BIGDATA TEAM

MF-BD-2019-Q2 | User Guides

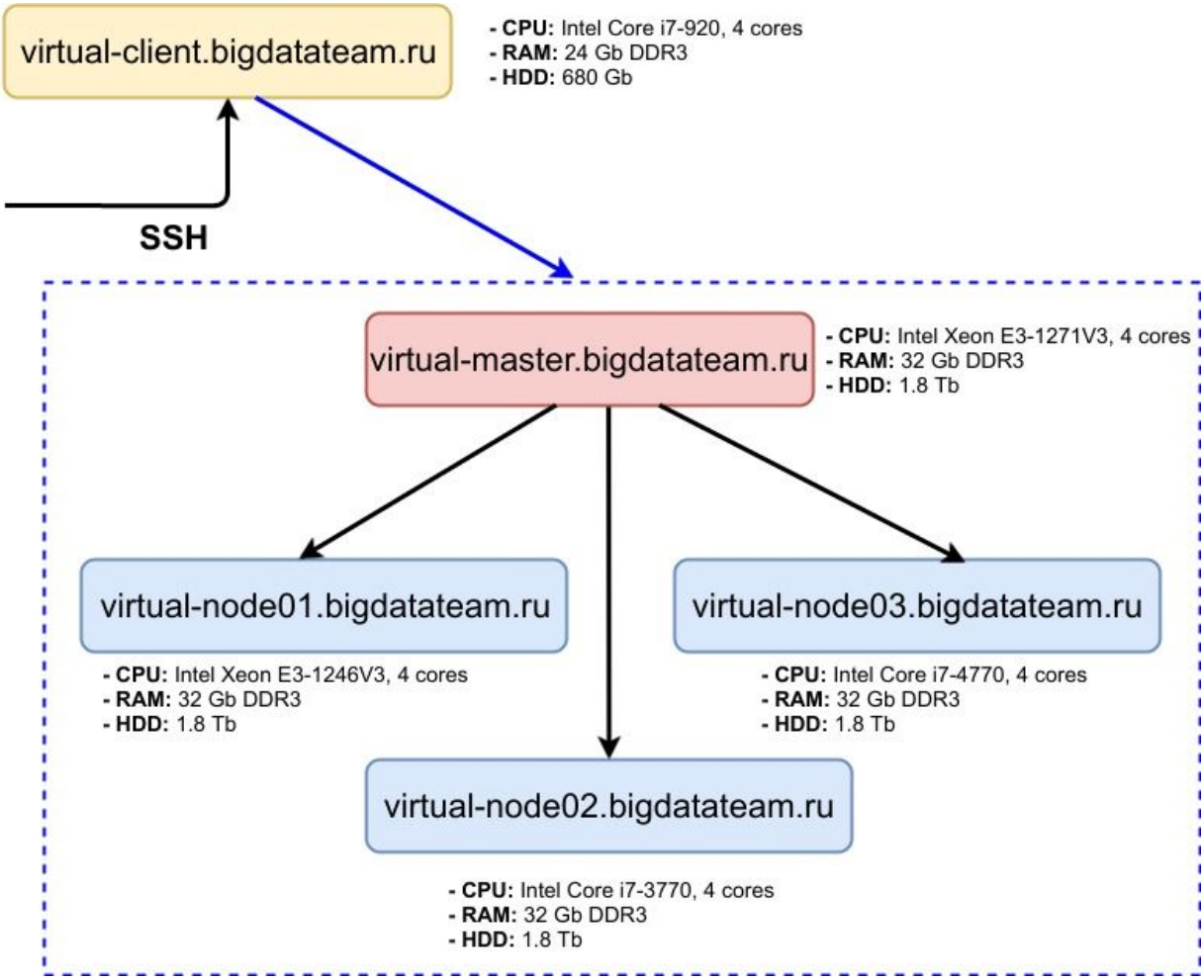
В рамках этого курса вас ожидает:

- Работа на Hadoop-кластере со сдачей заданий на программирование.
- Коммуникация в дружественной атмосфере с коллегами и преподавателями в Telegram-канале.

---

1. Описание кластера и сервисов Hadoop.

В первом приближении Hadoop кластер выглядит следующим образом (клиентский узел - virtual-client, мастер-сервер (где работает NameNode) - virtual-master, рабочие узлы кластера - virtual-node01, virtual-node02, …):



Hadoop экосистема предоставляет следующие сервисы (и соответствующие порты):

Service	Port	Доступность извне
HDFS Web UI	50070	НЕТ[1]
Resource Manager	8088	НЕТ[2]
YARN JobHistory	19888	ДА
Spark2 History server	18089	ДА

2. Доступ к кластеру и проброс портов (port forwarding).

К сожалению, бравые ребята из интернета любят взламывать Hadoop-кластер по этим портам, чтобы запускать майнинговые фермы. Поэтому, некоторые порты (e.g. порты 8088 и 50070) закрыты извне. Таким образом, необходимо использовать ssh-туннели, чтобы достучаться до Web-интерфейсов.

2.1. Unix, Linux, Mac OS.

Например, чтобы увидеть HDFS Web UI необходимо пробросить порт 50070 с мастера:

```
ssh your_login@virtual-client.bigdatateam.ru -L 50070:virtual-master:50070
```

*your_login - замените на свой логин

И пока открыта ssh-сессия Вы сможете заходить по адресу: http://localhost:50070/.

Проброс дополнительно порта осуществляется дополнительным ключом -L и значением, пример (команда ниже должна быть записана в одну строку):

```
ssh your_login@virtual-client.bigdatateam.ru -L 50070:virtual-master:50070 -L
8088:virtual-master:8088
```

Для того, чтобы удобно копировать файлы с локального ноутбука, советуем пользоваться SCP.

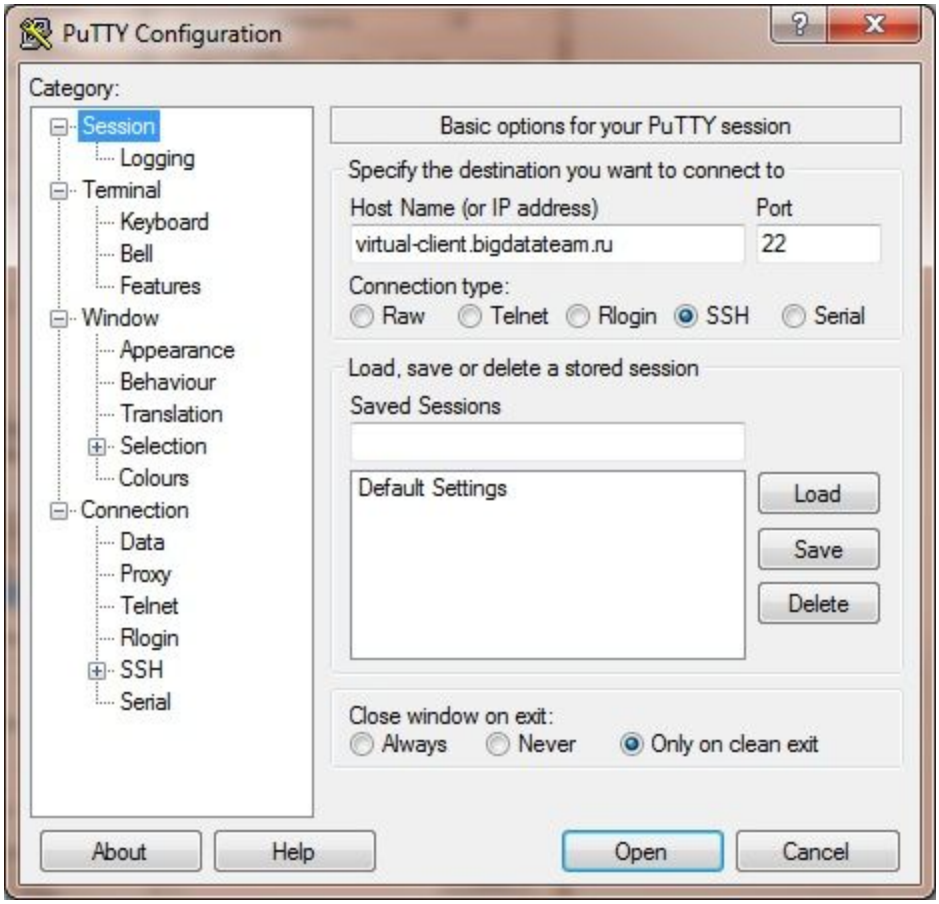
[1] См. раздел про “port forwarding”
[2] См. раздел про “port forwarding”

2.1. Windows (Putty).

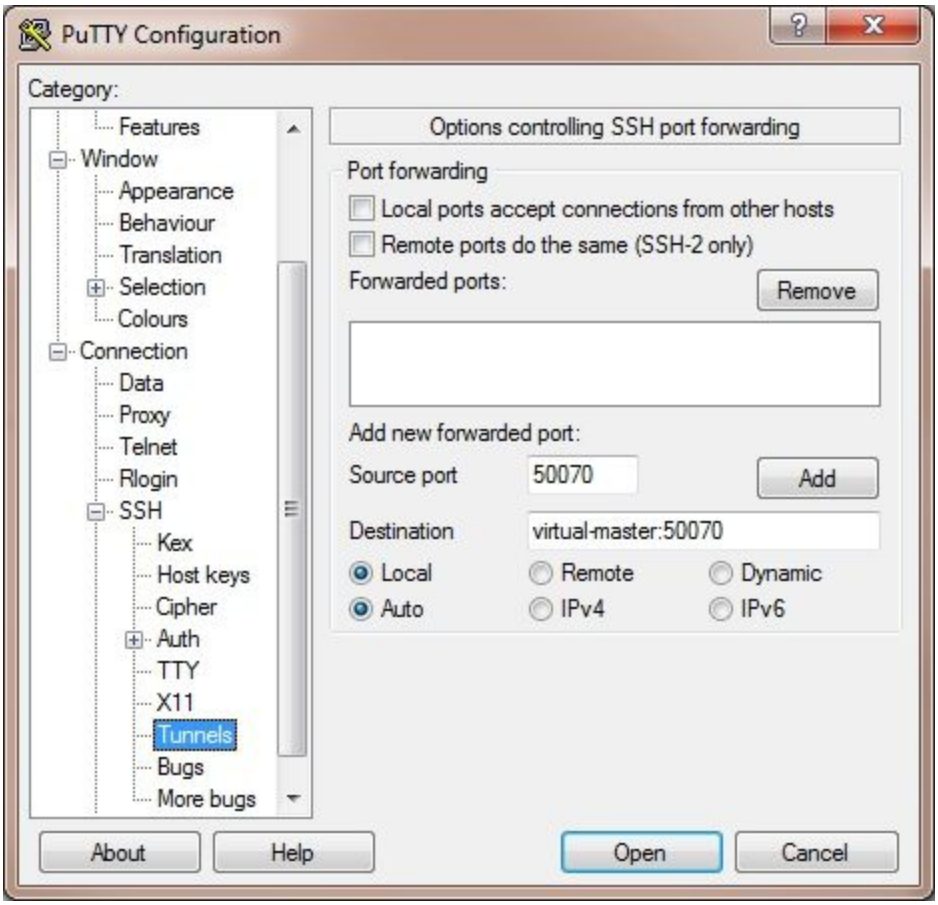
Если вы пользуетесь Windows, то жизнь у вас немного сложнее и вам необходимо правильно сконфигурировать Putty.

Необходимо настроить параметры в следующих двух категориях (выбор категории осуществляется с помощью двойного щелчка на имя категории в древовидной структуре слева):

1) Категория Session (открывается при запуске Putty), вводим “Host Name” (virtual-client.bigdatateam.ru), “Port” оставляем значение по умолчанию (22).



2) Необходимо добавить проброс портов в категории SSH->Tunnels как показано на скриншоте ниже:



Указывайте порт 50070 и/или 8088 в зависимости от того, какой UI нужен. После этого необходимо нажать кнопку “Add”.

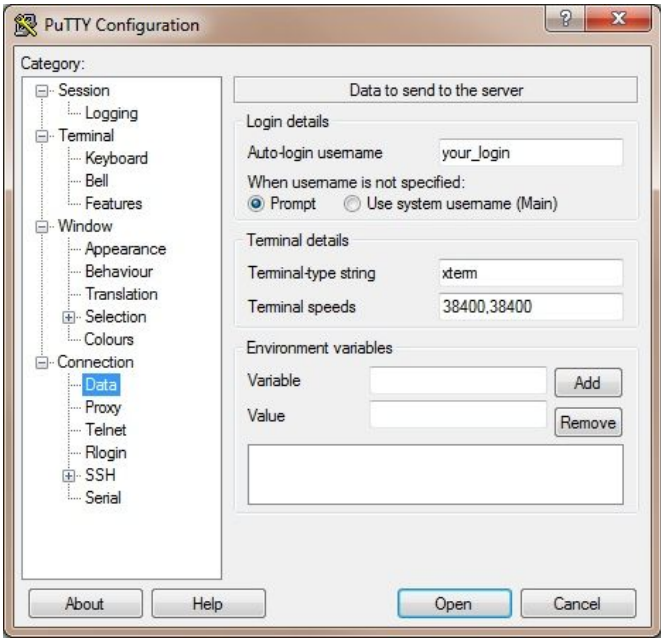
Для проброса нескольких портов еще раз укажите нужны source port и destination, а затем снова нажмите кнопку “Add”.

На данный момент вы уже можете (но прежде прочтите следующую страницу) нажать кнопку “Open” и затем открыть выбранный UI через localhost в браузере:

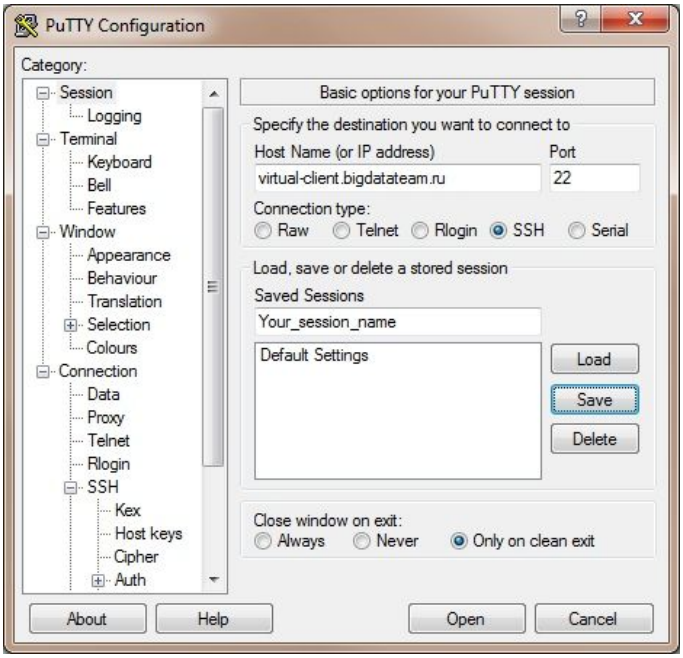
- http://localhost:50070/
- http://localhost:8088/

Но чтобы не настраивать все снова можно проделать следующие шаги для сохранения настроек:

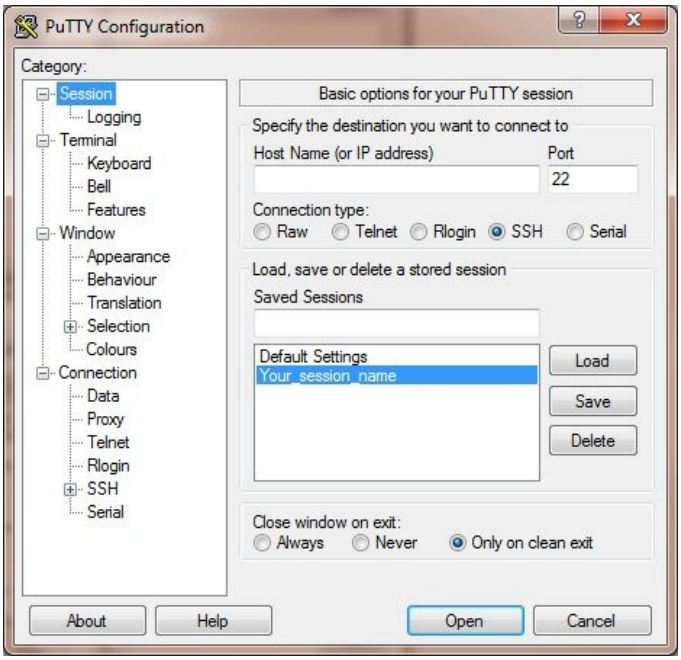
3) Позволяем Putty запомнить ваш логин, для этого вводим его в поле “Auto-login username”:



4) Чтобы непосредственно сохранить (перезаписать) настройки, необходимо ввести имя сессии (там, где на скриншоте написано “Your_session_name”) и нажать на “Save”:



5) Чтобы запустить с сохраненным параметрами можно дважды нажать на имя сессии ИЛИ нажать на имя сессии так, чтобы имя стало выделено синим, и затем на кнопку “Open”. Чтобы сохранить отредактированные параметры, необходимо нажать на имя сессии и кнопку “Load”, а после редактирования перезаписать.



Для того, чтобы удобно копировать файлы с локального ноутбука, советуем пользоваться PSCP.

3. Инструкция по работе с Apache Spark.

Для запуска Spark через Jupyter необходимо выполнить следующие шаги:

1. Усановить SSH-соединение к серверу virtual-client.bigdatateam.ru
2. В терминале выполнить следующую команду **(все в одной строке)**

```
PYSPARK_DRIVER_PYTHON=jupyter PYSPARK_DRIVER_PYTHON_OPTS='notebook
--ip=0.0.0.0 --NotebookApp.token= --port=<порт №1>' pyspark2 --conf
spark.ui.port=<порт №2>
```

3. Открыть в браузере `virtual-client.bigdatateam.ru:<порт №1>`

3.1. Соответствие логина и портов.

Hetzner Login	port_1	port_2
mf_begunov	10501	10601
mf_botvinkin	10502	10602
mf_vasilyev	10503	10603
mf_gorban	10504	10604
mf_goryacheva	10505	10605
mf_dzyatko	10506	10606
mf_kazyulin	10507	10607
mf_kirilin	10508	10608
mf_kozhevnikov	10509	10609
mf_kopin	10510	10610
mf_kostenev	10511	10611
mf_kosheleva	10512	10612
mf_kuznetsov	10513	10613
mf_lapteva	10514	10614
mf_morozov	10515	10615
mf_panov	10516	10616
mf_ponomarev	10517	10617
mf_popov	10518	10618
mf_seleznev	10519	10619
mf_stepanov	10520	10620
mf_tuvaleva	10521	10621
mf_tyukavin	10522	10622
mf_filimonova	10523	10623
mf_kholodov	10524	10624
mf_khuzhina	10525	10625
mf_chernyshev	10526	10626
mf_shadrina	10527	10627
mf_shelepanov	10528	10628
mf_shinkarenko	10529	10629

mf_surname	10500	10600

3.2. Документация по Spark

PySpark: https://spark.apache.org/docs/latest/api.html
Python API: https://spark.apache.org/docs/latest/api/python/pyspark.html
PySpark SQL API: https://spark.apache.org/docs/latest/api/python/pyspark.sql.html

4. Оконные функции

Про оконные функции в Hive можно:

- Послушать 5 минут видео на Coursera “Hive PTF (Window Functions)” (курс “Big Data Analysis: Hive, Spark SQL, DataFrames and GraphFrames”, https://www.coursera.org/learn/big-data-analysis)
- Почитать официальную документацию Hive: https://cwiki.apache.org/confluence/display/Hive/LanguageManual+WindowingAndAnalytics

По факту синтаксис оконных функций в Hive не отличается от синтаксиса в реляционных БД, а материалов и документации в Hive на эту тему мало. Поэтому имеет смысл изучить материалы про оконные функции, доступные для реляционных баз данных. Например:

- Как посчитать всё на свете одним SQL-запросом. Оконные функции PostgreSQL. https://habr.com/ru/post/268983/

5. FAQ

Q: Каким образом с помощью hdfs CLI узнать адрес Namenode (и других конфигурационных параметров)?
A:
В hdfs CLI есть модуль getconf, с помощью которого можно узнать значения конфигурационных параметров HDFS:

```
aadral@virtual-client:~$ hdfs getconf -namenodes
virtual-master.bigdatateam.ru
```

Значение любой переменной можно получить с помощью ключа -getconf, список параметров можно найти на сайте:

- https://hadoop.apache.org/docs/r2.4.1/hadoop-project-dist/hadoop-hdfs/hdfs-default.xml

Например получим стандартный размер блока HDFS:

```
aadral@virtual-client:~$ hdfs getconf -confKey dfs.blocksize
33554432
```
[3]

Q: Хочу узнать больше про архитектуру HDFS и состояния реплики блока. Что почитать?
A:
Очень понятное объяснение про состояния реплик и блоков, а также механизмы восстановления после сбоев доступно в следующей работе:

Hairong Kuang, Konstantin Shvachko, Nicholas Sze, Sanjay Radia, Robert Chansler
Yahoo! HDFS team 08/06/2009
http://files.cnblogs.com/files/inuyasha1027/appenddesign3.pdf

Еще одна приятная работа про пределы масштабирования HDFS написана нашим соотечественников Константином Швачко (он же соавтор предыдущей работы):

HDFS Scalability: The Limits to Growth
Author(s): Konstantin V. Shvachko
USENIX, Article Section: DISTRIBUTED SYSTEMS
April 2010, Volume 35, Number 2
http://c59951.r51.cf2.rackcdn.com/5424-1908-shvachko.pdf

Q: Как посмотреть в Hive используемую базу данных?
A: `set hive.cli.print.current.db=true;`

Q: Пишу в консоли скрипты MapReduce или Hive-запросы. Задача не выполняется и пишет странные ошибки, как убедиться, что у меня в коде нет “плохих” Unicode-символов, которые не видно глазом?
A: см. ресурс https://www.soscisurvey.de/tools/view-chars.php

[3] 32 MB