

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**ОБНИНСКИЙ ИНСТИТУТ АТОМНОЙ ЭНЕРГЕТИКИ - филиал**  
Федерального государственного автономного образовательного учреждения  
высшего образования  
**«Национальный исследовательский ядерный университет «МИФИ»**  
**(ИАТЭ НИЯУ МИФИ)**

**Кафедра «Информационных систем»**

**ОТЧЕТ**  
По лабораторной работе №1

**«Экосистема Cloudera»**

Выполнил: студент группы ИС-М16  
Рябов П. В.

Проверил: Бобков И. А.  
Грицюк С.В

Обнинск – 2016 г.

В данной лабораторной работе проводилось ознакомление с экосистемой Apache Hadoop на примере дистрибутива Cloudera.

## 1. ПОДГОТОВКА К УСТАНОВКЕ

### 1.1 Установка дистрибутива Cent OS

На локальной машине через веб-интерфейс VirtualBox  
localhost/vbox/

Была создана новая конфигурация для виртуальной машины на базе открытого Red Hat Linux дистрибутива Cent OS:

Тип Подключения сети: Bridge

Объем HDD: 100GB. (Рекомендуется от 40Gb и больше)

Объем RAM: 6GB

Остальные настройки: По-умолчанию

В ходе установки был задан пароль для root: root.

В ходе послеустановочной настройки Cent OS:

Была осуществлена проверка работоспособности сети между гостевой и родительской ОС посредством команды ping. IP адрес гостевой ОС был получен командой ifconfig  
стандартный файрволл iptables был выключен командой: iptables -F

Подключение к удаленному рабочему столу осуществлялось с использованием протокола RDP через ПО Remmina; С терминала родительской ОС - посредством сетевого протокола ssh:

```
ssh 10.0.191.54 -l root
```

### 1.2 Установка Веб-интерфейса Cloudera Manager

На сайте Cloudera в разделе загрузок дистрибутива

<http://www.cloudera.com/downloads.html>

Для загрузки была выбрана последняя на момент выполнения лабораторной работы версия Cloudera Manager 5.9.0

# Download Cloudera Manager 5.9.0

The recommended tool for installing Cloudera Enterprise

## Easily Manage Hadoop in Production

Cloudera Manager makes it easy to manage Hadoop deployments of any scale in production. Quickly deploy, configure, and monitor your cluster through an intuitive UI - complete with rolling upgrades, backup and disaster recovery, and customizable alerting.

Cloudera Manager is available as an integrated and supported part of Cloudera Enterprise.

## Cloudera Manager 5.9.0

[DOWNLOAD CLOUDERA MANAGER](#)[SELECT A DIFFERENT VERSION](#)

Перед получением ссылок на скачивание были введены фейковые данные для ускорения процесса прохождения формальной регистрации. После принятия лицензионного соглашения, были получены следующие ссылки:

Thank you for choosing Cloudera Manager, your download instructions are below:

## Automated Installation

Ideal for trying Cloudera enterprise data hub, the installer will download Cloudera Manager from Cloudera's website and guide you through the **setup process**.

**Pre-requisites:** multiple, Internet-connected Linux machines, with SSH access, and significant free space in /var and /opt.

```
$ wget http://archive.cloudera.com/cm5/installer/latest/cloudera-manager-installer.bin
```

```
$ chmod u+x cloudera-manager-installer.bin
```

```
$ sudo ./cloudera-manager-installer.bin
```

Далее перед запуском установщика cloudera manager необходимо выключить систему принудительного контроля доступа SELinux, правкой параметра через любой стандартный текстовый редактор, например vi

```
vi /etc/sysconfig/selinux
```

Меняем параметр enforcing на disabled и перезагружаемся

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of these two values:
#   targeted - Targeted processes are protected,
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Также для запуска Cloudera был отредактирован файл `/etc/hosts`

- 1) Были удалены `ipv6` адреса
- 2) Имя домена и имя хоста необходимо поменять местами

```
#reboot
```

После перезагрузки и авторизации под логином `root` были выполнены команды установки по ранее полученным ссылкам с сайта cloudera. А именно:

- 1) Скачивание `bin` файла установки через клиент `wget`.
- 2) Присвоение прав на исполнение командой `chmod`.
- 3) Запуск установочного файла с правами суперпользователя через команду `sudo`

Далее командой `netstat -ln` было проверено что нужный порт для работы веб интерфейса Cloudera manager открылся (tcp port 7180).

## 2. УСТАНОВКА И НАСТРОЙКА КЛАСТЕРА

### 2.1 Работа с веб-интерфейсом Cloudera Manager

#### 2.1.1 Установка Кластера

Подключение к интерфейсу было осуществлено по адресу гостевой ОС и соответствующего порта:

```
10.0.191.54:7180
```

Авторизация проходила под стандартным логином администратора:

```
Login: admin
```

```
pass: admin
```

Далее, для установки на кластер была выбрана бесплатная версия Cloudera Express

Cloudera поддерживает развертывание на кластер многочисленное число пакетов:

**Apache Hadoop (Common, HDFS, MapReduce, YARN)**

**Apache HBase**

**Apache ZooKeeper**

**Apache Oozie**

**Apache Hive**

**Hue (Apache Licenced)**

**Apache Spark**

**Apache Flume**

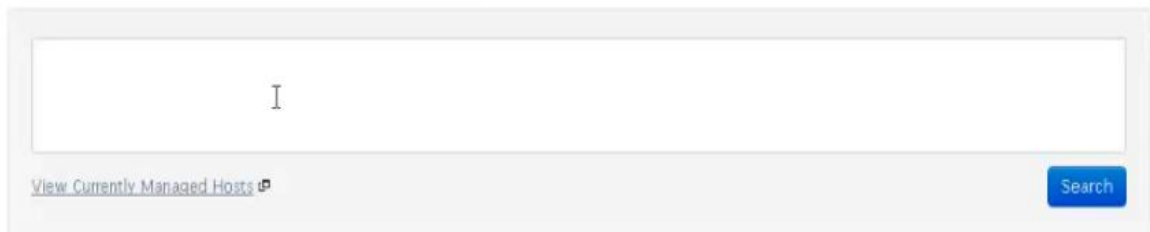
**Cloudera Impala**

**Apache Sqoop**

Далее в качестве Хоста, на который будет установлен CDH кластер была выбрана локальная машина localhost (удаленное подключение к Cent OS).

**Specify hosts for your CDH cluster installation.**

**Hint:** Search for hostnames and/or IP addresses using [patterns](#).



Далее после запуска поиска был обнаружен хост с ip: 10.0.191.54 и со статусом Ready for Installation. Был запущен процесс установки кластера.

Вся установка делится на *восемь этапов*:

**Этап 1:** Выбор версии (CDH-5.9)

**Этап 2:** Был отмечен для установки *Oracle Java SE Development Kit (JDK)*

**Этап 3:** Выбор single-user-mode: Данная опция не активировалась за ненадобностью

**Этап 4** Настройка SSH авторизации: В качестве пароля для соединения по 22 порту ssh был задан ранее зарегистрированный пароль для root (pass: root).

**Этап 5-7:** Сам процесс установки. Данный этап делится еще на несколько подэтапов:

1) Download (скачивание необходимых файлов для установки кластера)

2) Distributing (Распределение по узлам). В данной работе у нас был 1 узел - наша

локальная машина.

3) Unpacking (распаковка ранее скачанных данных)

4) Activating (активация кластера)

**Этап 8: Inspecting Hosts** (Инспектирование хостов на корректность конфигурации)

На данном этапе хосты проходят ряд валидации на предмет корректности настроек. Это необходимо для успешной работы кластера.

На этом этапе процесс установки завершается.

## 2.2.2 Настройка Кластера

Настройка делится на *шесть этапов*:

**Этап 1:** Производился выбор сервисов, которые будут установлены на кластер. В данной лабораторной работе были выбраны компоненты, входящие в редакцию **Core Hadoop + HBase, Spark**

**Cluster Setup**

Choose the CDH 5 services that you want to install on your cluster.

Choose a combination of services to install.

- ☒ **Core Hadoop**  
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, and Hue
- ☐ **Core with HBase**  
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, and HBase
- ☐ **Core with Impala**  
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, and Impala
- ☐ **Core with Search**  
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, and Solr
- ☐ **Core with Spark**  
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, and Spark
- ☐ **All Services**  
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, HBase, Impala, Solr, Spark, and Key-Value Store Indexer
- ☐ **Custom Services**  
Choose your own services. Services required by chosen services will automatically be included. Flume can be added after your initial cluster has been set up.

This wizard will also install the **Cloudera Management Service**. These are a set of components that enable monitoring, reporting, events, and alerts; these components require databases to store information, which will be configured on the next page.

☐ Include Cloudera Navigator

Back Continue

**Этап 2:** Настройка Ролей для выбранных компонентов - была выбрана по-умолчанию

**Этап 3:** Настройка подключения к Базе данных - была выбрана встроенная база(embedded base). На данном этапе был также проведен тест соединения, который завершился успешно для выбранной конфигурации базы данных.

cloudera manager Support + admin

### Cluster Setup

#### Database Setup

Configure and test database connections. If using custom databases, create the databases first according to the [Installing and Configuring an External Database](#) section of the [Installation Guide](#).

☐ Use Custom Databases  
☒ Use Embedded Database

When using the embedded database, passwords are automatically generated. Please copy them down.

##### Hive

Database Host Name:	Database Type:	Database Name:	Username:	Password:
node1.c.famous-rhythm-124921.internal	PostgreSQL	hive	hive	PdfrwbQgRXZ

##### Reports Manager

Currently assigned to run on node1.c.famous-rhythm-124921.internal.

Database Host Name:	Database Type:	Database Name:	Username:	Password:
node1.c.famous-rhythm-124921.internal	PostgreSQL	rman	rman	99XcdmE4W

[Back](#) 1 2 3 4 5 6 [Continue](#)

**Этап 4:** Более Тонкая настройка директорий, размер блока распределенной файловой системы HDFS другое. Было выбрано по-умолчанию.

#### Review Changes

<b>HDFS Block Size</b> dfs.block.size, dfs.blocksize	Cluster 1 > HDFS (Service-Wide)	128 MiB	?
<b>DataNode Failed Volumes Tolerated</b> dfs.datanode.failed.volumes.tolerated	Cluster 1 > DataNode Default Group	0	?
<b>DataNode Data Directory</b> dfs.data.dir, dfs.datanode.data.dir	Cluster 1 > DataNode Default Group	/dfs/dn	?
<b>NameNode Data Directories</b> dfs.name.dir, dfs.namenode.name.dir	Cluster 1 > NameNode Default Group	/dfs/nn	?
<b>HDFS Checkpoint Directories</b> fs.checkpoint.dir, dfs.namenode.checkpoint.dir	Cluster 1 > SecondaryNameNode Default Group	/dfs/snn	?
<b>Hive Warehouse Directory</b> hive.metastore.warehouse.dir	Cluster 1 > Hive (Service-Wide)	/user/hive/warehouse	?

[Back](#) 1 2 3 4 5 6 [Continue](#)

**Этап 5:** Deploy конфигурации клиента, а также запуск сервисов (HDFS Hive Hue YARN Oozie). Происходит в автоматическом режиме скриптом. В ходе лабораторной работы этот процесс завершился успешно.

**Этап 6:** На этом этапе процесс настройки кластера завершается.

Далее был совершен переход к управлению кластером через панель управления Cloudera Manager.

### 3. УПРАВЛЕНИЕ КЛАСТЕРОМ

#### 3.1 Сервисы Cloudera: описание и назначение

**HDFS** — это обычная файловая система, только больше. Обычная ФС, по большому счёту, состоит из таблицы файловых дескрипторов и области данных. В HDFS вместо таблицы используется специальный сервер — сервер имён (**NameNode**), а данные разбросаны по серверам данных (**DataNode**). данные разбиты на блоки (обычно по 64Мб или 128Мб), для каждого файла сервер имён хранит его путь, список блоков и их реплик. HDFS имеет классическую unix-овую древовидную структуру директорий, пользователей с `rxwx` правами, и даже схожий набор консольных команд. Сервер имён раскрывает для всех желающих **расположение блоков данных** на машинах. Почему это важно, смотрим в следующем разделе.

**YARN** - распределенный менеджер контейнеров. Или же менеджер ресурсов кластеров. Может конфигурировать пулл ресурсов между всеми фреймворками, запущенными на YARN (Например, SPARK).

При правильной архитектуре приложения, информация о том, на каких машинах расположены блоки данных, позволяет запустить на них же вычислительные процессы и выполнить большую часть вычислений **локально**, т.е. без передачи данных по сети. Именно эта идея лежит в основе парадигмы **MapReduce** и её конкретной реализации в Hadoop. Каждая MapReduce работа состоит из двух фаз:

1. *map* — выполняется параллельно и (по возможности) локально над каждым блоком данных. Вместо того, чтобы доставлять терабайты данных к программе, небольшая, определённая пользователем программа копируется на сервера с данными и делает с ними всё, что не требует перемешивания и перемещения данных (*shuffle*).
2. *reduce* — дополняет *map* агрегирующими операциями

Стандартный MapReduce спроектирован так, что все результаты — как конечные, так и промежуточные — записываются на диск. **Spark** использует идею локальности данных, однако выносит большинство вычислений в память вместо диска. Ключевым понятием в Spark-е является RDD (*resilient distributed dataset*) — указатель на ленивую распределённую коллекцию данных. Большинство операций над RDD не приводит к каким-либо вычислениям, а только создаёт очередную обёртку, обещая выполнить операции только тогда, когда они понадобятся. это фреймворк с помощью которого можно создавать приложения для распределенной обработки данных. Со своей стороны Spark предоставляет программное API для работы с данными, в которое входят: загрузка, сохранение, трансформация и агрегация, плюс множество всяких мелочей, например возможность локального запуска в целях разработки и отладки кода.

Кроме того, Spark отвечает за распределенное выполнение вашего приложения. Он сам раскидывает ваш код по всем узлам кластера, разбивает на подзадачи, создаёт план выполнения и следит за успешностью. Если на каком либо узле произошел сбой, и какая то подзадача завершилась с ошибкой, она обязательно будет перезапущена.



В инфраструктуре Hadoop есть несколько SQL-ориентированных инструментов: **Hive** - один из них. В качестве языка запросов использует HiveQL — урезанный диалект SQL, который, тем не менее, позволяет выполнять довольно сложные запросы над данными, хранимыми в HDFS.

Иногда всё-таки приходится бороться с другими проблемами, для которых лучше приспособлены NoSQL базы. В качестве NoSQL используется **HBase**.

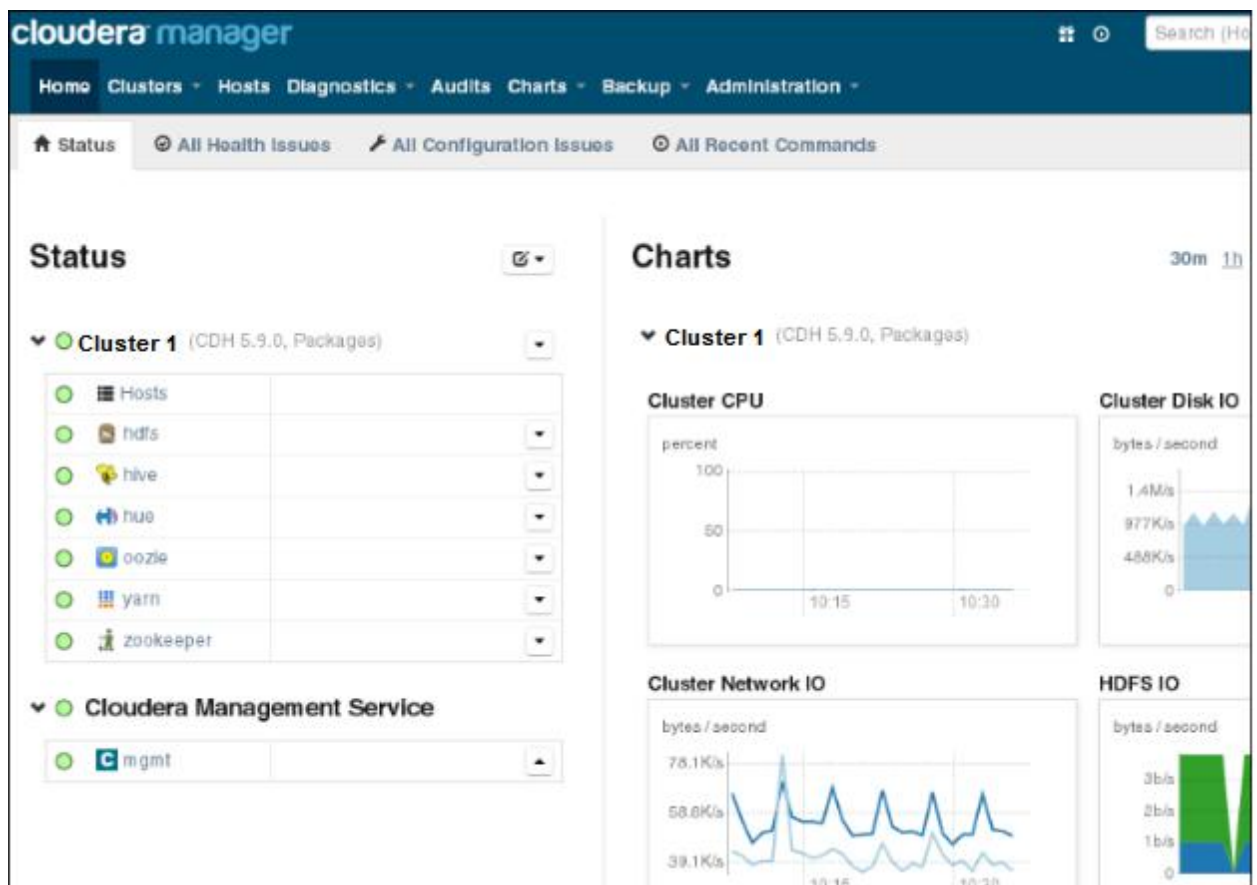
**ZooKeeper** — главный инструмент координации для всех элементов инфраструктуры Hadoop. Чаще всего используется как централизованный сервис конфигурации. Хранит данные по принципу ключ-значение. Удобно использовать для резервных копий конфигурационных файлов.

**Hue** — веб-интерфейс к сервисам Hadoop, часть Cloudera Manager.

**Oozie** — планировщик потоков задач. Изначально спроектирован для объединения отдельных MapReduce работ в единый конвейер и запуска их по расписанию.

### 3.2 Работа с Панелью управления Cloudera Manager

Панель Управления позволяет визуализировать информацию, получаемую от сервисов, развернутых на текущем кластере.



В левой панели можно увидеть статус всех развернутых на кластере сервисов. Справа же отображается визуальная информация в виде графиков, гистограмм и других графических элементов. Можно получить полную информацию о работе кластера. Его производительности и так далее. Подробная конфигурация может быть получена по открытию ZooKeeper.

Во Вкладке Hosts можно посмотреть все управляемые узлы, а также полные логи происходящих событий.

Особое внимание было уделено состоянию ролей для сервисов. (Role Instance). Например для сервиса HDFS можно добавить DataNode role instance к нашему единственному хосту. (Во вкладке Instances у HDFS пункт Add role instances).

## Заключение

В данной лабораторной работе было проведено ознакомление с экосистемой Hadoop на примере дистрибутива Cloudera. Была проведена предварительная настройка рабочей машины для установки кластера. Далее - сама установка и конфигурирование кластера, а также были изучены основные возможности сервисов Cloudera по управлению большими данными.