

# Технологии облачных вычислений

## Практическая работа №3

### Контейнерная виртуализация

Работа выполняется на вашей виртуальной машине.

**Задача:** научиться создавать контейнеры из шаблона и управлять ими.  
Развернуть вебсервер внутри контейнера и настроить приложение.  
Настроить проксирование запросов внутрь контейнера.

Перед началом выполнения практической работы сделайте снапшот вашей VM из интерфейса системы виртуализации (<https://psi.cc.dvo.ru:8006>).

Информацию по управлению контейнерами вы можете брать отсюда <https://ph0en1x.net/113-lxc-containers-linux-installation-cheat-sheet.html> или искать в официальной документации.

#### Быстрый старт:

Ниже представлена последовательность команд, которая позволит быстро развернуть простой контейнер. Дополнительную информацию ищите по ссылке выше (копия статьи лежит рядом с файлом этой практ. работы).

- 1) **apt update**
- 2) **apt install lxc**
- 3) **reboot**
- 4) **DOWNLOAD\_KEYSERVER="keyserver.ubuntu.com" lxc-create -  
-template download --name container1** (почему так читайте здесь - <https://www.claudiokuenzler.com/blog/1094/unable-to-create-lxc-container-error-unable-to-fetch-gpg-key-from-keyserver>)

Будет выведен список доступных шаблонов.

- 5) Вводим **Distribution: debian**
- 6) Вводим **Release: bullseye**
- 7) Вводим **Architecture: amd64**
- 8) Ждем пока загрузятся все необходимые файлы.
- 9) Команды для работы с контейнерами начинаются с **lxc-**смотрим список развернутых контейнеров **lxc-ls -f**

```
root@devozh-1:~# lxc-ls -f
NAME          STATE    AUTOSTART GROUPS IPV4  IPV6 UNPRIVILEGED
container1 STOPPED 0        -      -    -    false
```

10) **`lxc-start -n container1`**

11) **`lxc-ls -f`**

```
root@devozh-1:~# lxc-ls -f
NAME          STATE    AUTOSTART GROUPS IPV4      IPV6 UNPRIVILEGED
container1 RUNNING 0        -      10.0.3.156 -    false
```

Состояние контейнера должно быть **RUNNING**

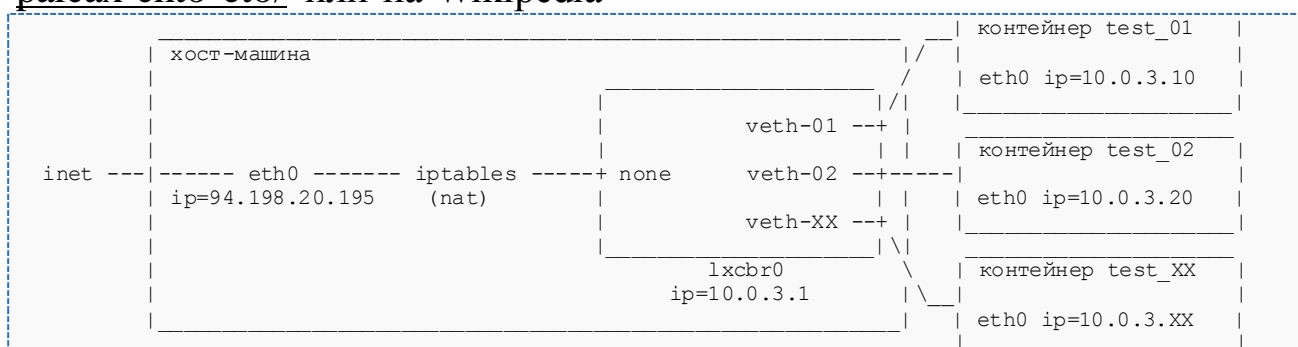
12) **`lxc-attach -n container1`** — присоединиться к контейнеру.

Вы находитесь внутри контейнера. Теперь все что вы будете делать будет затрагивать только ваш контейнер.

13) **`exit`** — выйти из контейнера в основную систему.

Контейнер имеет «серый» IP адрес, поэтому чтобы получить доступ к сети интернет используется NAT.

Можете прочитать о NAT здесь <https://wiki.merionet.ru/seti/13/nat-na-palcah-cto-eto/> или на Wikipedia



## Настройка контейнера:

Настройки выполняются на выключенном контейнере.

Сейчас ваш контейнер не имеет ограничений на использование ресурсов CPU и памяти, что не правильно. Файл конфигурации вашего контейнера находится по пути **`/var/lib/lxc/container1/config`**

Откройте файл и добавьте

**`# Memory limit 1024M`**

**`lxc.cgroup2.memory.max = 1024M`**

**`# CPU Cores 1`** (указываются номера ядер)

**`lxc.cgroup2.cpuset.cpus = 0`**

**`# Autostart`**

**`lxc.start.auto = 1`**

# MAC адрес виртуального интерфейса. Если не установить будет контейнер будет при каждом запуске получать новый IP. Замените «\*» на любые несколько символов в 16-ричном виде.

**lxc.net.0.hwaddr = 00:1E:2D:F7:\*:\***

Перезапустите контейнер (стоп и старт).

**ВНИМАНИЕ:** Обратите внимание на параметр **cgroup2**. В инструкциях в интернете часто предлагают использовать просто cgroup. Cgroup работать не будет. Даже ругаться не будет.

### **На вашей ВМ**

У каждой ВМ есть домен вида bik1-03.cc.dvo.ru

Для каждого подобного домена создан еще один вида bik1-03-s1.cc.dvo.ru.

Вам необходимо настроить проксирование запросов, которые будут приходить на «белый» адрес вашей ВМ, на адрес вашего контейнера (адрес можно узнать из **lxc-ls -f**). В качестве прокси сервера будем использовать **nginx reverse proxy**

информация по настройке тут [https://serveradmin.ru/nginx-proxy\\_pass/](https://serveradmin.ru/nginx-proxy_pass/)

### **Задание в контейнере:**

1) Устанавливаем и настраиваем в контейнере связку nginx+ php+ mysql как в практической работе №2.

2) Загружаем MODX в контейнер wget

<https://modx.com/download/direct/modx-3.0.0-pl.zip>

3) Устанавливаем согласно инструкции

<https://docs.modx.com/3.x/ru/getting-started/installation/standard>