Shared Memory

Shared memory is one of the fastest IPC mechanisms, allowing multiple processes to share a segment of memory. Below is an explanation of the key system calls used in shared memory IPC:

1. shmget – Allocate a Shared Memory Segment

This system call is used to create or access a shared memory segment.

Syntax:

```
#include <sys/ipc.h>
#include <sys/shm.h>
int shmget(key_t key, size_t size, int shmflg);
```

Parameters:

- key: A unique identifier for the shared memory segment.
- size: The size of the shared memory segment in bytes.
- shmflg: Flags for permissions and behavior (e.g., IPC_CREAT).

Return Value:

- Returns the shared memory segment ID (shmid) on success, or -1 on failure.

Example:

```
int shmid = shmget(IPC_PRIVATE, 1024, IPC_CREAT |
                   0666);
if (shmid == -1) {
    perror("shmget failed");
}
```

2. shmat – Attach a Shared Memory Segment to a Process

This system call attaches the shared memory segment to the address space of the calling process.

Syntax:

```
#include <sys/shm.h>
void *shmat(int shmid, const void *shmaddr,
            int shmflg);
```

Parameters:

- shmid: Shared memory segment ID.
- shmaddr: Address at which to attach (usually NULL).
- shmflg: Flags for behavior (e.g., SHM_RDONLY for read-only access).

Return Value:

- Returns a pointer to the shared memory segment on success, or (void *) -1 on failure.

Example:

```
void *shmaddr = shmat(shmid, NULL, 0);
if (shmaddr == (void *) -1) {
    perror("shmat failed");
} else {
    printf("Shared memory attached at %p\n",
shmaddr);
}
```

3. shmdt - Detach a Shared Memory Segment

This system call detaches the shared memory segment from the address space of the calling process.

Syntax:

```
#include <sys/shm.h>
int shmdt(const void *shmaddr);
```

Parameters:

- shmaddr: Pointer to the shared memory segment.

Return Value:

- Returns 0 on success, or -1 on failure.

Example:

```
if (shmdt(shmaddr) == -1) {
    perror("shmdt failed");
} else {
    printf("Shared memory detached.\n");
}
```

4. shmctl - Control Shared Memory Segment

This system call performs control operations on the shared memory segment, such as removing it or retrieving its status.

Syntax:

```
#include <sys/shm.h>
int shmctl(int shmid, int cmd, struct shmid_ds *buf);
```

Parameters:

- shmid: Shared memory segment ID.
- cmd: Command to perform (IPC_RMID to delete the segment).
- buf: Pointer to a shmid_ds structure (used for IPC_STAT or IPC_SET).

Commands:

- IPC_RMID: Remove the shared memory segment.
- IPC_STAT: Retrieve information about the segment.
- IPC_SET: Set information about the segment.

Example (Deleting a Shared Memory Segment):

```
if (shmctl(shmid, IPC_RMID, NULL) == -1) {
    perror("shmctl failed");
} else {
    printf("Shared memory segment deleted.\n");
}
```

Summary of Commands

System Call	Description
shmget	Create or access a shared memory segment
shmat	Attach a shared memory segment to a process
shmdt	Detach a shared memory segment from a process
shmctl	Perform control operations (e.g., delete the segment)