

TP AISE — Conteneurs / VMs

Exercice 1 : Lancement de VM qemu

Récupérez la VM (soit via clef USB ou bien via france.paratools.com/chps.qcow2) et lancez la avec le support réseau.

Réalisez ensuite un lancement en `-nographic` en passant les options compatibles avec le ballooning. Testez ensuite ces configurations dans votre VM avec la console (CTRL+a puis c). Pouvez-vous arrêter la VM depuis la console ?

Notez que cette VM contient Docker pour ceux qui n'arrivent pas à l'installer sur leur machine. Pour avoir docker-compose faire *apt-get install docker-compose*.

Exercice 2: PingPong SYS-V vers POSIX

En partant de l'exemple `./exemples_IPC_sysv/file/pingpong.c` qui repose sur les files SystemV (`msgrcv`, `msgsnd`, `msgctl`) portez le sur les files POSIX (message queues) en utilisant les fonctions (`mq_open`, `mq_receive`, `mq_send`, `mq_close`, `mq_unlink`).

Exercice 3: Segment SHM SYS-V vers POSIX

En partant de l'exemple `./exemples_IPC_sysv/shm/{shm_client.c,shm_server.c}` qui repose sur les segment de mémoire partagée SystemV (`shm_*`, `ftuncate`) portez le sur les segment de mémoire partagée POSIX (`shm_open`, `shm_unlink`).

Exercice 4: IO avec MMAP

Réalisez un programme qui manipule de manière structurée un fichier projeté en mémoire de 5 MB. Ce fichier sera constitué d'entiers (int) il vous appartiendra d'en réaliser le tri en mémoire dans le segment de mémoire partagée.

Exercice 5: Communication Via SHM

Réalisez deux programmes qui échangent des données via un fichier de 1MB projeté en mémoire. Vous devrez vous en servir pour faire transiter 100MB de donnée de la mémoire du processus A vers la mémoire du processus B.

Exercice 6: Shell

Réalisez votre propre shell simple capable de lancer des commandes. Implémentez le support des redirections de flux.