

Linux Internals & Networking

System programming using Kernel interfaces

Team Emertxe



Contents



Linux Internals & Networking

Contents

- .Introduction
- .Transition to OS programmer
- .System Calls
- .Process
- .IPC
- .Signals
- .Networking
- .Threads
- .Synchronization
- .Process Management
- .Memory Management



Inter Process Communications (IPC)

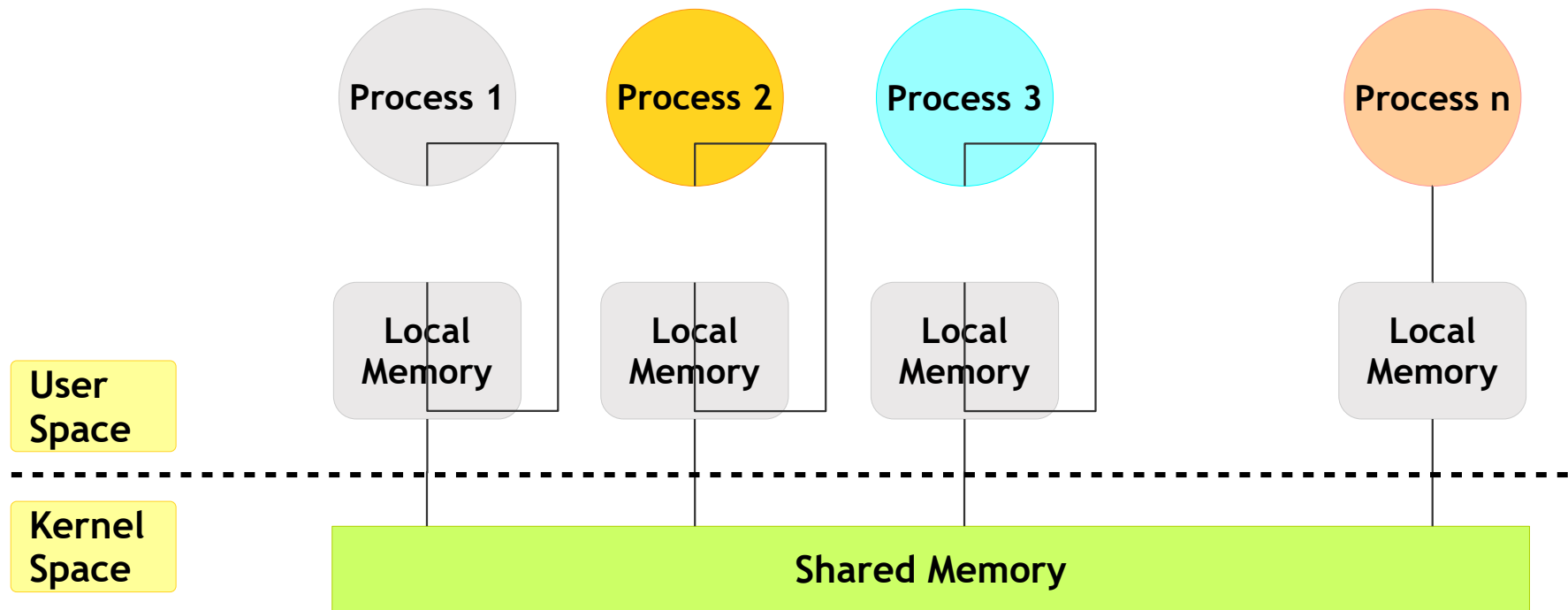


Team Emertxe



Inter Process Communications

Shared vs Local Memory



Inter Process Communications

Shared Memories - Procedure



.Create

.Attach

.Read/Write

.Detach

.Remove

} **95%**



Team Emertxe



Inter Process Communications

Shared Memories – Function calls



Function	Meaning
<code>int shmget(key_t key, size_t size, int shmflag)</code>	<ul style="list-style-type: none">✓ Create a shared memory segment✓ key: Seed input✓ size: Size of the shared memory✓ shmflag: Permission (similar to file)✓ RETURN: Shared memory ID / Failure
<code>void *shmat(int shmid, void *shmaddr, int shmflag)</code>	<ul style="list-style-type: none">✓ Attach to a particular shared memory location✓ shmid: Shared memory ID to get attached✓ shmaddr: Exact address (if you know or leave it 0)✓ shmflag: Leave it as 0✓ RETURN: Shared memory address / Failure
<code>int shmdt(void *shmaddr)</code>	<ul style="list-style-type: none">✓ Detach from a shared memory location✓ shmaddr: Location from where it needs to get detached✓ RETURN: SUCCESS / FAILURE (-1)
<code>shmctl(shmid, IPC_RMID, NULL)</code>	<ul style="list-style-type: none">✓ shmid: Shared memory ID✓ Remove and NULL

Inter Process Communications

Synchronization - Debugging



- The **ipcs** command provides information on inter-process communication facilities, including shared segments.
- Use the -m flag to obtain information about shared memory.
- For example, this image illustrates that one shared memory segment, numbered 392316, is in use:

Semaphores in the system

```
user@user:~$ ipcs -s
----- Semaphore Arrays -----
key      semid    owner    perms    nsems
0x00000000 392316   user     600      2
```

Shared Memory in the system

```
user@user:~$ ipcs -m | more
----- Shared Memory Segments -----
key      shmid    owner    perms    bytes    nattch   status
0x00000000 392316   user     600      524288   2        dest
0x00000000 557057   user     700      2116     2        dest
0x00000000 589826   user     700      5152     2        dest
```

Inter Process Communications

Summary



.We have covered

Data exchange

Communication

- .Pipes
- .FIFO
- .Shared memory
- .Signals
- .Sockets

Resource usage/access/control

Synchronization

- .Semaphores

Team Emertxe



Thank You