

Инструкция к занятию "Cloudera и Hortonworks"

1. Создадим новую виртуальную машину со следующими параметрами

Название ?

Регион ? **Зона** ?

us-central1 (Айова) ▼ us-central1-a ▼

Тип машины
Чтобы выбрать количество ядер, объем памяти и количество графических процессоров, нажмите "Настроить".


2 виртуальных... ▼ 7,5 ГБ памяти [Настроить](#)

Если вы [перейдете на платный аккаунт](#), то сможете создавать экземпляры с количеством ядер до 96.

Контейнер ?

Развернуть образ контейнера в этом экземпляре VM. [Подробнее...](#)

Загрузочный диск ?

 Новый стандартный постоянный диск объемом 30 ГБ
Образ
Ubuntu 18.04 LTS [Изменить](#)

Профиль и API-доступ ?

Сервисный аккаунт ?

Compute Engine default service account ▼

Область действия доступа ?

Доступ по умолчанию
 Полный доступ ко всем Cloud API
 Настроить доступ к каждому API

Брандмауэр ?
Чтобы разрешить определенный трафик из Интернета, добавьте теги и правила брандмауэра.

Разрешить трафик HTTP
 Разрешить трафик HTTPS

[Настройка параметров управления, безопасности, дисков, сети и многоклиентского обслуживания](#)

Для оплаты этого экземпляра VM будут использованы бонусные средства.
[Бесплатный план GCP](#) ↗

[Создать](#) [Отмена](#)

2. Установим Docker: `curl -sSL https://get.docker.com/ | sh`
3. Скачаем докер-образ Cloudera QuickstartVM:

- a. `sudo wget`
https://downloads.cloudera.com/demo_vm/docker/cloudera-quickstart-vm-5.13.0-0-beta-docker.tar.gz
 - b. `tar xzf cloudera-quickstart-vm-*-docker.tar.gz`
 - c. `rm -f cloudera-quickstart-vm-5.13.0-0-beta-docker.tar.gz`
 - d. `sudo docker import`
`cloudera-quickstart-vm-5.13.0-0-beta-docker/cloudera-quickstart-vm-5.13.0-0-beta-docker.tar`
4. Запустим наш докер-контейнер с Cloudera на борту:
`sudo docker run --hostname=quickstart.cloudera --privileged=true -t -i -p 8777:8888 -p 7190:7180 -p 90:80 fdf3678bea64 /usr/bin/docker-quickstart`
 (fdf3678bea64 - hash образа, он может быть у вас другим, проверить можно командой `sudo docker images`)
 5. После всех этих манипуляций у вас должна запуститься командная строка внутри контейнера
 6. (Уже внутри контейнера) `/home/cloudera/cloudera-manager --express --force`
 7. Настройте firewall, чтобы иметь доступ к веб-интерфейсам

Сеть VPC

← Создание правила брандмауэра

- Сети VPC
- Внешние IP-адреса
- Правила брандмауэра
- Маршруты
- Точки обмена трафиком V...
- Общая сеть VPC
- Бессерверный доступ к VPC

С помощью правил брандмауэра можно управлять входящим и исходящим трафиком для экземпляра. По умолчанию блокируется весь входящий трафик из-за пределов вашей сети. [Подробнее...](#)

Название ?

Описание (Необязательно)

Журналы
 В результате включения этой функции может создаваться большое количество журналов брандмауэра, что приведет к увеличению расходов в Stackdriver. [Подробнее...](#)

Вкл.
 Выкл.

Сеть ?

default

Приоритет ?
 Укажите значение в диапазоне 0–65535. [Проверить приоритет других правил брандмауэра](#)

Направление трафика ?

Входящий трафик
 Исходящий трафик

Действие ?

Разрешить
 Запретить

Целевые экземпляры ?

Все экземпляры в сети

Фильтр источника ?

По диапазонам IP-адресов

Диапазоны IP-адресов источников ?

0.0.0.0/x

Дополнительный фильтр источника ?

Нет

Создание таблицы в Hive

1. Скачайте датасет
https://storage.googleapis.com/otus_sample_data/athlete.snappy.parquet
2. Через HUE загрузите файл в папку /user/cloudera/athlete
 - a. В навигационном меню выберите "Files"
 - b. Создайте папку
 - c. Загрузите файл, нажав Upload
3. Перейдите в "Editor > Hive" и выполните запрос

```
CREATE EXTERNAL TABLE athlete (
```

```
  ID INT,
```

```
  Name STRING,
```

```
  Sex STRING,
```

```
  Age INT,
```

```
  Height INT,
```

```
  Weight INT,
```

```
  Team STRING,
```

```
  NOC STRING,
```

```
  Games STRING,
```

```
  `Year` INT,
```

```
  Season STRING,
```

```
  City STRING,
```

```
  Sport STRING,
```

```
  Event STRING,
```

```
  Medal STRING
```

```
)
```

```
STORED AS PARQUET
```

```
LOCATION '/user/cloudera/athlete'
```