1.

d*sid* The default delta number, or SID, to be used by an SCCS **get** command.

i Treat the '**No id keywords (ge6)**' message issued by an SCCS **get** or **delta** command as an error rather than a warning.

j Allow concurrent updates.

la
l*release*[**,***release* . . .]
Lock the indicated list of releases against deltas. If **a** is used, lock out deltas to all releases. An SCCS '**get –e**' command fails when applied against a locked release.

n Create empty releases when releases are skipped. These null (empty) deltas serve as anchor points for branch deltas.

q*value* Supply a *value* to which the **%Q%** keyword is to expand when a read-only version is retrieved with the SCCS **get** command.

m*module* Supply a value for the module name to which the **%M%** keyword is to expand. If the **m** flag is not specified, the value assigned is the name of the SCCS file with the leading **s.** removed.

t*type* Supply a value for the module type to which the **%Y%** keyword is to expand.

v [*program*] Specify a validation *program* for the MR numbers associated with a new delta. The optional *program* specifies the name of an MR number validity checking *program*. If this flag is set when creating an SCCS file, the –**m** option must also be used, in which case the list of MRs may be empty.

–**i** [*filename*] Initialize the history file with text from the indicated file. This text constitutes the initial delta, or set of checked-in changes. If *filename* is omitted, the initial text is obtained from the standard input. Omitting the –**i** option altogether creates an empty **s.**file. You can only initialize one **s.**file with text using –**i**. This option implies the –**n** option.

–**m** [*mr-list*] Insert the indicated Modification Request (MR) numbers into the commentary for the initial version. When specifying more than one MR number on the command line, *mr-list* takes the form of a quoted, space-separated list. A warning results if the **v** flag is not set or the MR validation fails.

–**r***release* Specify the release for the initial delta. –**r** may be used only in conjunction with –**i**. The initial delta is inserted into release 1 if this option is omitted. The level of the initial delta is always 1; initial deltas are named 1.1 by default.

–**t** [*description-file*] Insert descriptive text from the file *description-file*. When –**t** is used in conjunction with –**n**, or –**i** to initialize a new s.file, the *description-file* must be supplied. When modifying the description for an existing file: a –**t** option without a *description-file* removes the descriptive text, if any; a –**t** option with a *description-file* replaces the existing text.

–**y** [*comment*] Insert the indicated *comment* in the ''Comments:'' field for the initial delta. Valid only in conjunction with –**i** or –**n**. If –**y** option is omitted, a default comment line is inserted that notes the date and time the history file was created.

EXIT STATUS

The following exit values are returned:

0 Successful completion.
1 An error occurred.

FILES

s.∗ history file
SCCS/s.∗ history file in SCCS subdirectory
z.∗ temporary lock file

SEE ALSO

sccs(1), **sccs-cdc**(1), **sccs-delta**(1), **sccs-get**(1), **sccs-help**(1), **sccs-rmdel**(1), **sccs-val**(1), **sccsfile**(4)

Programming Utilities Guide

DIAGNOSTICS

Use the SCCS **help** command for explanations (see **sccs-help**(1)).

WARNINGS

The last component of all SCCS filenames must have the '**s.**' prefix. New SCCS files are given mode 444 (see **chmod**(1)). All writing done by **admin** is to a temporary file with an **x.** prefix, created with mode 444 for a new SCCS file, or with the same mode as an existing SCCS file. After successful execution of **admin**, the existing **s.** file is removed and replaced with the **x.**file. This ensures that changes are made to the SCCS file only when no errors have occurred.

It is recommended that directories containing SCCS files have permission mode 755, and that the **s.**files themselves have mode 444. The mode for directories allows only the owner to modify the SCCS files contained in the directories, while the mode of the **s.**files prevents all modifications except those performed using SCCS commands.

If it should be necessary to patch an SCCS file for any reason, the mode may be changed to 644 by the owner to allow use of a text editor. However, extreme care must be taken when doing this. The edited file should *always* be processed by an '**admin** −**h**' to check for corruption, followed by an '**admin** −**z**' to generate a proper check-sum. Another '**admin** −**h**' is recommended to ensure that the resulting **s.**file is valid.

admin also uses a temporary lock **s.**file, starting with the '**z.**' prefix, to prevent simultaneous updates to the **s.**file. See **sccs-get**(1) for further information about the '**z.**file'.