

## Домашнее задание №2

БПИ227 Артемьев Александр

Следующая инструкция представляет из себя пошаговое руководство по разворачиванию YARN. После выполнения первого дз у нас уже есть развернутый на 3 нодах hdfs и узлы со следующими адресами:

jn\_glob\_ip – узел для входа в jn  
jn\_local\_ip – локальный адрес jn  
nn\_ip – локальный адрес nn  
dn-00\_ip – локальный адрес dn-00  
dn-01\_ip – локальный адрес dn-01  
user\_name – имя пользователя, который может подключиться к jn

Порядок действий:

1) Подключаемся к edge-node:

```
ssh <user_name>@<jn_glob_ip>
```

2) Переходим на пользователя hadoop:

```
sudo -i -u hadoop
```

3) Сейчас для успешного последующего запуска yarn – наша цель очистить результаты предыдущих запусков hdfs.

Для этого выполним на каждой из 3 нод следующую команду (перемещаемся между нодами через `ssh <название ноды>`):

```
rm -r /tmp/hadoop-hadoop/
```

4) Теперь форматируем и создаем файловую систему на nn

```
cd hadoop-3.4.0/  
bin/hdfs namenode -format
```

5) Запускаем hdfs

```
sbin/start-dfs.sh
```

6) Теперь для работы веб-интерфейса hdfs настроим nginx:

Перейдем в jn(`exit` или `ssh jn`)

7) Посмотрим есть ли у нас уже конфигурация для веб интерфейса:

```
ls /etc/nginx/sites-available/
```

Если конфигурации нет, то выйдем с пользователя hadoop и создадим копию default под названием nn в директории `/etc/nginx/sites-available/`

```
cd /etc/nginx/sites-available/  
sudo cp ./default ./nn
```

Затем изменим конфиг nn следующим образом:

```
sudo vim nn
```

```
aleksandrartemev — team@team-1-jn: /etc/nginx/sites-available — ssh team@176.109.91.3 — 113x46
# Default server configuration
#
server {
    listen 9870;
    #listen [::]:80 default_server;

    # SSL configuration
    #
    # listen 443 ssl default_server;
    # listen [::]:443 ssl default_server;
    #
    # include snippets/snakeoil.conf;

    root /var/www/html;

    # Add index.php to the list if you are using PHP
    index index.html index.htm index.nginx-debian.html;

    server_name _;

    location / {
        auth_basic "Administrator's Area";
        auth_basic_user_file /etc/.htpasswd;
        proxy_pass http://nn:9870;
    }

    # pass PHP scripts to FastCGI server
    #
    #location ~ \.php$ {
    #    include snippets/fastcgi-php.conf;
    #
    #    # With php-fpm (or other unix sockets):
    #    fastcgi_pass unix:/run/php/php7.4-fpm.sock;
    #    # With php-cgi (or other tcp sockets):
    #    fastcgi_pass 127.0.0.1:9000;
    #}
    # deny access to .htaccess files, if Apache's document root
    # concurs with nginx's one
    #
    #location ~ /\.ht {
    #    deny all;
    #}
}

-- INSERT --
58,4-11 50%
```

8) После настройки nn выходим из облака и получаем доступ с помощью ssh туннеля:  
`ssh -L 9870:nn:9870 <user_name>@<jn_glob_ip>`  
После чего на localhost:9870 будет веб-интерфейс NameNode.

9)Перейдем к настройке YARN, как и в первом дз настроим конфиги на jm hadoop, а затем раскинем их по всем узлам. Перейдем в папку с конфигами:

```
sudo -i -u hadoop
cd hadoop-3.4.0/etc/hadoop/
```

10) Редактируем первый конфиг:

```
vim yarn-site.xml
```

```
<configuration>
<!-- Site specific YARN configuration properties -->
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
  <property>
```

```

        <name>yarn.nodemanager.env-whitelist</name>
        <value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,H
        ADOOP_CONF_DIR,CLA
        SSPATH_PREPEND_DISTCACHE,HADOOP_YARN_HOME,HADOOP_HOME,
        PATH,LANG, TZ,HADOOP_MAPRED_HOME
    </value>
</property>
<property>
    <name>yarn.resourcemanager.hostname</name>
    <value>nn</value>
</property>
<property>
    <name>yarn.resourcemanager.address</name>
    <value>nn:8032</value>
</property>
<property>
    <name>yarn.resourcemanager.resource-tracker.address</name>
    <value>nn:8031</value>
</property>
</configuration>

```

11) Редактируем второй конфиг

```
vim mapred-site.xml
```

```

<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
  <property>
    <name>mapreduce.application.classpath</name>
    <value>$HADOOP_HOME/share/hadoop/mapreduce/*:$HADOOP_HOME/share/h
    adoop/mapreduce/lib*
  </value>
  </property>
</configuration>

```

12) Теперь разошлем конфиги по нашим нодам:

```

scp mapred-site.xml dn-00:/home/hadoop/hadoop-3.4.0/etc/hadoop
scp mapred-site.xml dn-01:/home/hadoop/hadoop-3.4.0/etc/hadoop
scp mapred-site.xml nn:/home/hadoop/hadoop-3.4.0/etc/hadoop

```

```

scp yarn-site.xml nn:/home/hadoop/hadoop-3.4.0/etc/hadoop
scp yarn-site.xml dn-01:/home/hadoop/hadoop-3.4.0/etc/hadoop
scp yarn-site.xml dn-00:/home/hadoop/hadoop-3.4.0/etc/hadoop

```

13) И наконец запустим YARN

```

ssh nn
hadoop-3.4.0/sbin/start-yarn.sh

```

14) Запустим historyserver

```
mapred --daemon start historyserver
```

15) Следующая цель настроить веб интерфейс для YARN, поэтому сначала перейдем на jn и выйдем из hadoop.

```
exit  
exit
```

16) Для настройки веб интерфейса, возьмем конфиг и добавим в нем нужные порты:  
конфиг для ResourceManager

```
sudo cp /etc/nginx/sites-available/nn /etc/nginx/sites-available/ya
```

конфиг для HistoryServer

```
sudo cp /etc/nginx/sites-available/nn /etc/nginx/sites-available/dh
```

Исправляем порты на 8088

```
sudo vim /etc/nginx/sites-available/ya
```

Исправляем порты на 19888

```
sudo vim /etc/nginx/sites-available/dh
```

17) Делаем конфигурации активными

```
sudo ln -s /etc/nginx/sites-available/ya /etc/nginx/sites-enabled/ya
```

```
sudo ln -s /etc/nginx/sites-available/dh /etc/nginx/sites-enabled/dh
```

18) Команда для nginx обновить конфиги

```
sudo systemctl reload nginx
```

19) И в конце пересоздаем сессию с сервером добавив 2 порта

```
exit
```

```
ssh -L 9870:nn:9870 -L 8088:nn:8088 -L 19888:nn:19888 <user_name>@<jn_glob_ip>
```

Мои данные:

Env team-1

User team

пароль для входа x=T35T\_sMdm4

узел для входа 176.109.91.3

jn 192.168.1.6

nn 192.168.1.7

dn-00 192.168.1.8

dn-01 192.168.1.9