Менеджер контейнерів islander

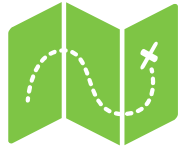
Денис Герасимук, Ярослав Морозевич, Дмитро Лопушанський

Суть проекту

Розробити аналог docker'а, який зможе запускати процеси в повністю ізольованих середовищах. **islander матиме такі функції:**

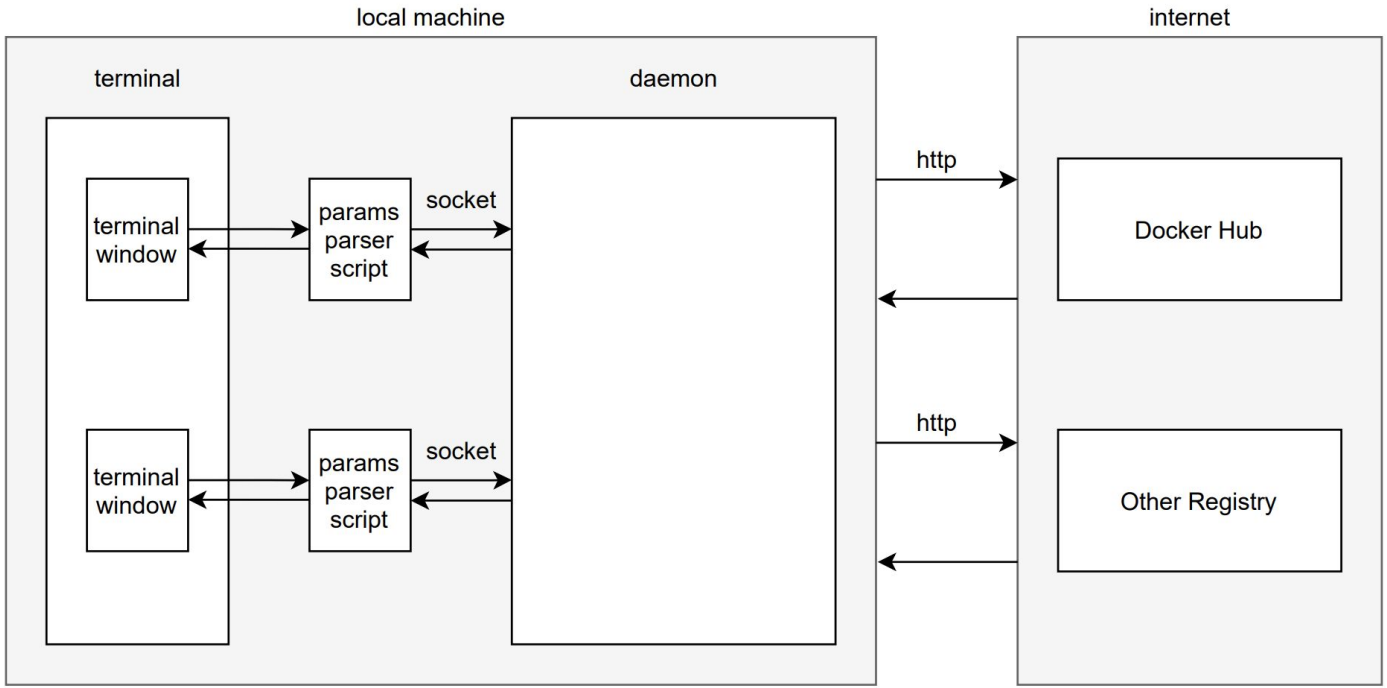
- Обмеження використання файлової системи, процесорної завантаженості, пам'яті, мережі
- Налаштування cgroups і створення namespace'ів
- client-server архітектура. CLI парсер та демон-процес

Етапи розробки

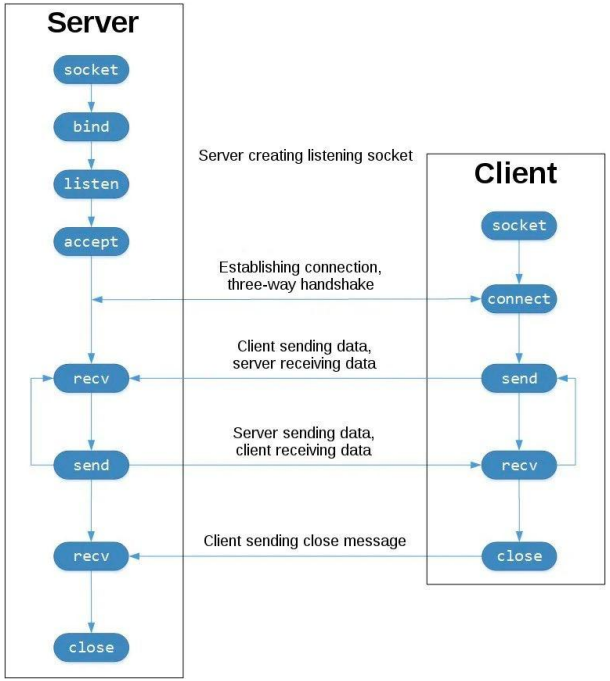


1. Етап дослідження: Технології, функціонал, деталі реалізації.
2. Розробка скриптів, які зможуть ізолювати процеси за певними параметрами
3. Написання парсера і сервера, які будуть спілкуватися через сокети
4. Поєднання всіх частин проекту, підтримка rootless mode
5. Додавання більшої кількості параметрів для ізоляції процесу, підтримка Volumes
6. Менеджмент контейнерів, налаштування комунікації між контейнерами
7. Підтримка більше форматів зовнішніх програм, які будуть ізолюватися
8. Покриття коду тестами, додавання нових фіч, загальне покращення

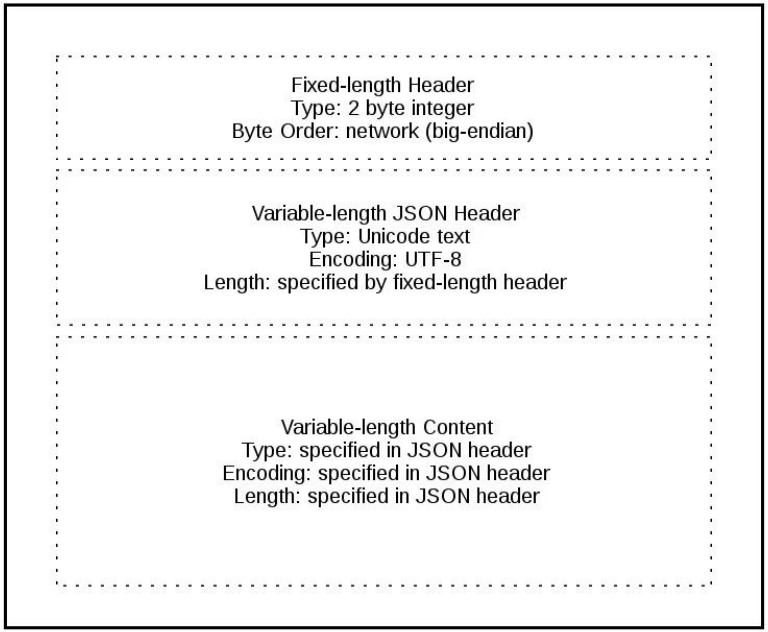
Загальний концепт системи



Принцип роботи сокетів



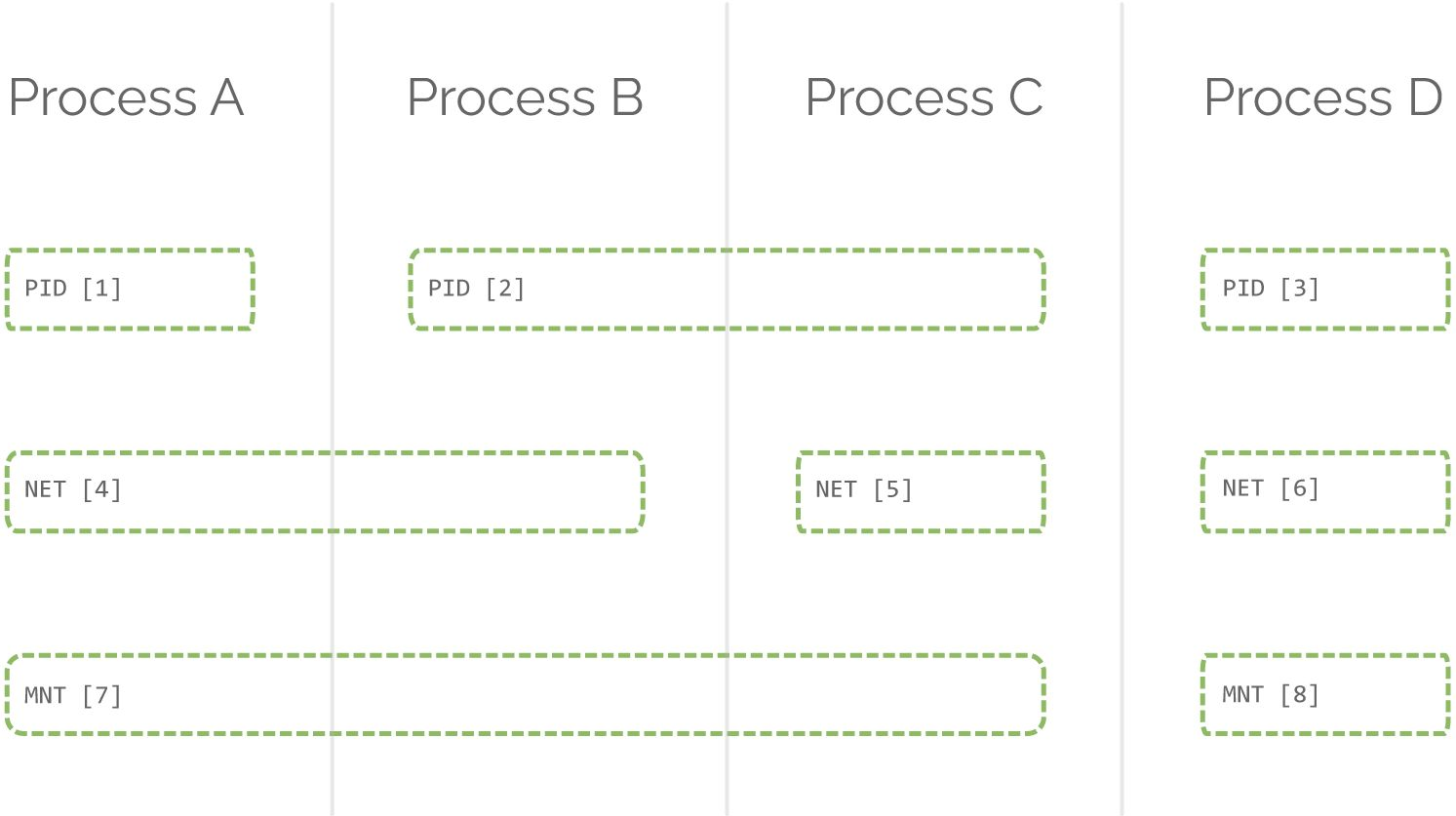
Принцип роботи сокетів



Message

Що таке namespace?

- Ізоляційний механізм для ресурсів
- Забезпечує відображення ресурсів зі змінами дозволів
- Зміни до процесів, які знаходяться в певному просторі імен, є невидимі поза його межами


Process A
Process B
Process C
Process D
PID [1]
PID [2]
PID [3]
NET [4]
NET [5]
NET [6]
MNT [7]
MNT [8]

Види namespace'ів

- **Mount** - керує точками монтування
- **Network** - керує мережевим стеком
- **PID** - надає процесам незалежний набір id
- **UTS** - дозволяє одній системі мати різні імена хостів/доменів
- **User Namespace** - забезпечує ізоляцію привілеїв користувача
- **IPC** - забезпечує комунікацію між процесами

Що таке Cgroup?

- **Namespace** — обмежує привілеї процесу

  Cgroup — ставить ліміти та обмежує типи ресурсів

- Дозволяють розподіляти ресурси серед визначених груп процесів

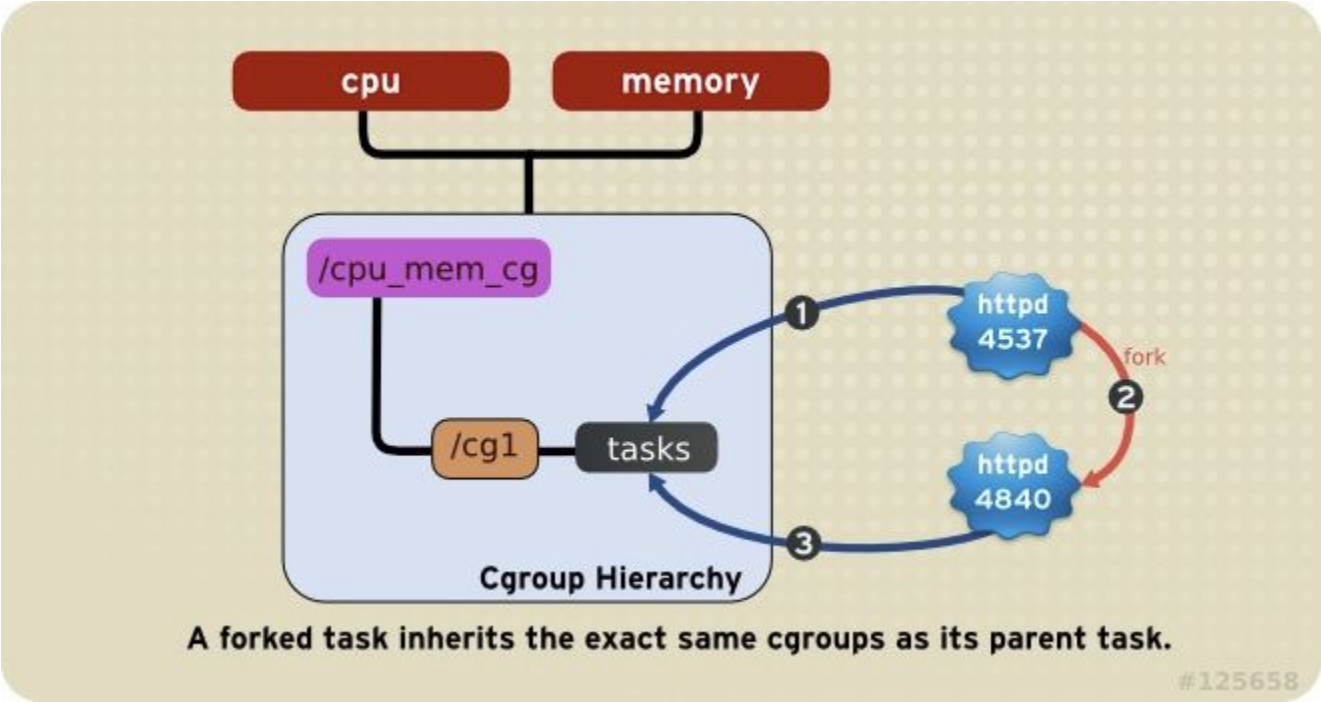
- Контроль над розподілом, визначенням пріоритетів, забороною, динамічне переналаштування лімітів → підвищення загальної ефективності

Модель Cgroup

- Подібні до процесів
  - ієрархічні
  - дочірні cgroups успадковують певні атрибути від батьківської cgroup

- Відмінне:
  - Linux є єдиним деревом процесів
  - модель cgroup — одне або кілька окремих, не пов'язаних між собою дерев процесів

Модель Cgroup



Cgroup subsystems

- **blkio** - читання та запис блочних девайсів
- **cpu** - доступ до процесора
- **devices** - доступ до девайсів
- **net_cls** - ліміти network io
- **memory** - RAM ліміти для cgroup

```
$ ls /sys/fs/cgroup/

blkio     cpu,cpuacct  freezer  net_cls          perf_event
cpu       cpuset       hugetlb  net_cls,net_prio pids
cpuacct   devices      memory   net_prio         systemd
```

Приклад використання

 # Create a group

$ cd /sys/fs/cgroup

$ mkdir -p memory/group1

 # Set a memory limit of 150M

$ echo 150M > memory/group1/memory.limit_in_bytes

 # Add shell to group

$ echo $$ > memory/group1/tasks

Реалізовано

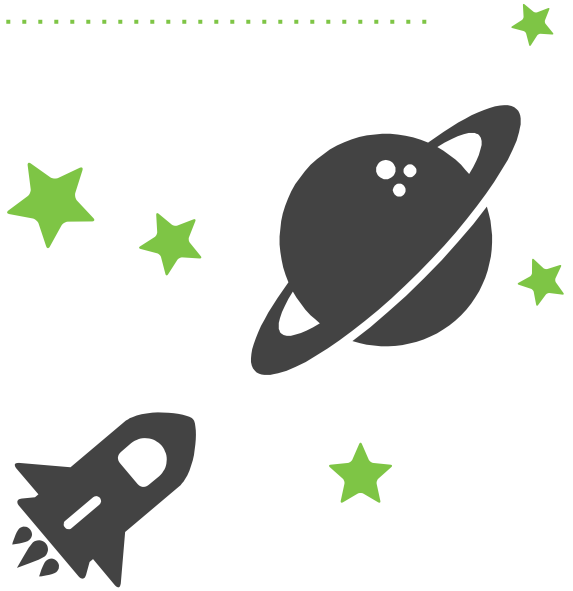
- Скрипт на C++, який парсить аргументи і надсилає структуроване повідомлення через socket
- Сервер на Python, який приймає повідомлення

- Mount, UTS, User, PID неймспейси
- Налаштовано ліміти через cgroup
- Використано syscalls, написано на C, не використано жодної додаткової cgroup бібліотеки

Налаштовані ліміти

Option [default]	Description
--memory-in-bytes [500M]	Memory limit (format: `<number>` [`<unit>`]). Number is a positive integer. Unit can be one of b, k, m, or g. Minimum is 4M.
--cpu-shares [100]	CPU shares (relative weight). For example, tasks in two cgroups that have cpu.shares set to 100 will receive equal CPU time, but tasks in a cgroup that has cpu.shares set to 200 receive twice the CPU time of tasks in a cgroup where cpu.shares is set to 100. The value specified in the cpu.shares file must be 2 or higher.
--cpu-period [100_000]	Limit the CPU CFS (Completely Fair Scheduler) period. If tasks in a cgroup should be able to access a single CPU for 0.2 seconds out of every 1 second, set cpu.cfs_quota_us to 200000 and cpu.cfs_period_us to 1000000.
--cpu-quota [1000_000]	Limit the CPU CFS (Completely Fair Scheduler) quota.
--device-read-bps [500M]	Limit read rate from the host filesystem (format: `<number>` [`<unit>`]). Number is a positive integer. Unit can be one of kb, mb, or gb.
--device-write-bps [100M]	Limit write rate the host filesystem (format: `<number>` [`<unit>]`). Number is a positive integer. Unit can be one of kb, mb, or gb.

Демо



Дякуємо!

Час для запитань

https://github.com/denysgerasymuk799/UCU_OS_Course_Project