

**NAME**

sa - shell accounting

**SYNOPSIS**

sa [ -abcijlmnorstuv ] [ file ]

**DESCRIPTION**

When a user logs in, if the Shell is able to open the file `/usr/adm/sh_acct`, then as each command completes, the Shell writes at the end of this file the name of the command, the user, system time and real time consumed, and the user ID. *Sa* reports on, cleans up, and generally maintains this and other accounting files. To turn accounting on and off, the accounting file must be created or destroyed externally. If the user is the super-user, accounting is placed into `/usr/adm/su_acct` instead. As with `sh_acct`, it must be created or destroyed externally. Similarly, if the file `/usr/adm/acct` is created and system accounting is activated (using `accton(2)`), an accounting record is written when each process terminates.

*Sa* is able to condense the information in `/usr/adm/su_acct` or `/usr/adm/acct` into a summary file `/usr/adm/savacct` which contains a count of the number of times each command was called and the time resources consumed. The summary file `/usr/adm/usracct` is used to record for each user the total number of commands executed, and the total cpu time used. This condensation is desirable because on a large system accounting files can grow by 100 blocks per day. The summary file is read before the accounting file, so the reports include all available information.

If a file name is given as the last argument, that file will be treated as the accounting file; `acct` is the default. When a Shell accounting file (`sh_acct` or `su_acct`) is processed, commands that were executed from a command file have an asterisk appended to their name. If the system accounting file, `acct`, is processed, an asterisk is appended to those commands that did not do an `exec(2)` (e.g. daemons). There are many options:

- a Do not place all command names containing unprintable characters and those used only once under the name "\*\*\*\*other".
- b Sort output by sum of user and system time divided by number of calls. Default sort is by sum of user and system times.
- c Besides total user time, system time, and real time for each command, print percentage of total time over all commands.
- i Do not read the summary files `/usr/adm/savacct` and `/usr/adm/usracct`, so only information in the accounting files is condensed and reported.
- j Instead of total minutes time for each category, give seconds per call.
- l Separate system and user time; normally they are combined.
- m Superseding all other flags except `u`, print for each user the login name, number of commands executed, and the total number of minutes of cpu time used.
- n Sort by number of calls.
- o Superseding the `l` flag, report for each command the ratio of user cpu time to user + system cpu time.
- r Reverse order of sort.
- s Merge accounting file into summary files `/usr/adm/savacct` and `/usr/adm/usracct` when done. This option also causes the `a` option to be used.
- t For each command report ratio of real time to the sum of user and system times.
- u Superseding all other options, print the raw data.
- v If the next character is a digit `n`, then type the name of each command used `n` times or

fewer. Await a reply from the typewriter; if it begins with "y", add the command to the category "\*\*\*junk\*\*". This is used to strip out garbage.

**FILES**

/usr/adm/sh\_acct  
/usr/adm/acct  
/usr/adm/savacct  
/usr/adm/usracct  
/usr/adm/su\_acct

**SEE ALSO**

ac(1), accton(1), accton(2)

**BUGS**

Probably.