

ПР №1 РАЗРАБОТКА КЛИЕНТ-СЕРВЕРНЫХ ПРИЛОЖЕНИЙ

Прежде всего продемонстрируем, как удалить пакеты старых версий Docker с именами `docker`, `docker.io` или `docker-engine` (содержимое директории `/var/lib/docker/`, включая образы, контейнеры, тома будет сохранено). Для этого в терминале ВМ выполним команду:

```
$ sudo apt-get remove docker docker-engine docker.io containerd runc
```

Если на созданной ВМ Docker ранее не устанавливалась, получаем ожидаемый вывод, а именно

```
«Невозможно найти пакет docker-engine»
```

Далее рассмотрим процесс установки, используя официальный репозиторий Docker. Это самый простой способ. Выполняем следующие команды:

1. Обновляем индексацию пакетов

```
$ sudo apt-get update
```

2. Устанавливаем пакеты, позволяющие менеджеру пакетов `apt` взаимодействовать с репозиторием по протоколу HTTPS

```
$ sudo apt-get install \  
apt-transport-https \  
ca-certificates \  
curl \  
gnupg-agent \  
software-properties-common
```

3. Добавляем официальный GPG–ключ репозитория Docker

```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

Далее необходимо проверить что получен следующий ключ

```
9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88
```

для чего введем команду поиска по последним 8 символам

```
$ sudo apt-key fingerprint 0EBFCD88
```

В результате получается вывод, представленный на Рисунке 1.

```
zheny@zhenys-VM: ~  
Setting up software-properties-common (0.99.13.1) ...  
Setting up software-properties-gtk (0.99.13.1) ...  
Processing triggers for dbus (1.12.20-2ubuntu2) ...  
Processing triggers for shared-mime-info (2.1-1) ...  
Processing triggers for mailcap (3.69ubuntu1) ...  
Processing triggers for desktop-file-utils (0.26-1ubuntu2) ...  
Processing triggers for hicolor-icon-theme (0.17-2) ...  
Processing triggers for gnome-menus (3.36.0-1ubuntu1) ...  
Processing triggers for libgl1b2.0-0:amd64 (2.68.4-1ubuntu1) ...  
Processing triggers for man-db (2.9.4-2) ...  
zheny@zhenys-VM:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo  
o apt-key add -  
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see  
apt-key(8)).  
OK  
zheny@zhenys-VM:~$ sudo apt-key fingerprint 0EBFCD88  
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see  
apt-key(8)).  
pub rsa4096 2017-02-22 [SCEA]  
9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88  
uid [ unknown] Docker Release (CE deb) <docker@docker.com>  
sub rsa4096 2017-02-22 [S]  
zheny@zhenys-VM:~$
```

Рисунок 1 – Вывод команды проверки ключа

Далее добавляем репозиторий stable, Рисунок 2.

```
$ sudo add-apt-repository \  
"deb [arch=amd64] https://download.docker.com/linux/ubuntu \  
$(lsb_release -cs) \  
stable"
```

```
zheny@zhenys-VM:~$ sudo apt-key fingerprint 0EBFCD88  
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).  
pub rsa4096 2017-02-22 [SCEA]  
9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88  
uid [ unknown] Docker Release (CE deb) <docker@docker.com>  
sub rsa4096 2017-02-22 [S]  
zheny@zhenys-VM:~$ sudo add-apt-repository \  
"deb [arch=amd64] https://download.docker.com/linux/ubuntu \  
$(lsb_release -cs) \  
stable"  
Repository: 'deb [arch=amd64] https://download.docker.com/linux/ubuntu inipish stable'  
Description:  
Archive for codename: inipish components: stable  
More info: https://download.docker.com/linux/ubuntu  
Adding repository.  
Press [ENTER] to continue or Ctrl-C to cancel.  
Adding deb entry to /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list  
Adding disabled deb-src entry to /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list  
Ign:1 https://packages.erlang-solutions.com/ubuntu focal InRelease  
Get:2 https://packages.erlang-solutions.com/ubuntu focal Release [9132 B]  
Get:3 https://packages.erlang-solutions.com/ubuntu focal Release.gpg [833 B]  
Hit:4 http://security.ubuntu.com/ubuntu inipish-security InRelease  
Hit:5 https://download.docker.com/linux/ubuntu focal InRelease  
Ign:3 https://packages.erlang-solutions.com/ubuntu focal Release.gpg  
Hit:6 https://download.docker.com/linux/ubuntu inipish InRelease  
Hit:7 http://ru.archive.ubuntu.com/ubuntu inipish InRelease  
Hit:8 http://ru.archive.ubuntu.com/ubuntu inipish-updates InRelease  
Hit:9 http://ru.archive.ubuntu.com/ubuntu inipish-backports InRelease  
Reading package lists... Done  
W: GPG error: https://packages.erlang-solutions.com/ubuntu focal Release: The following signatures couldn't be verified because the public key is not available: NO_PUBKEY D208507CA14F4FCA  
E: The repository 'https://packages.erlang-solutions.com/ubuntu focal Release' is not signed.  
N: Updating from such a repository can't be done securely, and is therefore disabled by default.  
N: See apt-secure(8) manpage for repository creation and user configuration details.  
M: Target Packages (stable/binary-amd64/Packages) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target Packages (stable/binary-all/Packages) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target Translations (stable/i18n/Translation-en_US) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target Translations (stable/i18n/Translation-en) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target DEP-11 (stable/deb11/Components-amd64.yml) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target DEP-11 (stable/deb11/Components-all.yml) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target DEP-11-Icons (stable/deb11/Icons-48x48.tar) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target DEP-11-Icons (stable/deb11/Icons-64x64.tar) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target DEP-11-Icons-hidpi (stable/deb11/Icons-64x64-hidpi.tar) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target CNF (stable/cnf/Commands-amd64) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target CNF (stable/cnf/Commands-all) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target Packages (stable/binary-amd64/Packages) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target Packages (stable/binary-all/Packages) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target Translations (stable/i18n/Translation-en_US) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target Translations (stable/i18n/Translation-en) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target DEP-11 (stable/deb11/Components-amd64.yml) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target DEP-11 (stable/deb11/Components-all.yml) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target DEP-11-Icons (stable/deb11/Icons-48x48.tar) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target DEP-11-Icons (stable/deb11/Icons-64x64.tar) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target DEP-11-Icons-hidpi (stable/deb11/Icons-64x64-hidpi.tar) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target CNF (stable/cnf/Commands-amd64) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1  
M: Target CNF (stable/cnf/Commands-all) is configured multiple times in /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-inipish.list:3 and /etc/apt/sources.list.d/docker.list:1
```

Рисунок 2 – Вывод команды добавления репозитория

После этого можно перейти к установке Docker Engine – Community, для этого:

1. Обновляем индексацию пакетов

```
$ sudo apt-get update
```

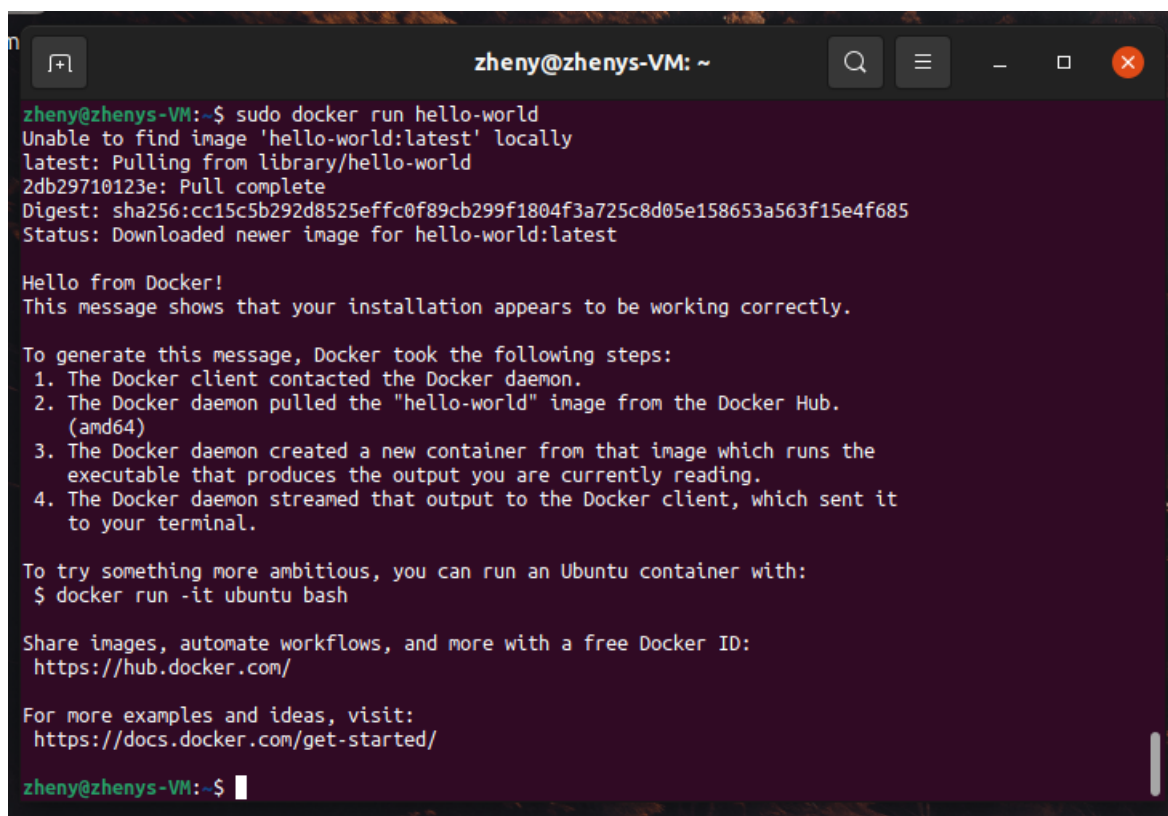
2. Устанавливаем последнюю версию docker-ce (собственно, Docker Engine – Community), docker-ce-cli (интерфейс клиента) и containerd.io (исполняемая среда для запуска контейнеров).

```
$ sudo apt-get install docker-ce docker-ce-cli containerd.io
```

После, необходимо убедиться, что Docker Engine – Community установлен корректно и работоспособен. Следующая команда загружает тестовый образ в контейнер и запускает его на выполнение.

\$ sudo docker run hello-world

Пример результата выполнения этой команды приведен на Рисунке 3. Он показывает характерную последовательность из 4 шагов.



```
zheny@zhenys-VM: ~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:cc15c5b292d8525effc0f89cb299f1804f3a725c8d05e158653a563f15e4f685
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

zheny@zhenys-VM: ~$
```

Рисунок 3 – Проверка работоспособности Docker Engine – Community

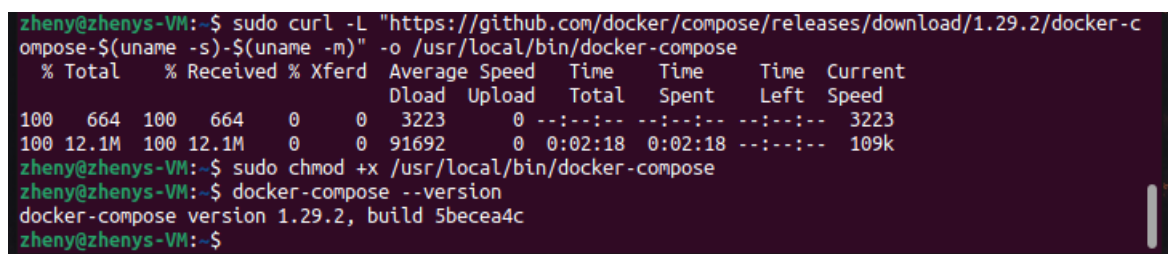
Далее устанавливаем средство docker-compose и проверим установленную версию. Сначала по ссылке определяем номер последней версии. К примеру установим версию 1.29.2.

<https://github.com/docker/compose/releases>

Это значение будет фигурировать в последующей команде, Рисунок 4.

\$ sudo curl -L

"https://github.com/docker/compose/releases/download/1.29.2/docker-compose-\$(uname -s)-\$(uname -m)" -o /usr/local/bin/docker-compose

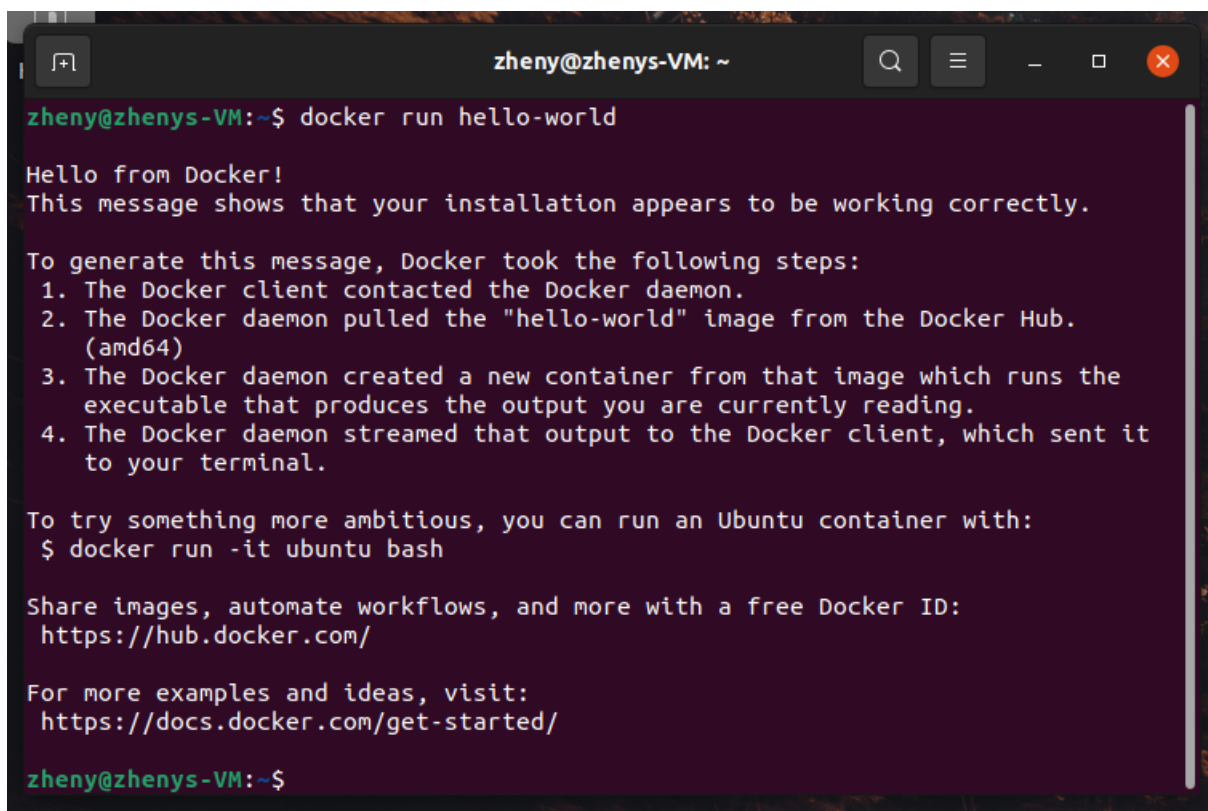


```
zheny@zhenys-VM: ~$ sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-c
ompose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
% Total % Received % Xferd Average Speed Time Time Current
Dload Upload Total Spent Left Speed
100 664 100 664 0 0 3223 0 --:--:-- --:--:-- --:--:-- 3223
100 12.1M 100 12.1M 0 0 91692 0 0:02:18 0:02:18 --:--:-- 109k
zheny@zhenys-VM: ~$ sudo chmod +x /usr/local/bin/docker-compose
zheny@zhenys-VM: ~$ docker-compose --version
docker-compose version 1.29.2, build 5becea4c
zheny@zhenys-VM: ~$
```

Рисунок 4 – Вывод установки и проверка установленной версии Docker-compose

В процессе установки была создана группа docker, но пользователи в нее не добавлены. Поэтому для ввода любой docker команды необходимо использовать sudo с последующим вводом пароля, т.к. Docker предполагает взаимодействие только с привилегированным пользователем root.

Осуществим вывод команды docker run hello-world без sudo после перезагрузки ВМ, чтобы убедиться, что все работает корректно, пример представлен на Рисунке 5. Docker первоначально не смог найти локальный образ hello-world, поэтому он загрузил образ из Docker Hub, который является репозиторием по умолчанию. После того как образ был загружен, Docker создал контейнер из образа, а приложение внутри контейнера было исполнено, отобразив сообщение.



```
zheny@zhenys-VM: ~  
zheny@zhenys-VM:~$ docker run hello-world  
  
Hello from Docker!  
This message shows that your installation appears to be working correctly.  
  
To generate this message, Docker took the following steps:  
1. The Docker client contacted the Docker daemon.  
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.  
   (amd64)  
3. The Docker daemon created a new container from that image which runs the  
   executable that produces the output you are currently reading.  
4. The Docker daemon streamed that output to the Docker client, which sent it  
   to your terminal.  
  
To try something more ambitious, you can run an Ubuntu container with:  
$ docker run -it ubuntu bash  
  
Share images, automate workflows, and more with a free Docker ID:  
https://hub.docker.com/  
  
For more examples and ideas, visit:  
https://docs.docker.com/get-started/  
  
zheny@zhenys-VM:~$
```

Рисунок 5 – Вывод команды docker run hello-world без sudo после перезагрузки ВМ

Дополнительно задание:

Добавить в группу docker непривилегированного пользователя.