

# EPAM University Programs

## DevOps external course

### Module 2 Virtualization and Cloud Basic

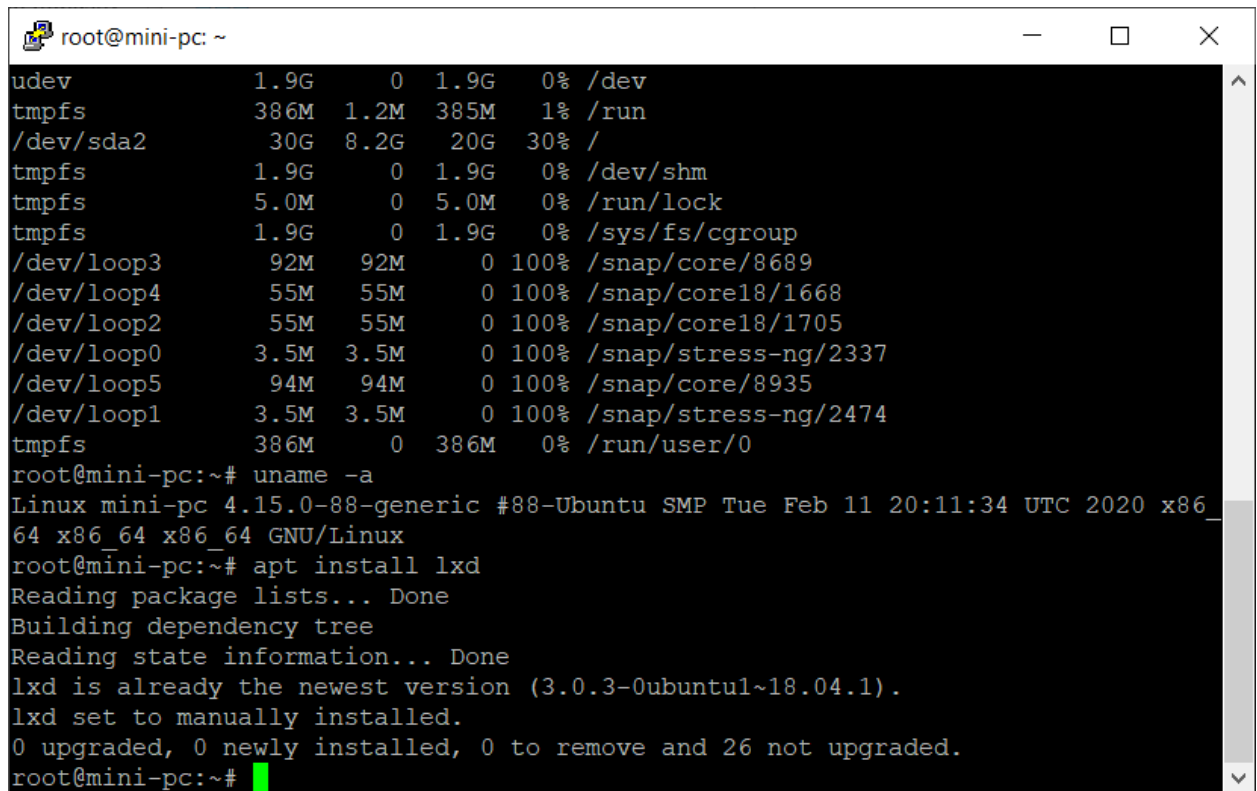
#### TASK 2.4

Работа с lxc в Ubuntu

Documentation - <https://help.ubuntu.com/its/serverguide/lxd.html>

<https://linuxcontainers.org/lxd/getting-started-cli/>

#### 1. Установить lxc



```
root@mini-pc: ~  
udev                1.9G      0    1.9G    0% /dev  
tmpfs               386M    1.2M   385M    1% /run  
/dev/sda2           30G    8.2G   20G   30% /  
tmpfs               1.9G      0    1.9G    0% /dev/shm  
tmpfs               5.0M      0    5.0M    0% /run/lock  
tmpfs               1.9G      0    1.9G    0% /sys/fs/cgroup  
/dev/loop3          92M    92M      0 100% /snap/core/8689  
/dev/loop4          55M    55M      0 100% /snap/core18/1668  
/dev/loop2          55M    55M      0 100% /snap/core18/1705  
/dev/loop0          3.5M    3.5M      0 100% /snap/stress-ng/2337  
/dev/loop5          94M    94M      0 100% /snap/core/8935  
/dev/loop1          3.5M    3.5M      0 100% /snap/stress-ng/2474  
tmpfs               386M      0   386M    0% /run/user/0  
root@mini-pc:~# uname -a  
Linux mini-pc 4.15.0-88-generic #88-Ubuntu SMP Tue Feb 11 20:11:34 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux  
root@mini-pc:~# apt install lxd  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
lxd is already the newest version (3.0.3-0ubuntu1~18.04.1).  
lxd set to manually installed.  
0 upgraded, 0 newly installed, 0 to remove and 26 not upgraded.  
root@mini-pc:~#
```

#### 2. Запустить lxc launch для любой из версий Убунту

```
root@mini-pc: ~
root@mini-pc:~# lxd init
Would you like to use LXD clustering? (yes/no) [default=no]:
Do you want to configure a new storage pool? (yes/no) [default=yes]:
Name of the new storage pool [default=default]:
Name of the storage backend to use (btrfs, dir, lvm) [default=btrfs]:
Create a new BTRFS pool? (yes/no) [default=yes]:
Would you like to use an existing block device? (yes/no) [default=no]:
Size in GB of the new loop device (1GB minimum) [default=15GB]: 10GB
Would you like to connect to a MAAS server? (yes/no) [default=no]:
Would you like to create a new local network bridge? (yes/no) [default=yes]:
What should the new bridge be called? [default=lxdbr0]:
What IPv4 address should be used? (CIDR subnet notation, "auto" or "none") [default=auto]:
What IPv6 address should be used? (CIDR subnet notation, "auto" or "none") [default=auto]: none
Would you like LXD to be available over the network? (yes/no) [default=no]: yes
Address to bind LXD to (not including port) [default=all]:
Port to bind LXD to [default=8443]:
Trust password for new clients:
Again:
Would you like stale cached images to be updated automatically? (yes/no) [default=yes]: yes
Would you like a YAML "lxd init" preseed to be printed? (yes/no) [default=no]:
root@mini-pc:~#
```

```
root@mini-pc: ~
Create and start containers from images

Usage:
  lxc launch [<remote>:]<image> [<remote>:][<name>] [flags]

Examples:
  lxc launch ubuntu:16.04 ul

Flags:
  -c, --config          Config key/value to apply to the new container
  -e, --ephemeral        Ephemeral container
  -n, --network          Network name
  --no-profiles          Create the container with no profiles applied
  -p, --profile          Profile to apply to the new container
  -s, --storage          Storage pool name
  --target              Cluster member name
  -t, --type             Instance type

Global Flags:
  --debug              Show all debug messages
  --force-local        Force using the local unix socket
  -h, --help           Print help
  -v, --verbose        Show all information messages
  --version            Print version number

Error: Invalid number of arguments
root@mini-pc:~# lxc launch ubuntu:lts fill_lxc
Creating fill_lxc
Error: Failed container creation: Container name isn't a valid hostname
root@mini-pc:~# lxc launch ubuntu:lts fill-lxc
Creating fill-lxc
Starting fill-lxc
root@mini-pc:~#
```

3. По окончании загрузки убедиться, что машина стартовала `lxc list`

```
root@mini-pc: ~  
lxc launch ubuntu:16.04 u1  
  
Flags:  
-c, --config          Config key/value to apply to the new container  
-e, --ephemeral       Ephemeral container  
-n, --network          Network name  
--no-profiles         Create the container with no profiles applied  
-p, --profile          Profile to apply to the new container  
-s, --storage          Storage pool name  
--target              Cluster member name  
-t, --type             Instance type  
  
Global Flags:  
--debug              Show all debug messages  
--force-local        Force using the local unix socket  
-h, --help           Print help  
-v, --verbose        Show all information messages  
--version            Print version number  
Error: Invalid number of arguments  
root@mini-pc:~# lxc launch ubuntu:16.04 fill_lxc  
Creating fill_lxc  
Error: Failed container creation: Container name isn't a valid hostname  
root@mini-pc:~# lxc launch ubuntu:16.04 fill-lxc  
Creating fill-lxc  
Starting fill-lxc  
root@mini-pc:~# lxc list  
+-----+-----+-----+-----+-----+-----+  
| NAME   | STATE | IPV4   | IPV6   | TYPE   | SNAPSHOTS |  
+-----+-----+-----+-----+-----+-----+  
| fill-lxc | RUNNING | 10.25.70.175 (eth0) | | PERSISTENT | 0 |  
+-----+-----+-----+-----+-----+-----+  
root@mini-pc:~#
```

4. Зайдите в контейнер с командной строкой `bash /bin/bash`

```
root@mini-pc:~# lxc exec fill-lxc bash
root@fill-lxc:~# pwd
/root
root@fill-lxc:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.25.70.175 netmask 255.255.255.0 broadcast 10.25.70.255
    inet6 fe80::216:3eff:fef3:1ba3 prefixlen 64 scopeid 0x20<link>
    ether 00:16:3e:f3:1b:a3 txqueuelen 1000 (Ethernet)
    RX packets 390 bytes 591665 (591.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 267 bytes 22977 (22.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 15 bytes 1227 (1.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 15 bytes 1227 (1.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@fill-lxc:~# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=56 time=21.9 ms
^C
--- 8.8.8.8 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 21.956/21.956/21.956/0.000 ms
root@fill-lxc:~#
```

5. Запустите обновление apt-get update

```
root@fill-lxc: ~  
root@fill-lxc:~# apt-get update  
Hit:1 http://archive.ubuntu.com/ubuntu bionic InRelease  
Get:2 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]  
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]  
Get:4 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]  
Get:5 http://archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [8570 kB]  
Get:6 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [677 kB]  
Get:7 http://security.ubuntu.com/ubuntu bionic-security/main Translation-en [218 kB]  
Get:8 http://archive.ubuntu.com/ubuntu bionic/universe Translation-en [4941 kB]  
Get:9 http://security.ubuntu.com/ubuntu bionic-security/restricted amd64 Packages [28.5 kB]  
Get:10 http://security.ubuntu.com/ubuntu bionic-security/restricted Translation-en [7568 B]  
Get:11 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [653 kB]  
Get:12 http://security.ubuntu.com/ubuntu bionic-security/universe Translation-en [217 kB]  
Get:13 http://archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [151 kB]  
Get:14 http://archive.ubuntu.com/ubuntu bionic/multiverse Translation-en [108 kB]  
Get:15 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [897 kB]  
Get:16 http://archive.ubuntu.com/ubuntu bionic-updates/main Translation-en [310 kB]  
Get:17 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [6968 B]  
Get:18 http://security.ubuntu.com/ubuntu bionic-security/multiverse Translation-en [2732 B]  
Get:19 http://archive.ubuntu.com/ubuntu bionic-updates/restricted amd64 Packages [37.5 kB]  
Get:20 http://archive.ubuntu.com/ubuntu bionic-updates/restricted Translation-en [9524 B]  
Get:21 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1061 kB]  
Get:22 http://archive.ubuntu.com/ubuntu bionic-updates/universe Translation-en [329 kB]  
Get:23 http://archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 Packages [10.5 kB]  
Get:24 http://archive.ubuntu.com/ubuntu bionic-updates/multiverse Translation-en [4696 B]  
Get:25 http://archive.ubuntu.com/ubuntu bionic-backports/main amd64 Packages [2512 B]  
Get:26 http://archive.ubuntu.com/ubuntu bionic-backports/main Translation-en [1644 B]  
Get:27 http://archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [4020 B]  
Get:28 http://archive.ubuntu.com/ubuntu bionic-backports/universe Translation-en [1900 B]  
Fetched 18.5 MB in 8s (2195 kB/s)  
Reading package lists... Done  
root@fill-lxc:~#
```

6. Установите (apt-get install) любую программу в контейнер. Например mc. Проверьте работоспособность

mc [root@fill-lxc]:~

Left	File	Command	Options	Right			
< ~				< ~			
.n	Name	Size	Modify time	.n	Name	Size	Modify time
/..	UP--DIR	Mar 17 15:53		/..	UP--DIR	Mar 17 15:53	
/.cache	4	Apr 3 15:06		/.cache	4	Apr 3 15:06	
/.config	4	Apr 3 15:06		/.config	4	Apr 3 15:06	
/.local	10	Apr 3 15:06		/.local	10	Apr 3 15:06	
/.ssh	30	Apr 3 15:01		/.ssh	30	Apr 3 15:01	
.bashrc	3106	Apr 9 2018		.bashrc	3106	Apr 9 2018	
.profile	148	Aug 17 2015		.profile	148	Aug 17 2015	

UP--DIR 8562M/9537M (89%)

UP--DIR 8562M/9537M (89%)

Hint: Want your plain shell? Press C-o, and get back to MC with C-o again.

root@fill-lxc:~#

1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit

## 7. Загрузите в контейнер файл

```
root@fill-lxc: ~  
-v, --verbose      Show all information messages  
--version          Print version number  
root@mini-pc:~# lxc config device list fill-lxc  
opt  
root@mini-pc:~# lxc config device remove fill-lxc opt  
Device opt removed from fill-lxc  
root@mini-pc:~# lxc config device add fill-lxc fill_home disk source=/home/fill path=/host_home_  
fill  
Device fill_home added to fill-lxc  
root@mini-pc:~# lxc exec fill-lxc bash  
root@fill-lxc:~# mc  
  
root@fill-lxc:~#  
root@fill-lxc:~# exit  
exit  
root@mini-pc:~# lxc config device add fill-lxc fill_home disk source=/home/fill path=/host_home_  
fill  
Error: The device already exists  
root@mini-pc:~# lxc config device remove fill-lxc fill_home  
Device fill_home removed from fill-lxc  
root@mini-pc:~# lxc config device add fill-lxc fill_home disk source=/home/fill path=/host_home_  
fill  
Device fill_home added to fill-lxc  
root@mini-pc:~# cd /home/fill  
root@mini-pc:/home/fill# touch download_test  
root@mini-pc:/home/fill# ls  
download_test  
root@mini-pc:/home/fill# lxc exec fill-lxc bash  
root@fill-lxc:~# cd /host_home_fill/  
root@fill-lxc:/host_home_fill# ls  
download_test  
root@fill-lxc:/host_home_fill# cp download_test /root/  
root@fill-lxc:/host_home_fill# cd /root/  
root@fill-lxc:~# ls  
download_test  
root@fill-lxc:~#
```

и скачайте с контейнера другой файл

```
root@mini-pc: /home/fill/tmp
drwxr-xr-x 1 root root 0 Apr 3 15:09 host_root
drwxr-xr-x 1 root root 438 Mar 17 15:50 lib
drwxr-xr-x 1 root root 40 Mar 17 15:48 lib64
drwxr-xr-x 1 root root 0 Mar 17 15:48 media
drwxr-xr-x 1 root root 0 Mar 17 15:48 mnt
drwxr-xr-x 1 root root 0 Mar 17 15:48 opt
dr-xr-xr-x 203 nobody nogroup 0 Apr 3 15:01 proc
drwx----- 1 root root 158 Apr 3 15:16 root
drwxr-xr-x 19 root root 680 Apr 3 15:01 run
drwxr-xr-x 1 root root 3694 Mar 17 15:50 sbin
drwxr-xr-x 1 root root 12 Apr 3 15:01 snap
drwxr-xr-x 1 root root 0 Mar 17 15:48 srv
dr-xr-xr-x 13 nobody nogroup 0 Apr 3 15:10 sys
drwxrwxrwt 1 root root 268 Apr 3 15:06 tmp
drwxr-xr-x 1 root root 70 Mar 17 15:48 usr
drwxr-xr-x 1 root root 108 Mar 17 15:50 var
root@fill-lxc:/# chmod o+w host_home_fill/
chmod: changing permissions of 'host_home_fill/': Operation not permitted
root@fill-lxc:/# exit
exit
root@mini-pc:/home/fill# mkdir tmp
root@mini-pc:/home/fill# chmod 777 tmp/
root@mini-pc:/home/fill# lxc exec fill-lxc bash
root@fill-lxc:~# cd /root/
root@fill-lxc:~# cp upload_from_lxc /host_
host_home_fill/ host_root/
root@fill-lxc:~# cp upload_from_lxc /host_home_fill/tmp/
root@fill-lxc:~# exit
exit
root@mini-pc:/home/fill# cd tmp/
root@mini-pc:/home/fill/tmp# ls -la
total 8
drwxrwxrwx 2 root root 4096 Apr 3 15:18 .
drwxr-xr-x 7 fill fill 4096 Apr 3 15:18 ..
-rw-r--r-- 1 100000 100000 0 Apr 3 15:18 upload_from_lxc
root@mini-pc:/home/fill/tmp#
```

Stop LXC container



```
root@mini-pc: /home/fill/tmp
drwxrwxrwt  1 root  root    268 Apr  3 15:06 tmp
drwxr-xr-x  1 root  root     70 Mar 17 15:48 usr
drwxr-xr-x  1 root  root    108 Mar 17 15:50 var
root@fill-lxc:/# chmod o+w host_home_fill/
chmod: changing permissions of 'host_home_fill/': Operation not permitted
root@fill-lxc:/# exit
exit
root@mini-pc:/home/fill# mkdir tmp
root@mini-pc:/home/fill# chmod 777 tmp/
root@mini-pc:/home/fill# lxc exec fill-lxc bash
root@fill-lxc:~# cd /root/
root@fill-lxc:~# cp upload_from_lxc /host_
host_home_fill/ host_root/
root@fill-lxc:~# cp upload_from_lxc /host_home_fill/tmp/
root@fill-lxc:~# exit
exit
root@mini-pc:/home/fill# cd tmp/
root@mini-pc:/home/fill/tmp# ls -la
total 8
drwxrwxrwx 2 root  root   4096 Apr  3 15:18 .
drwxr-xr-x 7 fill  fill   4096 Apr  3 15:18 ..
-rw-r--r-- 1 100000 100000    0 Apr  3 15:18 upload_from_lxc
root@mini-pc:/home/fill/tmp# lxc list
+-----+-----+-----+-----+-----+-----+
| NAME   | STATE | IPV4   | IPV6 | TYPE   | SNAPSHOTS |
+-----+-----+-----+-----+-----+-----+
| fill-lxc | RUNNING | 10.25.70.175 (eth0) |  | PERSISTENT | 0 |
+-----+-----+-----+-----+-----+-----+
root@mini-pc:/home/fill/tmp# lxc stop fill-lxc
root@mini-pc:/home/fill/tmp# lxc list
+-----+-----+-----+-----+-----+-----+
| NAME   | STATE | IPV4 | IPV6 | TYPE   | SNAPSHOTS |
+-----+-----+-----+-----+-----+-----+
| fill-lxc | STOPPED |  |  | PERSISTENT | 0 |
+-----+-----+-----+-----+-----+-----+
root@mini-pc:/home/fill/tmp#
```

Работа с Docker в Ubuntu

Documentation - <https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-18-04>

<https://docs.docker.com>

## 1. Установить docker

```
root@mini-pc: /home/fill/tmp
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.
Setting up cgroupfs-mount (1.4) ...
Setting up libltdl7:amd64 (2.4.6-2) ...
Setting up docker-ce-cli (5:19.03.8~3-0~ubuntu-bionic) ...
Setting up pigz (2.4-1) ...
Setting up docker-ce (5:19.03.8~3-0~ubuntu-bionic) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for libc-bin (2.27-3ubuntu1) ...
Processing triggers for systemd (237-3ubuntu10.38) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ureadahead (0.100.0-21) ...
root@mini-pc: /home/fill/tmp# systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2020-04-03 15:32:32 UTC; 20s ago
     Docs: https://docs.docker.com
    Main PID: 4190 (dockerd)
      Tasks: 12
     CGroup: /system.slice/docker.service
             └─4190 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Apr 03 15:32:31 mini-pc dockerd[4190]: time="2020-04-03T15:32:31.514205514Z" level=warning msg="You
Apr 03 15:32:31 mini-pc dockerd[4190]: time="2020-04-03T15:32:31.514298962Z" level=warning msg="You
Apr 03 15:32:31 mini-pc dockerd[4190]: time="2020-04-03T15:32:31.514345323Z" level=warning msg="You
Apr 03 15:32:31 mini-pc dockerd[4190]: time="2020-04-03T15:32:31.514970056Z" level=info msg="Loadi
Apr 03 15:32:31 mini-pc dockerd[4190]: time="2020-04-03T15:32:31.847934752Z" level=info msg="Defau
Apr 03 15:32:32 mini-pc dockerd[4190]: time="2020-04-03T15:32:32.018555676Z" level=info msg="Loadi
Apr 03 15:32:32 mini-pc dockerd[4190]: time="2020-04-03T15:32:32.113102691Z" level=info msg="Docker
Apr 03 15:32:32 mini-pc dockerd[4190]: time="2020-04-03T15:32:32.113455807Z" level=info msg="Daemon
Apr 03 15:32:32 mini-pc systemd[1]: Started Docker Application Container Engine.
Apr 03 15:32:32 mini-pc dockerd[4190]: time="2020-04-03T15:32:32.253742082Z" level=info msg="API 1
root@mini-pc: /home/fill/tmp#
```

## 2. Запустить поиск сконфигурированных решений для “ubuntu”

```
root@mini-pc: /home/fill/tmp
1 [OK] ubuntu-16-php-7.1
1 [OK] ubuntu with smartentry
1 [OK] Ubuntu images for GPDB development
root@mini-pc: /home/fill/tmp# docker search ubuntu
NAME                DESCRIPTION                STARS     OFFICIAL   AUTOMATED
ubuntu              Ubuntu is a Debian-based Linux operating sys... 10706     [OK]
dorowu/ubuntu-desktop-lxde-vnc Docker image to provide HTML5 VNC interface... 411       [OK]
rastasheep/ubuntu-sshd Dockerized SSH service, built on top of offi... 245       [OK]
consol/ubuntu-xfce-vnc Ubuntu container with "headless" VNC session... 212       [OK]
ubuntu-upstart      Upstart is an event-based replacement for th... 107       [OK]
ansible/ubuntu14.04-ansible Ubuntu 14.04 LTS with ansible 98          [OK]
neurodebian         NeuroDebian provides neuroscience research s... 68        [OK]
landinternet/ubuntu-16-nginx-php-phpmyadmin-mysql-5 ubuntu-16-nginx-php-phpmyadmin-mysql-5 50         [OK]
ubuntu-debootstrap  debootstrap --variant=minbase --components=m... 43        [OK]
nuagebec/ubuntu     Simple always updated Ubuntu docker images w... 24         [OK]
i386/ubuntu         Ubuntu is a Debian-based Linux operating sys... 19
ubuntu-16-apache-php-5.6 ubuntu-16-apache-php-5.6 14         [OK]
landinternet/ubuntu-16-apache-php-7.0 ubuntu-16-apache-php-7.0 13         [OK]
landinternet/ubuntu-16-nginx-php-phpmyadmin-mariadb-10 ubuntu-16-nginx-php-phpmyadmin-mariadb-10 11         [OK]
landinternet/ubuntu-16-nginx-php-5.6 ubuntu-16-nginx-php-5.6 8          [OK]
landinternet/ubuntu-16-nginx-php-5.6-wordpress-4 ubuntu-16-nginx-php-5.6-wordpress-4 7          [OK]
landinternet/ubuntu-16-apache-php-7.1 ubuntu-16-apache-php-7.1 6          [OK]
darkheer/ubuntu     Base Ubuntu Image -- Updated hourly 5          [OK]
landinternet/ubuntu-16-nginx-php-7.0 ubuntu-16-nginx-php-7.0 4          [OK]
pivotaldata/ubuntu A quick freshening-up of the base Ubuntu doc... 4
pivotaldata/ubuntu16.04-build Ubuntu 16.04 image for GPDB compilation 2
landinternet/ubuntu-16-php-7.1 ubuntu-16-php-7.1 1          [OK]
landinternet/ubuntu-16-sshd ubuntu-16-sshd 1          [OK]
smartentry/ubuntu   ubuntu with smartentry 1          [OK]
pivotaldata/ubuntu-gpdb-dev Ubuntu images for GPDB development 1
```

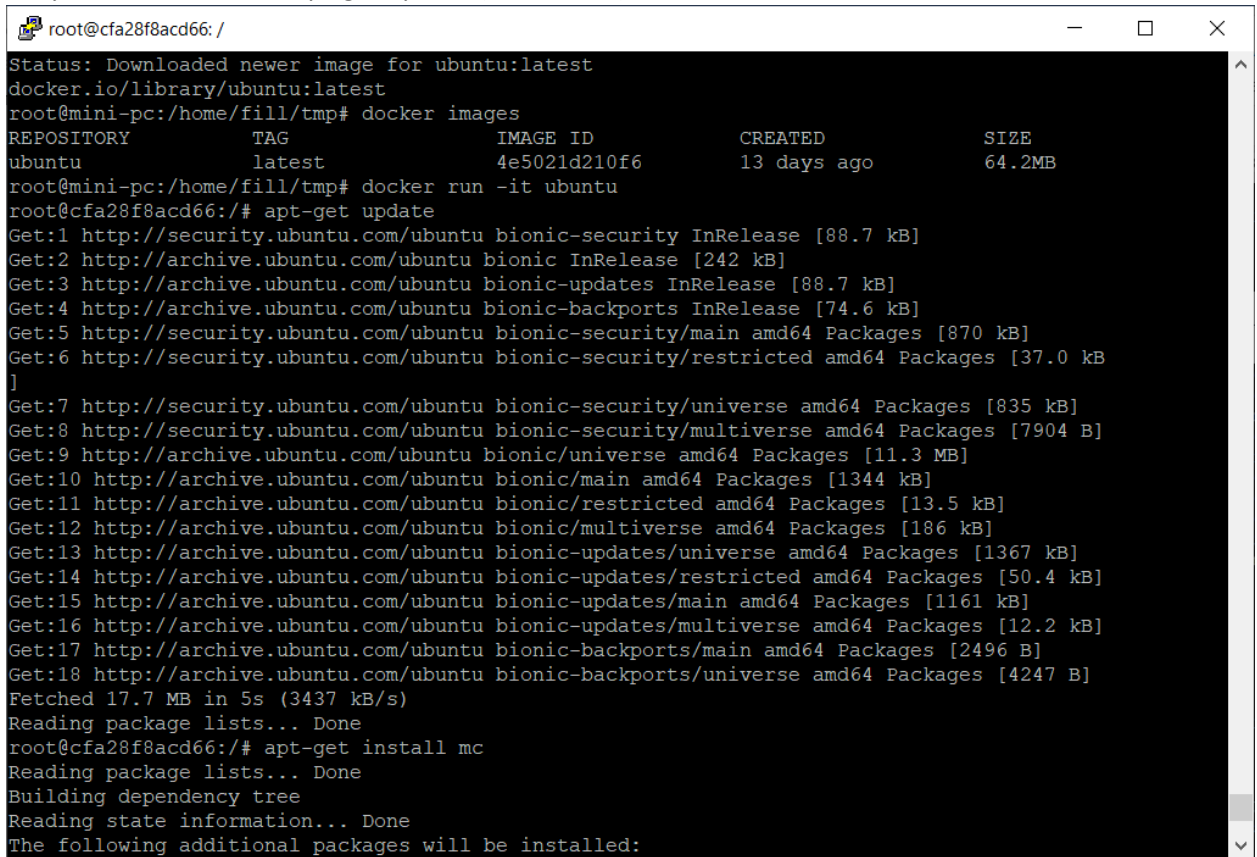
3. Скачать любой из образов на локальную машину.

```
root@mini-pc: /home/fill/tmp
landlinternet/ubuntu-16-apache-php-7.0
landlinternet/ubuntu-16-nginx-php-phpmyadmin-mariadb-10
landlinternet/ubuntu-16-nginx-php-5.6
landlinternet/ubuntu-16-nginx-php-5.6-wordpress-4
landlinternet/ubuntu-16-apache-php-7.1
darksheer/ubuntu
landlinternet/ubuntu-16-nginx-php-7.0
pivotaldata/ubuntu
pivotaldata/ubuntu16.04-build
landlinternet/ubuntu-16-php-7.1
landlinternet/ubuntu-16-sshd
smartentry/ubuntu
pivotaldata/ubuntu-gpdb-dev
root@mini-pc:/home/fill/tmp# docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
5bed26d33875: Pull complete
f11b29a9c730: Pull complete
930bda195c84: Pull complete
78bf9a5ad49e: Pull complete
Digest: sha256:bec5a2727be7fff3d308193cfde3491f8fba1a2ba392b7546b43a051853a341d
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
root@mini-pc:/home/fill/tmp#
```

4. Запустить команду просмотра загруженных на компьютер образов.

```
root@mini-pc: /home/fill/tmp
landlinternet/ubuntu-16-nginx-php-5.6-wordpress-4
landlinternet/ubuntu-16-apache-php-7.1
darksheer/ubuntu
landlinternet/ubuntu-16-nginx-php-7.0
pivotaldata/ubuntu
pivotaldata/ubuntu16.04-build
landlinternet/ubuntu-16-php-7.1
landlinternet/ubuntu-16-sshd
smartentry/ubuntu
pivotaldata/ubuntu-gpdb-dev
root@mini-pc:/home/fill/tmp# docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
5bed26d33875: Pull complete
f11b29a9c730: Pull complete
930bda195c84: Pull complete
78bf9a5ad49e: Pull complete
Digest: sha256:bec5a2727be7fff3d308193cfde3491f8fba1a2ba392b7546b43a051853a341d
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
root@mini-pc:/home/fill/tmp# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
ubuntu               latest              4e5021d210f6       13 days ago        64.2MB
root@mini-pc:/home/fill/tmp#
```

5. Запустите обновление apt-get update (screenshot)



```
root@cfa28f8acd66: /
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
root@mini-pc:/home/kill/tmp# docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
ubuntu              latest          4e5021d210f6    13 days ago    64.2MB
root@mini-pc:/home/kill/tmp# docker run -it ubuntu
root@cfa28f8acd66:/# apt-get update
Get:1 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:2 http://archive.ubuntu.com/ubuntu bionic InRelease [242 kB]
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:4 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:5 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [870 kB]
Get:6 http://security.ubuntu.com/ubuntu bionic-security/restricted amd64 Packages [37.0 kB]
]
Get:7 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [835 kB]
Get:8 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [7904 B]
Get:9 http://archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [11.3 MB]
Get:10 http://archive.ubuntu.com/ubuntu bionic/main amd64 Packages [1344 kB]
Get:11 http://archive.ubuntu.com/ubuntu bionic/restricted amd64 Packages [13.5 kB]
Get:12 http://archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [186 kB]
Get:13 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1367 kB]
Get:14 http://archive.ubuntu.com/ubuntu bionic-updates/restricted amd64 Packages [50.4 kB]
Get:15 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [1161 kB]
Get:16 http://archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 Packages [12.2 kB]
Get:17 http://archive.ubuntu.com/ubuntu bionic-backports/main amd64 Packages [2496 B]
Get:18 http://archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [4247 B]
Fetched 17.7 MB in 5s (3437 kB/s)
Reading package lists... Done
root@cfa28f8acd66:/# apt-get install mc
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
```

- Установите (apt-get install) любую программу в контейнер. Например mc. Проверьте работоспособность

- Загрузите в контейнер файл и скачайте с контейнера другой файл  
 add shared folder to container ubuntu "docker run -it -v /home/fill/tmp:/opt ubuntu"  
 HOST putty screenshot:

Ubuntu container Screenshot:

```
root@3f3e380d4da3: /opt
root@3f3e380d4da3: /# ls
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp
root@3f3e380d4da3: /# pwd
/
root@3f3e380d4da3: /# df -h
Filesystem      Size  Used Avail Use% Mounted on
overlay          30G   9.1G   19G   33% /
tmpfs            64M    0    64M    0% /dev
tmpfs            1.9G    0   1.9G    0% /sys/fs/cgroup
shm              64M    0    64M    0% /dev/shm
/dev/sda2        30G   9.1G   19G   33% /opt
tmpfs            1.9G    0   1.9G    0% /proc/asound
tmpfs            1.9G    0   1.9G    0% /proc/acpi
tmpfs            1.9G    0   1.9G    0% /proc/scsi
tmpfs            1.9G    0   1.9G    0% /sys/firmware
root@3f3e380d4da3: /# cd opt/
root@3f3e380d4da3: /opt# ls
upload_from_lxc
root@3f3e380d4da3: /opt# cat "upload test" > upload_from_docker.txt
cat: 'upload test': No such file or directory
root@3f3e380d4da3: /opt# echo "upload test" > upload_from_docker.txt
root@3f3e380d4da3: /opt# ls -la
total 12
drwxrwxrwx 2 root root 4096 Apr  3 16:02 .
drwxr-xr-x 1 root root 4096 Apr  3 15:59 ..
-rw-r--r-- 1 root root   12 Apr  3 16:02 upload_from_docker.txt
-rw-r--r-- 1 100000 100000  0 Apr  3 15:18 upload_from_lxc
root@3f3e380d4da3: /opt# ls
download_to_docker.txt  upload_from_docker.txt  upload_from_lxc
root@3f3e380d4da3: /opt# cat download_to_docker.txt
download test
root@3f3e380d4da3: /opt# ^C
root@3f3e380d4da3: /opt#
```

8. Прочитать документацию и кратко описать основные 7 команд Dockerfile

Команда **FROM** — данную команду можно назвать одной из самых необходимых при создании Докерфайла. Она определяет базовый образ для начала процесса построения контейнера. Это может быть любой образ, в том числе и созданные вами до этого. Если указанный вами образ не найден на хосте, Докер попытается найти и скачать его. Данная команда в Докерфайле всегда должна быть указана первой

Команда **MAINTAINER** — данная команда не является исполняемой, и просто определяет значение поля автора образа. Лучше всего ее указывать сразу после команды FROM.

Команда **ADD** — данная команда берет два аргумента, путь откуда скопировать файл и путь куда скопировать файлы в собственную файловую систему контейнера. Если же source путем является **URL** (т.е адрес веб-страницы) — то вся страница будет скачена и помещена в контейнер.

Команда **VOLUME** — данная команда используется для организации доступа вашего контейнера к директории на хосте (тоже самое, что и монтирование директории)

Команда **RUN** - является основной командой для исполнения команд при написании Докерфайла. Она берет команду как аргумент и запускает ее из образа. В отличие от CMD данная команда используется для построения образа (можно запустить несколько RUN подряд, в отличие от CMD)

Команда **ENV** используется для установки переменных среды (одной или многих). Данные переменные выглядят следующим образом «ключ = значение» и они доступны внутри контейнера скриптам и различным приложениям. Данный функционал Докера, по сути, очень сильно увеличивает гибкость в плане различных сценариев запуска приложений.

Команда **WORKDIR** указывает директорию, из которой будет выполняться команда CMD.

Команда **CMD** — похожая на команду RUN, используется для выполнения определенных программ, но, в отличие от RUN данная команда обычно применяется для запуска/инициации приложений или команд уже после их установки с помощью RUN в момент построения контейнера.

Работа с Kubernetes в Ubuntu

<https://ubuntu.com/kubernetes/install> ; <https://microk8s.io/docs/>

1. Установить microk8s (screenshot)
2. Проверьте статус (screenshot) и команды менеджера кластера (screenshot).
3. Просмотрите установленные в докере образы; заверните один из них в образ \*.tar
4. Импортируйте образ в Kubernetes (screenshot)
5. Запустите образ и убедитесь, что он работает. (screenshot)