

Требуется реализовать аналог докера под именем **mockер** на языке программирования на ваш выбор (кроме Bash). Рекомендуется Python или Go. Основная цель - попробовать на практике механизмы контейнеризации в Linux (Namespaces, Cgroups). Запрещается использовать готовые реализации OCI интерфейсов (например, runc, containerd). Для взаимодействия с Namespaces, Cgroups и другими механизмами ограничения предпочтительней использовать библиотеки выбранного языка, а не CLI утилиты.

1. **Требования к программе.** Должна предоставлять исполняемый файл с именем **mockер** и реализовывать следующие команды:

- a. **init <directory>** - создать образ контейнера используя указанную директорию как корневую. Возвращает в stdout id созданного образа.
- b. **pull <image>** - скачать последний (latest) тег указанного образа с Docker Hub. Возвращает в stdout id созданного образа.
- c. **rmi <image_id>** - удаляет ранее созданный образ из локального хранилища.
- d. **images** - выводит список локальных образов
- e. **ps** - выводит список контейнеров
- f. **run <image_id> <command>** - создает контейнер из указанного image_id и запускает его с указанной командой
- g. **exec <container_id> <command>** - запускает указанную команду внутри уже запущенного указанного контейнера
- h. **logs <container_id>** - выводит логи указанного контейнера
- i. **rm <container_id>** - удаляет ранее созданный контейнер
- j. **commit <container_id> <image_id>** - создает новый образ, применяя изменения из образа container_id к образу image_id
- k. **help** - выводит help по командам

Запущенный контейнер должен предоставлять доступ в сеть с помощью **nat** и ограничивать доступные ресурсы с помощью **Namespaces** и **Cgroups**.

2. **Тестирование и сборка.** С помощью GitLab CI нужно организовать тестирование проекта.

- a. Для отладки проекта удобно сделать тестировочный стенд в виде Vagrant-виртуалки т.к. для запуска Mockер понадобятся специфичные настройки ОС.
- b. Pipeline в GitLab CI должен запускаться только при push в master-ветку

3. **Сдача.** Для сдачи задания нужно заполнить [форму](#):

- a. В репозитории должно быть несколько коммитов, которые покажут, что gitlab-ci работает.
- b. Код для сборки стендов (Vagrantfile) нужно коммитить в этот же репозиторий.

4. Критерии оценивания

Реализация											Vagrantfile	Gitlab-ci	ИТОГ
a	b	c	d	e	f	g	h	i	j	k			
0,1	0,3	0,1	0,1	0,1	3,5	3	0,1	0,1	1	0,1	1	0,5	10,0

Deadline: 07.12, 23:59.

Рекомендации

В качестве примеров можно использовать:

<https://github.com/p8952/bocker> - немного старый проект на Bash (осторожно, там есть не очень правильно реализованные вещи)

<https://github.com/opencontainers/runc/> - реализация контейнерных ограничений, используемая в Docker

<https://github.com/containerd/containerd> - реализация верхнеуровневых операций (image и прочее), используемая в Docker

<https://github.com/janoszen/demo-container-runtime>

<https://lwn.net/Articles/531114/> - статья с примерами работы с Namespaces API