



Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'ipcs.1p' command

\$ man ipcs.1p

IPCS(1P) POSIX Programmer's Manual IPCS(1P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

ipcs ? report XSI interprocess communication facilities status

SYNOPSIS

ipcs [-qms] [-a|-bcopt]

DESCRIPTION

The ipcs utility shall write information about active interprocess communication facilities.

Without options, information shall be written in short format for message queues, shared memory segments, and semaphore sets that are currently active in the system. Otherwise, the information that is displayed is controlled by the options specified.

OPTIONS

The ipcs utility shall conform to the Base Definitions volume of POSIX.1?2017, Section 12.2, Utility Syntax Guidelines.

The ipcs utility accepts the following options:

- q Write information about active message queues.
- m Write information about active shared memory segments.

-s Write information about active semaphore sets.

If -q, -m, or -s are specified, only information about those facilities shall be written. If none of these three are specified, information about all three shall be written subject to the following options:

- a Use all print options. (This is a shorthand notation for -b, -c, -o, -p, and -t.)
- b Write information on maximum allowable size. (Maximum number of bytes in messages on queue for message queues, size of segments for shared memory, and number of semaphores in each set for semaphores.)
- c Write creator's user name and group name; see below.
- o Write information on outstanding usage. (Number of messages on queue and total number of bytes in messages on queue for message queues, and number of processes attached to shared memory segments.)
- p Write process number information. (Process ID of the last process to send a message and process ID of the last process to receive a message on message queues, process ID of the creating process, and process ID of the last process to attach or detach on shared memory segments.)
- t Write time information. (Time of the last control operation that changed the access permissions for all facilities, time of the last msgsnd() and msgrcv() operations on message queues, time of the last shmat() and shmdt() operations on shared memory, and time of the last semop() operation on semaphores.)

OPERANDS

None.

STDIN

Not used.

INPUT FILES

- * The group database
- * The user database

ENVIRONMENT VARIABLES

The following environment variables shall affect the execution of `ipcs`:

`LANG` Provide a default value for the internationalization variables that are unset or null. (See the Base Definitions volume of POSIX.1?2017, Section 8.2, Internationalization Variables for the precedence of internationalization variables used to determine the values of locale categories.)

`LC_ALL` If set to a non-empty string value, override the values of all the other internationalization variables.

`LC_CTYPE` Determine the locale for the interpretation of sequences of bytes of text data as characters (for example, single-byte as opposed to multi-byte characters in arguments).

`LC_MESSAGES`

Determine the locale that should be used to affect the format and contents of diagnostic messages written to standard error.

`NLSPATH` Determine the location of message catalogs for the processing of `LC_MESSAGES`.

`TZ` Determine the timezone for the date and time strings written by `ipcs`. If `TZ` is unset or null, an unspecified default timezone shall be used.

ASYNCHRONOUS EVENTS

Default.

STDOUT

An introductory line shall be written with the format:

```
"IPC status from %s as of %s\n", <source>, <date>
```

where `<source>` indicates the source used to gather the statistics and `<date>` is the information that would be produced by the `date` command when invoked in the POSIX locale.

The `ipcs` utility then shall create up to three reports depending upon the `-q`, `-m`, and `-s` options. The first report shall indicate the status of message queues, the second report shall indicate the status of shared memory segments, and the third report shall indicate the status

of semaphore sets.

If the corresponding facility is not installed or has not been used since the last reboot, then the report shall be written out in the format:

```
"%s facility not in system.\n", <facility>
```

where <facility> is Message Queue, Shared Memory, or Semaphore, as appropriate. If the facility has been installed and has been used since the last reboot, column headings separated by one or more <space> characters and followed by a <newline> shall be written as indicated below followed by the facility name written out using the format:

```
"%s:\n", <facility>
```

where <facility> is Message Queues, Shared Memory, or Semaphores, as appropriate. On the second and third reports the column headings need not be written if the last column headings written already provide column headings for all information in that report.

The column headings provided in the first column below and the meaning of the information in those columns shall be given in order below; the letters in parentheses indicate the options that shall cause the corresponding column to appear; ``all'' means that the column shall always appear. Each column is separated by one or more <space> characters. Note that these options only determine what information is provided for each report; they do not determine which reports are written.

T (all) Type of facility:

q Message queue.

m Shared memory segment.

s Semaphore.

This field is a single character written using the format %c.

ID (all) The identifier for the facility entry. This field shall be written using the format %d.

KEY (all) The key used as an argument to msgget(), semget(), or shmget() to create the facility entry.

Note: The key of a shared memory segment is changed to

IPC_PRIVATE when the segment has been removed un?

til all processes attached to the segment detach

it.

This field shall be written using the format 0x%x.

MODE (all) The facility access modes and flags. The mode shall consist

of 11 characters that are interpreted as follows.

The first character shall be:

S If a process is waiting on a msgsnd() operation.

- If the above is not true.

The second character shall be:

R If a process is waiting on a msgrcv() operation.

C or - If the associated shared memory segment is to be

cleared when the first attach operation is exe?

cuted.

- If none of the above is true.

The next nine characters shall be interpreted as three sets

of three bits each. The first set refers to the owner's permissions; the next to permissions of others in the user?

group of the facility entry; and the last to all others.

Within each set, the first character indicates permission

to read, the second character indicates permission to write

or alter the facility entry, and the last character is a

<hyphen-minus> ('-').

The permissions shall be indicated as follows:

r If read permission is granted.

w If write permission is granted.

a If alter permission is granted.

- If the indicated permission is not granted.

The first character following the permissions specifies if

there is an alternate or additional access control method

associated with the facility. If there is no alternate or

additional access control method associated with the facil?

ity, a single <space> shall be written; otherwise, another

printable character is written.

OWNER (all) The user name of the owner of the facility entry. If the user name of the owner is found in the user database, at least the first eight column positions of the name shall be written using the format %s. Otherwise, the user ID of the owner shall be written using the format %d.

GROUP (all) The group name of the owner of the facility entry. If the group name of the owner is found in the group database, at least the first eight column positions of the name shall be written using the format %s. Otherwise, the group ID of the owner shall be written using the format %d.

The following nine columns shall be only written out for message queues:

CREATOR (a,c)

The user name of the creator of the facility entry. If the user name of the creator is found in the user database, at least the first eight column positions of the name shall be written using the format %s. Otherwise, the user ID of the creator shall be written using the format %d.

CGROUP (a,c)

The group name of the creator of the facility entry. If the group name of the creator is found in the group database, at least the first eight column positions of the name shall be written using the format %s. Otherwise, the group ID of the creator shall be written using the format %d.

CBYTES (a,o)

The number of bytes in messages currently outstanding on the associated message queue. This field shall be written using the format %d.

QNUM (a,o) The number of messages currently outstanding on the associated message queue. This field shall be written using the format %d.

QBYTES (a,b)

The maximum number of bytes allowed in messages outstanding on the associated message queue. This field shall be written using the format %d.

LSPID (a,p) The process ID of the last process to send a message to the associated queue. This field shall be written using the format:

"%d", <pid>

where <pid> is 0 if no message has been sent to the corresponding message queue; otherwise, <pid> shall be the process ID of the last process to send a message to the queue.

LRPID (a,p) The process ID of the last process to receive a message from the associated queue. This field shall be written using the format:

"%d", <pid>

where <pid> is 0 if no message has been received from the corresponding message queue; otherwise, <pid> shall be the process ID of the last process to receive a message from the queue.

STIME (a,t) The time the last message was sent to the associated queue.

If a message has been sent to the corresponding message queue, the hour, minute, and second of the last time a message was sent to the queue shall be written using the format %d:%2.2d:%2.2d. Otherwise, the format "no-entry" shall be written.

RTIME (a,t) The time the last message was received from the associated queue. If a message has been received from the corresponding message queue, the hour, minute, and second of the last time a message was received from the queue shall be written using the format %d:%2.2d:%2.2d. Otherwise, the format "no-entry" shall be written.

The following eight columns shall be only written out for shared memory segments.

CREATOR (a,c)

The user of the creator of the facility entry. If the user name of the creator is found in the user database, at least the first eight column positions of the name shall be written using the format %s. Otherwise, the user ID of the creator shall be written using the format %d.

CGROUP (a,c)

The group name of the creator of the facility entry. If the group name of the creator is found in the group database, at least the first eight column positions of the name shall be written using the format %s. Otherwise, the group ID of the creator shall be written using the format %d.

NATTCH (a,o)

The number of processes attached to the associated shared memory segment. This field shall be written using the format %d.

SEGSZ (a,b) The size of the associated shared memory segment. This field shall be written using the format %d.

CPID (a,p) The process ID of the creator of the shared memory entry.

This field shall be written using the format %d.

LPID (a,p) The process ID of the last process to attach or detach the shared memory segment. This field shall be written using the format:

"%d", <pid>

where <pid> is 0 if no process has attached the corresponding shared memory segment; otherwise, <pid> shall be the process ID of the last process to attach or detach the segment.

ATIME (a,t) The time the last attach on the associated shared memory segment was completed. If the corresponding shared memory segment has ever been attached, the hour, minute, and second of the last time the segment was attached shall be written using the format %d:%2.2d:%2.2d. Otherwise, the

format " no-entry" shall be written.

DTIME (a,t) The time the last detach on the associated shared memory segment was completed. If the corresponding shared memory segment has ever been detached, the hour, minute, and second of the last time the segment was detached shall be written using the format %d:%2.2d:%2.2d. Otherwise, the format " no-entry" shall be written.

The following four columns shall be only written out for semaphore sets:

CREATOR (a,c)

The user of the creator of the facility entry. If the user name of the creator is found in the user database, at least the first eight column positions of the name shall be written using the format %s. Otherwise, the user ID of the creator shall be written using the format %d.

CGROUP (a,c)

The group name of the creator of the facility entry. If the group name of the creator is found in the group database, at least the first eight column positions of the name shall be written using the format %s. Otherwise, the group ID of the creator shall be written using the format %d.

NSEMS (a,b) The number of semaphores in the set associated with the semaphore entry. This field shall be written using the format %d.

OTIME (a,t) The time the last semaphore operation on the set associated with the semaphore entry was completed. If a semaphore operation has ever been performed on the corresponding semaphore set, the hour, minute, and second of the last semaphore operation on the semaphore set shall be written using the format %d:%2.2d:%2.2d. Otherwise, the format " no-entry" shall be written.

The following column shall be written for all three reports when it is requested:

CTIME (a,t) The time the associated entry was created or changed. The hour, minute, and second of the time when the associated entry was created shall be written using the format %d:%2.2d:%2.2d.

STDERR

The standard error shall be used only for diagnostic messages.

OUTPUT FILES

None.

EXTENDED DESCRIPTION

None.

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 An error occurred.

CONSEQUENCES OF ERRORS

Default.

The following sections are informative.

APPLICATION USAGE

Things can change while ipcs is running; the information it gives is guaranteed to be accurate only when it was retrieved.

EXAMPLES

None.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

ipcrm

The Base Definitions volume of POSIX.1?2017, Chapter 8, Environment Variables, Section 12.2, Utility Syntax Guidelines

The System Interfaces volume of POSIX.1?2017, msgrcv(), msgsnd(), semget(), semop(), shmat(), shmdt(), shmget()

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html>.

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html.

IEEE/The Open Group

2017

IPCS(1P)