Roberto Myftaraga

Senior project report 4

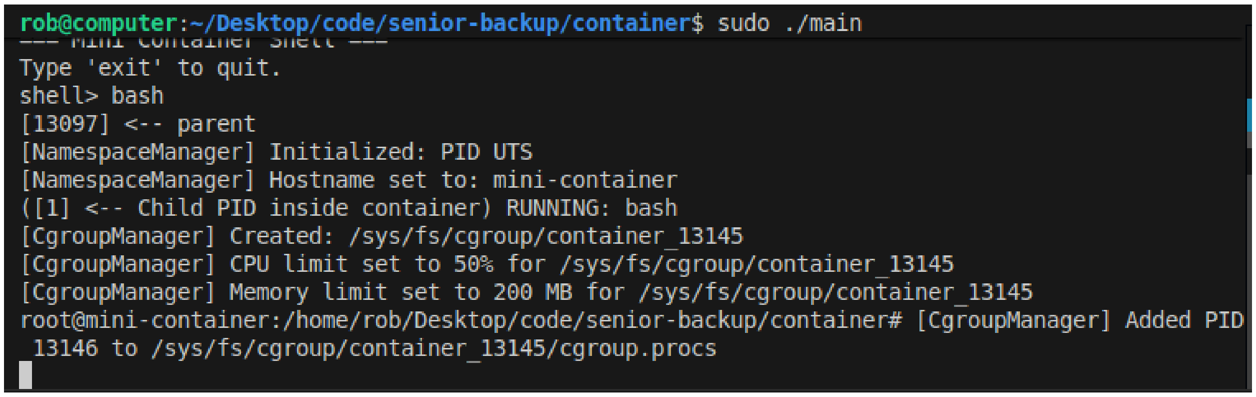
10/12/2025

Docker Based Container Engine: Process, Resource, and Filesystem Isolation

This week, I expanded my container prototype to include **resource control using Linux cgroups**, marking a significant step toward making it function like a real lightweight container runtime. After establishing process and hostname isolation with namespaces.

I implemented a new **CgroupManager** module that dynamically creates and manages control groups under **/sys/fs/cgroup**/. This allows each container to run with its own CPU and memory limits enforced by the kernel. I successfully integrated this feature with my existing Container class, so that when a new process is spawned inside the container, it is automatically assigned to its corresponding **cgroup**. The system can now restrict CPU usage to a set percentage and cap memory to a fixed size, preventing runaway processes from overwhelming the host. I also learned the importance of process ordering, ensuring that cgroup assignments happen after the inner container process is forked, and added cleanup logic to remove cgroup directories after termination.

Overall, this week's work deepened my understanding of **low-level resource isolation** and how modern runtimes like Docker use cgroups to provide stable and fair resource distribution across multiple containers.



```
rob@computer:~/Desktop/code/senior-backup/container$ sudo ./main
--- Mini Container Shell ---
Type 'exit' to quit.
shell> bash
[13097] <-- parent
[NamespaceManager] Initialized: PID UTS
[NamespaceManager] Hostname set to: mini-container
([1] <-- Child PID inside container) RUNNING: bash
[CgroupManager] Created: /sys/fs/cgroup/container_13145
[CgroupManager] CPU limit set to 50% for /sys/fs/cgroup/container_13145
[CgroupManager] Memory limit set to 200 MB for /sys/fs/cgroup/container_13145
root@mini-container:/home/rob/Desktop/code/senior-backup/container# [CgroupManager] Added PID
 13146 to /sys/fs/cgroup/container_13145/cgroup.procs
```

Img 1 demonstrates using cgroups to limit the system resources on the container.

Sources:

https://github.com/InspectRM/senior-backup