



[News](#) | [Features](#) | [Download](#) | [GitHub](#) | [Documentation](#) | [Tutorial](#) | [FAQ](#) | [Contact](#)

Useful links:  
[Freecode](#)  
[IceWalkers](#)

[The Linux Foundation](#)  
[Linux Kernel Archives](#)  
[Linux Kernel Mailing List](#)  
[The Linux Documentation Project](#)



## *sadf* manual page

### NAME

sadf - Display data collected by sar in multiple formats.

### SYNOPSIS

```
sadf [-C] [-d | -j | -p | -x] [-H] [-h] [-T | -t | -U] [-V] [-P {cpu  
[,...]| ALL}] [-s [hh:mm:ss]] [-e [hh:mm:ss]] [--sar_options...  
][interval [count]][datafile]
```

### DESCRIPTION

The `sadf` command is used for displaying the contents of data files created by the `sar(1)` command. But unlike `sar`, `sadf` can write its data in many different formats (CSV, XML, etc.) The default format is one that can easily be handled by pattern processing commands like `awk` (see option `-p`).

The `sadf` command extracts and writes to standard output records saved in the `datafile` file. This file must have been created by a version of `sar` which is compatible with that of `sadf`. If `datafile` is omitted, `sadf` uses the standard system activity file, the `/var/log/sa/sadd` file, where the `dd` parameter indicates the current day.

The interval and count parameters are used to tell `sadf` to select count records at interval seconds apart. If the count parameter is not set, then all the records saved in the data file will be displayed.

All the activity flags of `sar` may be entered on the command line to indicate which activities are to be reported. Before specifying them, put a pair of dashes (`--`) on the command line in order not to confuse the flags with those of `sadf`. Not specifying any flags selects only CPU activity.

## OPTIONS

**-C**

Tell `sadf` to display comments present in file.

**-d**

Print the contents of the data file in a format that can easily be ingested by a relational database system. The output consists of fields separated by a semicolon. Each record contains the hostname of the host where the file was created, the interval value (or `-1` if not applicable), the timestamp in a form easily acceptable by most databases, and additional semicolon separated data fields as specified by `sar_options` command line options. Note that timestamp output can be controlled by options `-T`, `-t` and `-U`.

**-e [ hh:mm:ss ]**

Set the ending time of the report, given in local time. The default ending time is `18:00:00`. Hours must be given in 24-hour format.

**-H**

Display the header of the report (when applicable). If no format has been specified, then the header data (metadata) of the data file are displayed.

**-h**

When used in conjunction with option `-d`, all activities will be displayed horizontally on a single line.

**-j**

Print the contents of the data file in JSON (JavaScript Object Notation) format. Timestamps can be controlled by options `-T` and `-t`.

**-P { cpu [,...] | ALL }**

Tell sadf that processor dependent statistics are to be reported only for the specified processor or processors. Specifying the **ALL** keyword reports statistics for each individual processor, and globally for all processors. Note that processor 0 is the first processor.

**-p**

Print the contents of the data file in a format that can easily be handled by pattern processing commands like awk. The output consists of fields separated by a tab. Each record contains the hostname of the host where the file was created, the interval value (or -1 if not applicable), the timestamp, the device name (or - if not applicable), the field name and its value. Note that timestamp output can be controlled by options -T, -t and -U.

**-s [ hh:mm:ss ]**

Set the starting time of the data (given in local time), causing the sadf command to extract records time-tagged at, or following, the time specified. The default starting time is 08:00:00. Hours must be given in 24-hour format.

**-T**

Display timestamp in local time instead of UTC (Coordinated Universal Time).

**-t**

Display timestamp in the original local time of the data file creator instead of UTC (Coordinated Universal Time).

**-U**

Display timestamp (UTC - Coordinated Universal Time) in seconds from the epoch.

**-V**

Print version number then exit.

**-x**

Print the contents of the data file in XML format. Timestamps can be controlled by option -T. The corresponding DTD (Document Type Definition) and XML Schema are included in the sysstat source package. They are also available at <http://pagesperso-orange.fr/sebastien.godard/download.html>.

## ENVIRONMENT

The `sadf` command takes into account the following environment variable:

## S\_TIME\_DEF\_TIME

If this variable exists and its value is **UTC** then `sadf` will use UTC time instead of local time to determine the current daily data file located in the `/var/log/sa` directory.

## EXAMPLES

```
sadf -d /var/log/sa/sa21 -- -r -n DEV
```

Extract memory and network statistics from system activity file 'sa21', and display them in a format that can be ingested by a database.

```
sadf -p -P 1
```

Extract CPU statistics for processor 1 (the second processor) from current daily data file, and display them in a format that can easily be handled by a pattern processing command.

## FILES

`/var/log/sa/sadd`

Indicate the daily data file, where the `dd` parameter is a number representing the day of the month.

## AUTHOR

Sebastien Godard (sysstat <at> orange.fr)

## SEE ALSO

[sar\(1\)](#), [sadc\(8\)](#), [sa1\(8\)](#), [sa2\(8\)](#)

<http://pagesperso-orange.fr/sebastien.godard/>