

## IPCS Inter-Process Communication System

ipcs shows information on the inter-process communication facilities for which the calling process has read access. By default, it shows information about all three resources: shared memory segments, message queues, and semaphore arrays. We can use below flags to get required information.

- -q : It lists only message queues for which the current process has read access.
- -m : To lists the shared memories.
- -b : Write information on maximum allowable size. (Maximum number of bytes in messages on queue for message queues, size of segments for shared memory).
- -c : Write creator's user name and group name;
- -t, -o, -s,

### Examples

1. # ipcs -m -i 425984 : To detailed information about an ipc facility(here for id-425984).
2. # ipcs -m -l : To get the system limits for each ipc facility.
3. # ipcs -l : To get the limits for all three IPC facilities.
4. # ipcs -u : To display current usage for all the IPC facility.

### Shared Memory:

A memory section is shared between different processes. In other words, one process writes to this memory and another process can read from this memory.

To determine the maximum size of a shared memory segment, run:

```
# cat /proc/sys/kernel/shmmax
```

The default shared memory limit for SHMMAX can be changed in the proc file system without reboot:

```
# echo 2147483648 > /proc/sys/kernel/shmmax
```

To make a change permanent, add the following line to the file /etc/sysctl.conf (your setting may vary). This file is used during the boot process.

```
# echo "kernel.shmmax=2147483648" >> /etc/sysctl.conf
```

To determine the system wide maximum number of shared memory segments, run:

```
cat /proc/sys/kernel/shmmni
```

The default shared memory limit for SHMMNI can be changed in the proc file system without reboot:

```
# echo 4096 > /proc/sys/kernel/shmmni
```

Alternatively, you can use sysctl(8) to change it:

```
# sysctl -w kernel.shmmni=4096
```

To make a change permanent, add the following line to the file /etc/sysctl.conf. This file is used during the boot process.

```
# echo "kernel.shmmni=4096" >> /etc/sysctl.conf
```

This parameter sets the total amount of shared memory pages that can be used system wide. Hence, SHMALL should always be at least  $\text{ceil}(\text{shmmax}/\text{PAGE\_SIZE})$ .

If you are not sure what the default PAGE\_SIZE is on your Linux system, you can run the following command:

```
$ getconf PAGE_SIZE
```

To determine the system wide maximum number of shared memory pages, run:

```
# cat /proc/sys/kernel/shmall
```

The default shared memory limit for SHMALL can be changed in the proc file system without reboot:

```
# echo 2097152 > /proc/sys/kernel/shmall
```

Alternatively, you can use sysctl(8) to change it:

```
sysctl -w kernel.shmall=2097152
```

To make the change permanent, add the following line to the file /etc/sysctl.conf. This file is used during the boot process.

```
# echo "kernel.shmall=2097152" >> /etc/sysctl.conf
```

To see all shared memory segments that are allocated on the system, execute:

```
$ ipcs -m
```

To find out more about this shared memory segment you can run:

```
ipcs -m -i 98305 //98305 is shmid
```

## IPCRM

Removes message queue, semaphore set, or shared memory identifiers.

```
ipcrm [ -m SharedMemoryID ] [ -M SharedMemoryKey ] [ -q MessageID ] [ -Q MessageKey ] [ -s SemaphoreID ] [ -S SemaphoreKey ] [ -@ WparName ]
```

To remove the shared memory segment associated with SharedMemoryID 18602, enter:

```
# ipcrm -m 18602
```

Link for ipcs, ipcrm, netstat etc

<https://www.ibm.com/docs/en/aix/7.3?topic=i-ipcs-command>

## File Descriptors

In Unix and Unix-like computer operating systems, a file descriptor (FD, less frequently fildes) is a process-unique identifier (handle) for a file or other input/output resource, such as a pipe or network socket.

To determine the maximum number of file handles for the entire system, run:

```
# cat /proc/sys/fs/file-max
```

To determine the current usage of file handles, run:

```
$ cat /proc/sys/fs/file-nr
```

The file-nr file displays three parameters:

- the total allocated file handles.
- the number of currently used file handles (with the 2.4 kernel); or the number of currently unused file handles (with the 2.6 kernel).
- the maximum file handles that can be allocated (also found in /proc/sys/fs/file-max).

The maximum number of file handles can be changed in the proc file system without reboot:

```
# echo 65536 > /proc/sys/fs/file-max
```

Alternatively, you can use `sysctl(8)` to change it:

```
# sysctl -w fs.file-max=65536
```

To make the change permanent, add or change the following line in the file `/etc/sysctl.conf`. This file is used during the boot process.

```
echo "fs.file-max=65536" >> /etc/sysctl.conf
```

Per-Process Limit: The maximum number of file descriptors that a single process can open is controlled by the `ulimit` command. You can view and set this limit using:

```
ulimit -n
```

To increase the per-process limit, you need to modify the `ulimit` settings.

Check the current limits:

```
ulimit -n
```

Set a new limit (e.g., 100000) for the current session:

```
ulimit -n 100000
```

Make the change permanent by adding the following lines to `/etc/security/limits.conf`:

```
* soft nfile 100000
```

```
* hard nfile 100000
```

## Configuration

In File `/etc/security/limits.conf`

```
#<domain> <type> <item> <value>
```

```
* soft msgqueue 5600000000100
```

```
* hard msgqueue 5600000000100
```

The line `* soft msgqueue 5600000000100` in `/etc/security/limits.conf` is intended to set a limit on the maximum number of messages that can be held in a POSIX message queue.

Soft means it can go far than it for some time like 7 days, after it will be treated as hard limit.

```
iris soft core unlimited
```

```
iris hard core unlimited
```

This configuration allows processes to generate core files of any size, which can be helpful for thorough debugging and analysis of crashes.