Luiz Vinicius Vidal, Diego Lopes, Josiel Sola

Soluções Emergentes

Passo à passo de atualização do CDH da versão 5 para a versão 6  
 Compwire Soluções em Tecnologia

**Atualização Cloudera Distributed Hadoop  
versão 5 para versão 6**

Big Data & Analytics

****

# Atualizando o CDH da versão 5 para 6

## Coletar informações para realizar o Backup

### **Informações sobre o SO**

[root@node0 ~]# lsb\_release -a

LSB Version: :core-4.1-amd64:core-4.1-noarch

Distributor ID: CentOS

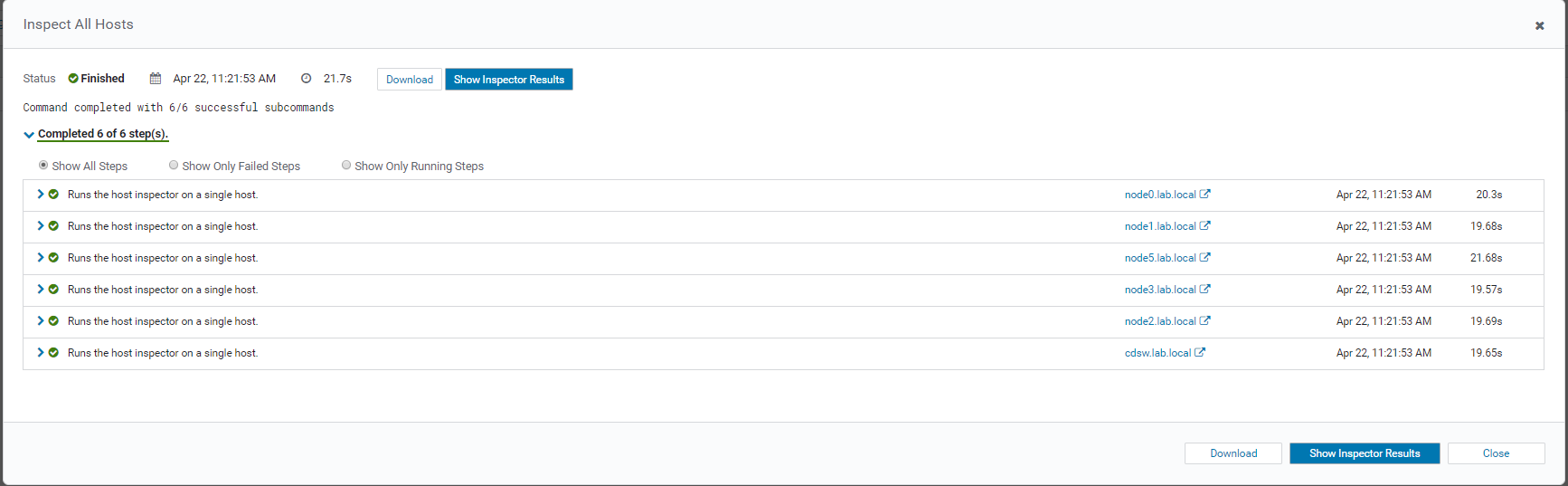
Description: CentOS Linux release 7.6.1810 (Core)

Release: 7.6.1810

Codename: Core

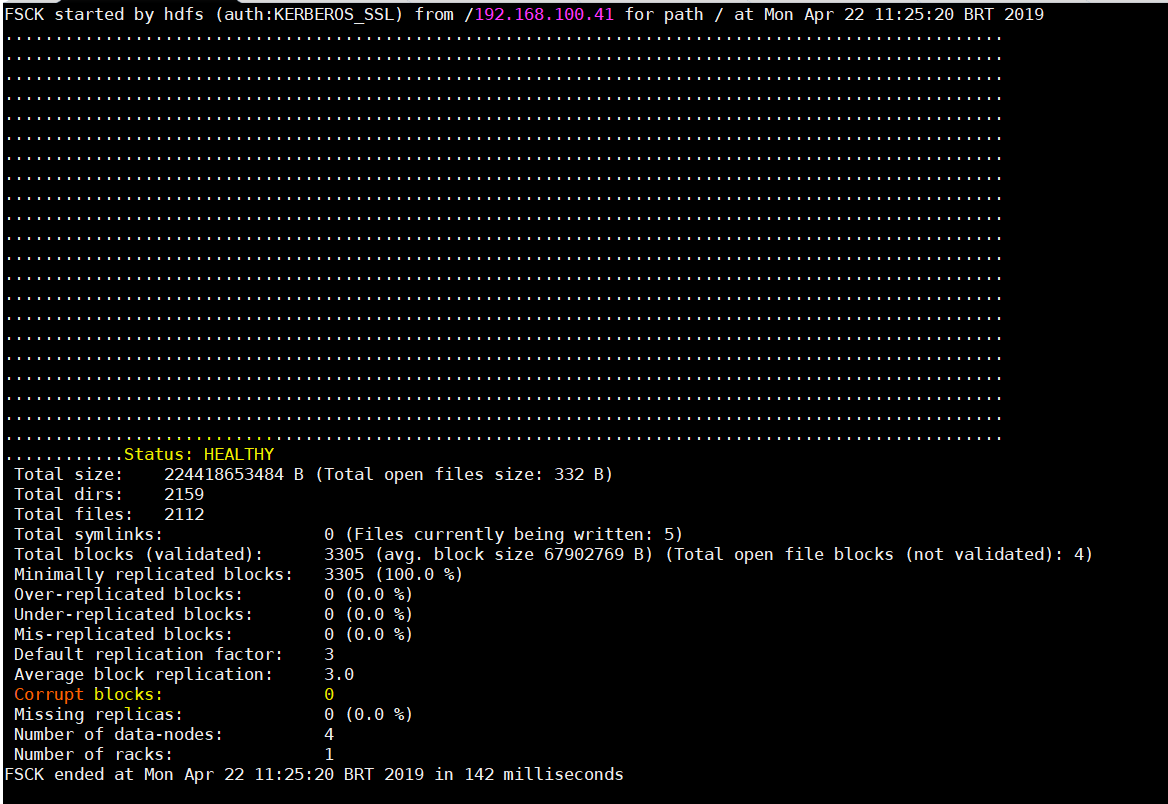
### **Execute o Security Inspector**

Administration > Security > Security Inspector



### **Verificar se o HDFS está “healthy”**

hdfs fsck / -includeSnapshots



hdfs dfsadmin -report

[root@node0 ~]# hdfs dfsadmin -report

Configured Capacity: 4346446135296 (3.95 TB)

Present Capacity: 4249039070180 (3.86 TB)

DFS Remaining: 3570478044132 (3.25 TB)

DFS Used: 678561026048 (631.96 GB)

DFS Used%: 15.97%

Under replicated blocks: 0

Blocks with corrupt replicas: 0

Missing blocks: 0

Missing blocks (with replication factor 1): 0

-------------------------------------------------

Live datanodes (4):

Name: 192.168.100.42:1004 (node1.lab.local)

Hostname: node1.lab.local

Rack: /default

Decommission Status : Normal

Configured Capacity: 865665232896 (806.21 GB)

DFS Used: 226174709760 (210.64 GB)

Non DFS Used: 0 (0 B)

DFS Remaining: 638953652556 (595.07 GB)

DFS Used%: 26.13%

DFS Remaining%: 73.81%

Configured Cache Capacity: 4294967296 (4 GB)

Cache Used: 0 (0 B)

Cache Remaining: 4294967296 (4 GB)

Cache Used%: 0.00%

Cache Remaining%: 100.00%

Xceivers: 11

Last contact: Mon Apr 22 11:27:34 BRT 2019

Name: 192.168.100.43:1004 (node2.lab.local)

Hostname: node2.lab.local

Rack: /default

Decommission Status : Normal

Configured Capacity: 865665232896 (806.21 GB)

DFS Used: 226182909952 (210.65 GB)

Non DFS Used: 0 (0 B)

DFS Remaining: 638945452364 (595.06 GB)

DFS Used%: 26.13%

DFS Remaining%: 73.81%

Configured Cache Capacity: 4294967296 (4 GB)

Cache Used: 0 (0 B)

Cache Remaining: 4294967296 (4 GB)

Cache Used%: 0.00%

Cache Remaining%: 100.00%

Xceivers: 12

Last contact: Mon Apr 22 11:27:33 BRT 2019

Name: 192.168.100.44:1004 (node3.lab.local)

Hostname: node3.lab.local

Rack: /default

Decommission Status : Normal

Configured Capacity: 865665232896 (806.21 GB)

DFS Used: 226134646784 (210.60 GB)

Non DFS Used: 0 (0 B)

DFS Remaining: 639127933177 (595.23 GB)

DFS Used%: 26.12%

DFS Remaining%: 73.83%

Configured Cache Capacity: 4294967296 (4 GB)

Cache Used: 0 (0 B)

Cache Remaining: 4294967296 (4 GB)

Cache Used%: 0.00%

Cache Remaining%: 100.00%

Xceivers: 10

Last contact: Mon Apr 22 11:27:33 BRT 2019

Name: 192.168.100.46:1004 (node5.lab.local)

Hostname: node5.lab.local

Rack: /default

Decommission Status : Normal

Configured Capacity: 1749450436608 (1.59 TB)

DFS Used: 68759552 (65.57 MB)

Non DFS Used: 95796453376 (89.22 GB)

DFS Remaining: 1653451006035 (1.50 TB)

DFS Used%: 0.00%

DFS Remaining%: 94.51%

Configured Cache Capacity: 4294967296 (4 GB)

Cache Used: 0 (0 B)

Cache Remaining: 4294967296 (4 GB)

Cache Used%: 0.00%

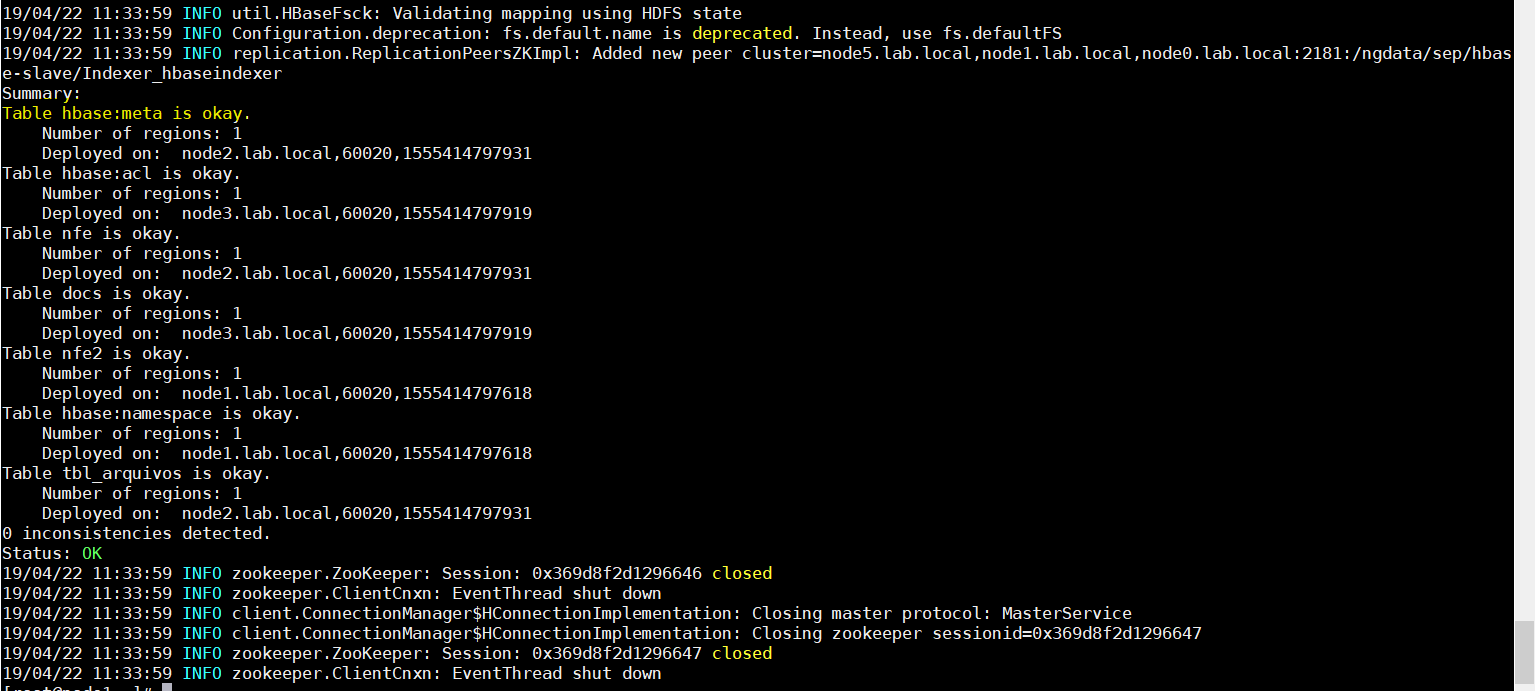
Cache Remaining%: 100.00%

Xceivers: 4

Last contact: Mon Apr 22 11:27:34 BRT 2019

### **Verificar se o HBase está “healthy”**

hbase hbck



### **Verificar se o Kudu está “healthy”**

kudu cluster ksck node0.lab.local

Master Summary

UUID | Address | Status

----------------------------------+-----------------+---------

473d7d5decb34593a7fe1a147b367822 | node0.lab.local | HEALTHY

Tablet Server Summary

UUID | Address | Status

----------------------------------+----------------------+---------

035d640e6cde4d19beaa4e61c55af522 | node3.lab.local:7050 | HEALTHY

59fdf16ec81041a5b82d6278434c0548 | node2.lab.local:7050 | HEALTHY

9de2d1b1f2eb4955b642942d80e776de | node1.lab.local:7050 | HEALTHY

Version Summary

Version | Servers

-----------------+-------------------------

1.7.0-cdh5.16.1 | all 4 server(s) checked

Summary by table

Name | RF | Status | Total Tablets | Healthy | Recovering | Under-replicated | Unavailable

------------------------------+----+---------+---------------+---------+------------+------------------+-------------

impala::bpe.emit | 3 | HEALTHY | 2 | 2 | 0 | 0 | 0

impala::bpe.ide | 3 | HEALTHY | 2 | 2 | 0 | 0 | 0

impala::docs.wc | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.arma | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.cide | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.cofins | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.combustiveis | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.dest | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.detexport | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.di | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.emit | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.encerrante | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.frete | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.icms | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.icmstot | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.icmsufdest | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.ide | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.ii | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.impostodevol | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.ipi | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.ipitrib | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.issqn | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.issqntot | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.lacres | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.medicamentos | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.modalidadefrete | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.pis | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.produto | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.rastro | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.recopi | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.refcte | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.refecf | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.refnf | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.refnfe | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.refnfp | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.rettransp | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.rettrib | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.total | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.transportador | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.veicprod | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.veictransp | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

impala::nifi.voltransp | 3 | HEALTHY | 1 | 1 | 0 | 0 | 0

| Total Count

----------------+-------------

Masters | 1

Tablet Servers | 3

Tables | 42

Tablets | 44

Replicas | 132

OK

## Backup

### **Realizando o backup dos bancos de dados dos serviços (Hue, Sentry, Hive, etc)**

Oozie

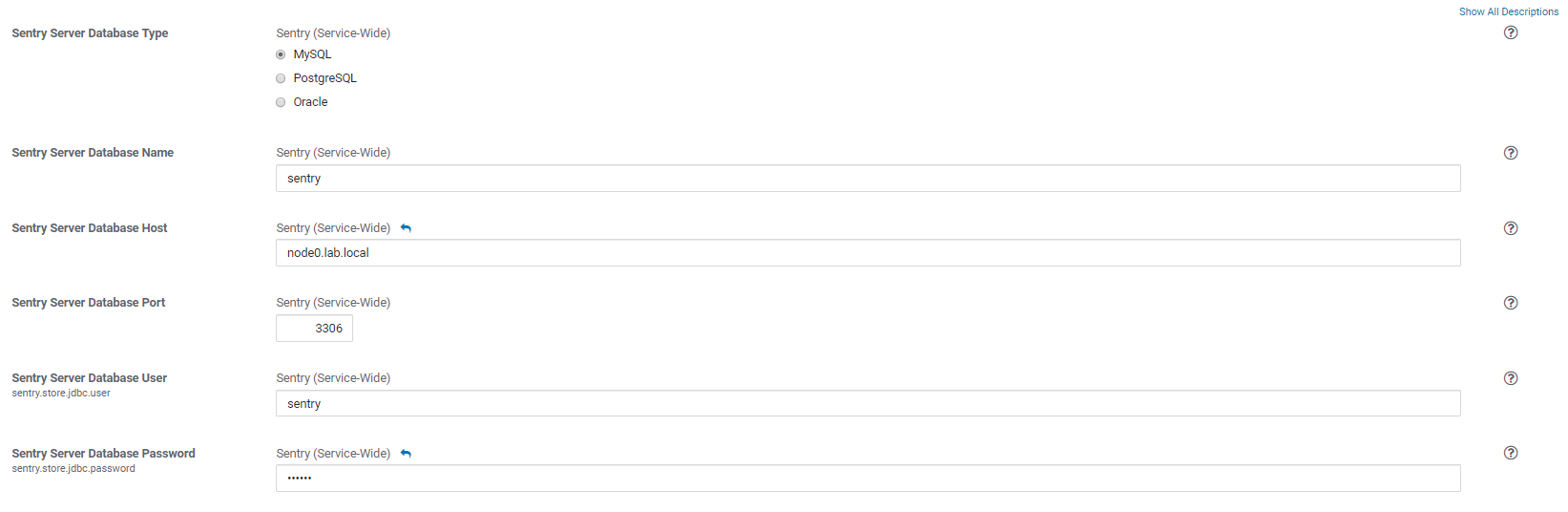


### **Hue**



### **Hive**

### **Sentry**

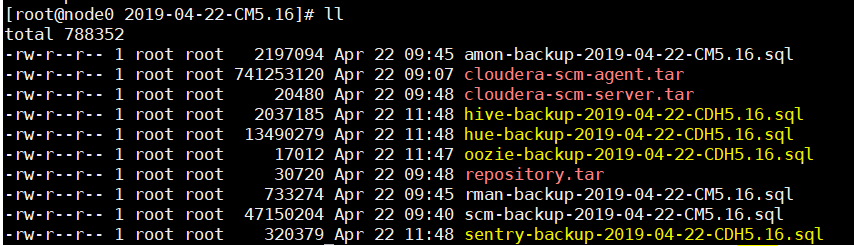


mysqldump --databases oozie --host=localhost -u oozie -p > oozie-backup-`date +%F`-CDH5.16.sql

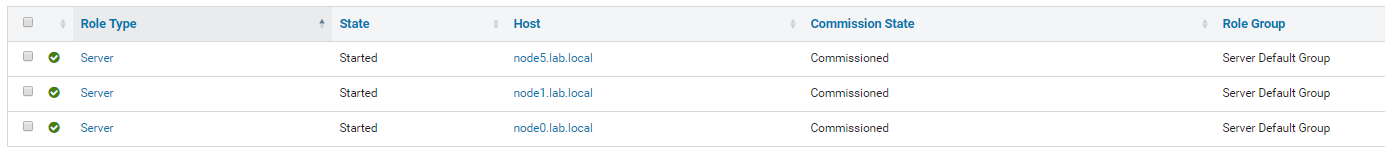
mysqldump --databases hue --host=localhost -u hue -p > hue-backup-`date +%F`-CDH5.16.sql

mysqldump --databases sentry --host=localhost -u sentry -p > sentry-backup-`date +%F`-CDH5.16.sql

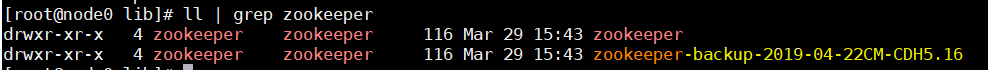
mysqldump --databases metastore --host=localhost -u hive -p > hive-backup-`date +%F`-CDH5.16.sql



### **Realizando o backup do Zookeeper**



cp -rp /var/lib/zookeeper/ /var/lib/zookeeper-backup-`date +%F`CM-CDH5.16



## Realizando o backup do HDFS

### **Namenode - executar o comando apenas no namenode ativo**

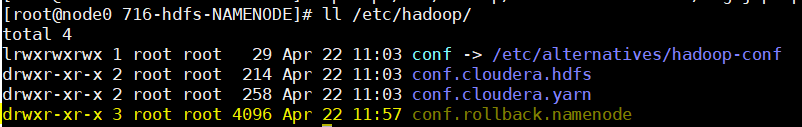
mkdir -p /etc/hadoop/conf.rollback.namenode

cd /var/run/cloudera-scm-agent/process/ && cd `ls -t1 | grep -e "-NAMENODE\$" | head -1`

cp -rp \* /etc/hadoop/conf.rollback.namenode/

rm -rf /etc/hadoop/conf.rollback.namenode/log4j.properties

cp -rp /etc/hadoop/conf.cloudera.hdfs/log4j.properties /etc/hadoop/conf.rollback.namenode/



### **Datanode - executar os comandos em todos os datanodes**

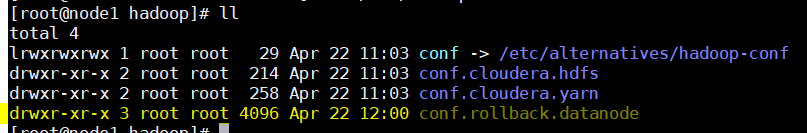
mkdir -p /etc/hadoop/conf.rollback.datanode/

cd /var/run/cloudera-scm-agent/process/ && cd `ls -t1 | grep -e "-DATANODE\$" | head -1`

cp -rp \* /etc/hadoop/conf.rollback.datanode/

rm -rf /etc/hadoop/conf.rollback.datanode/log4j.properties

cp -rp /etc/hadoop/conf.cloudera.hdfs/log4j.properties /etc/hadoop/conf.rollback.datanode/



### **Secondary NameNode - executar apenas no host do SNN**

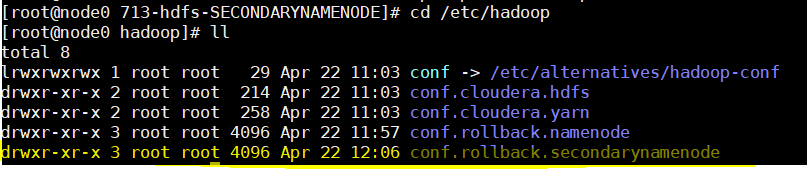
mkdir -p /etc/hadoop/conf.rollback.secondarynamenode/

cd /var/run/cloudera-scm-agent/process/ && cd `ls -t1 | grep -e "-SECONDARYNAMENODE\$" | head -1`

cp -rp \* /etc/hadoop/conf.rollback.secondarynamenode/

rm -rf /etc/hadoop/conf.rollback.secondarynamenode/log4j.properties

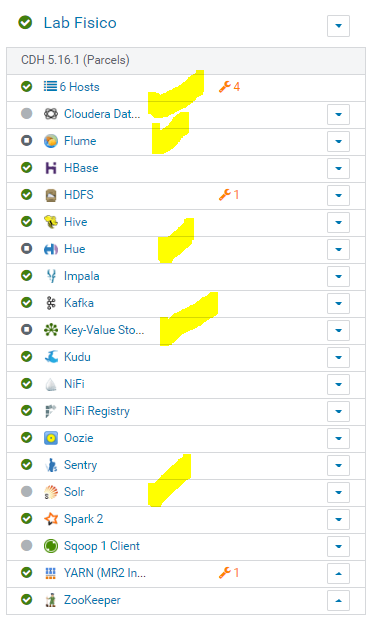
cp -rp /etc/hadoop/conf.cloudera.hdfs/log4j.properties /etc/hadoop/conf.rollback.secondarynamenode/



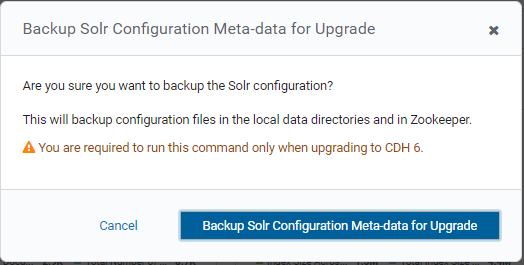
## Realizando o backup do Solr

Pare o serviço do Solr via CM, antes pare os serviços dependentes:

Hue, Flume, Key-Value Store Indexer e CDSW



Solr > Action > Backup Solr Configuration Metadata for Upgrade



Caso a pasta não exista, crie e depois de permissão ao usuário Solr para poder escrever na pasta

[root@node0 hadoop]# hdfs dfs -mkdir -p /user/solr/upgrade\_backup

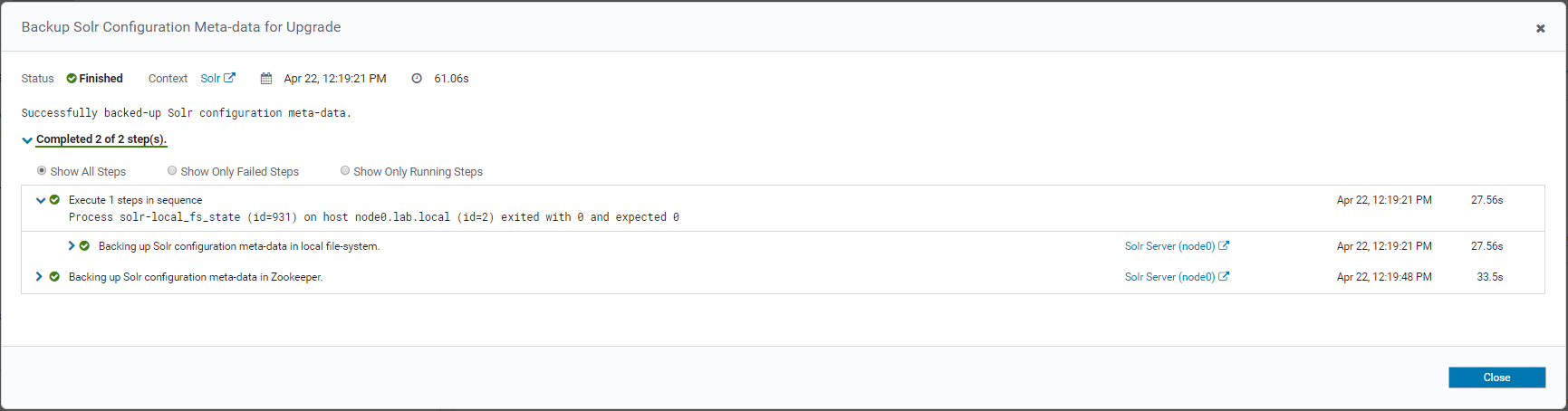
[root@node0 hadoop]# hdfs dfs -chown -R solr:solr /user/solr/

[root@node0 hadoop]# hdfs dfs -ls /user/solr/

Found 1 items

drwxr-xr-x+ - solr solr 0 2019-04-22 12:17 /user/solr/upgrade\_backup

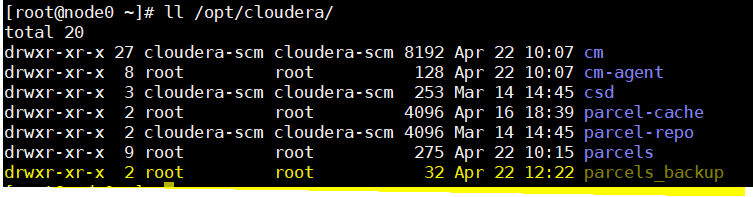
Caso não ocorra nenhum erro.



## Realizando o backup do Hue

mkdir -p /opt/cloudera/parcels\_backup

cp -rp /opt/cloudera/parcels/CDH/lib/hue/app.reg /opt/cloudera/parcels\_backup/app.reg-CM-CDH5.16

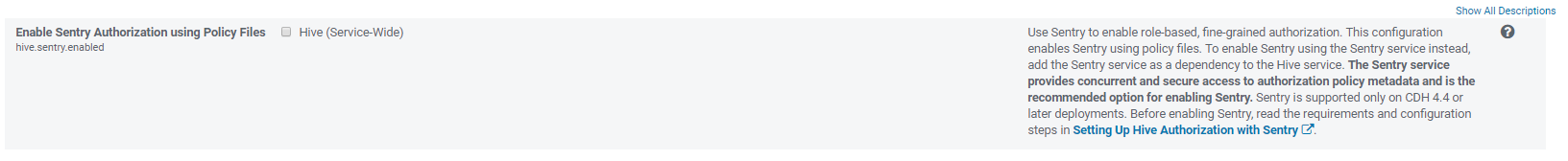


## CDH 6 - Pré-migração

### **Migrando os arquivos de Sentry Policy para o Sentry(serviço)**

Não é o caso do nosso cluster, visto que sentry policy files eram utilizados em versões mais antigas do CDH.

Segue exemplo abaixo ( o mesmo pode ser verificado no Impala e Solr)



### **Migrando as configurações do Solr(Cloudera Search) antes de atualizar o CDH para versão 6**

Antes de atualizar:

- Garantir que o CM está na versão 6;   
- Se estiver usando Apache Sentry policy files, migra-los para o Sentry service;   
- Para de utilizar o Solr durante a migração;   
- Para o serviço do Lily HBase (Key-Value Store Index);  
 - Não criar, remover ou modificar qualquer coleção durante a migração.

No nosso caso, apenas o item abaixo foi necessário realizar:



### **Rodando o script de migração do Solr**

Confirme que a variável SOLR\_ZK\_ENSEMBLE existe

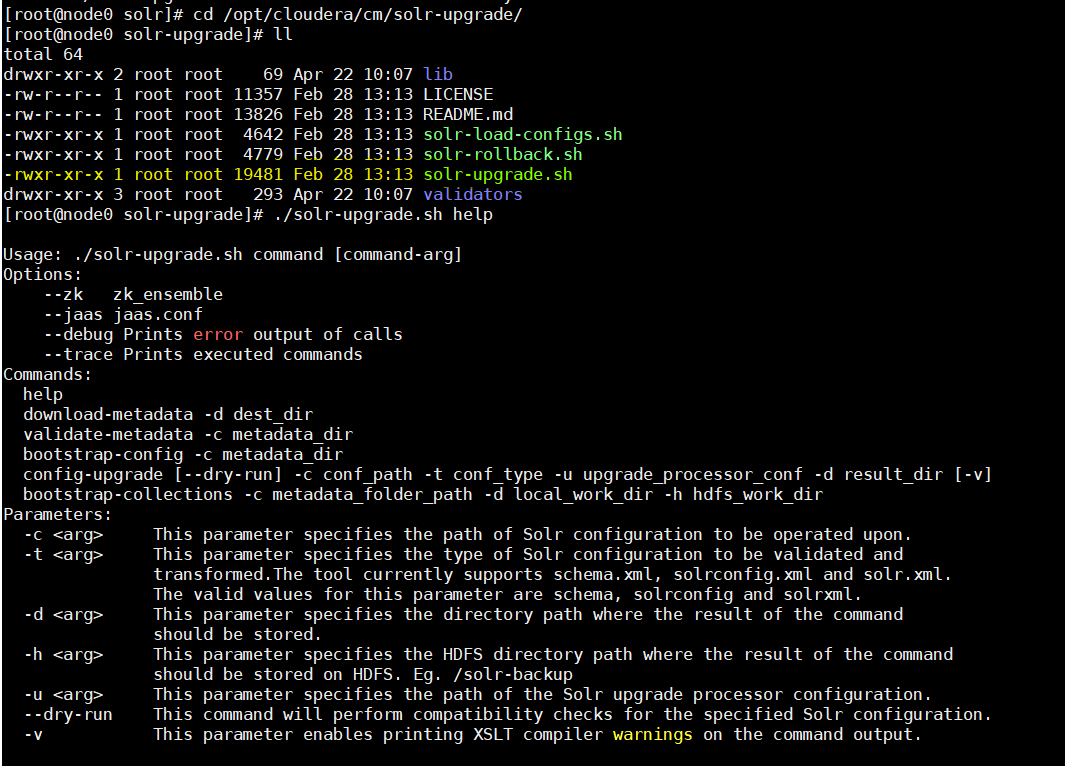
[root@node0 ~]# cat /etc/solr/conf/solr-env.sh

export SOLR\_ZK\_ENSEMBLE=node5.lab.local:2181,node1.lab.local:2181,node0.lab.local:2181/solr

[root@node0 ~]# export CDH\_SOLR\_HOME=/opt/cloudera/parcels/CDH/lib/solr

[root@node0 ~]# echo $CDH\_SOLR\_HOME

/opt/cloudera/parcels/CDH/lib/solr



## Migrando Solr configuration para compatibilidade com o CDH6

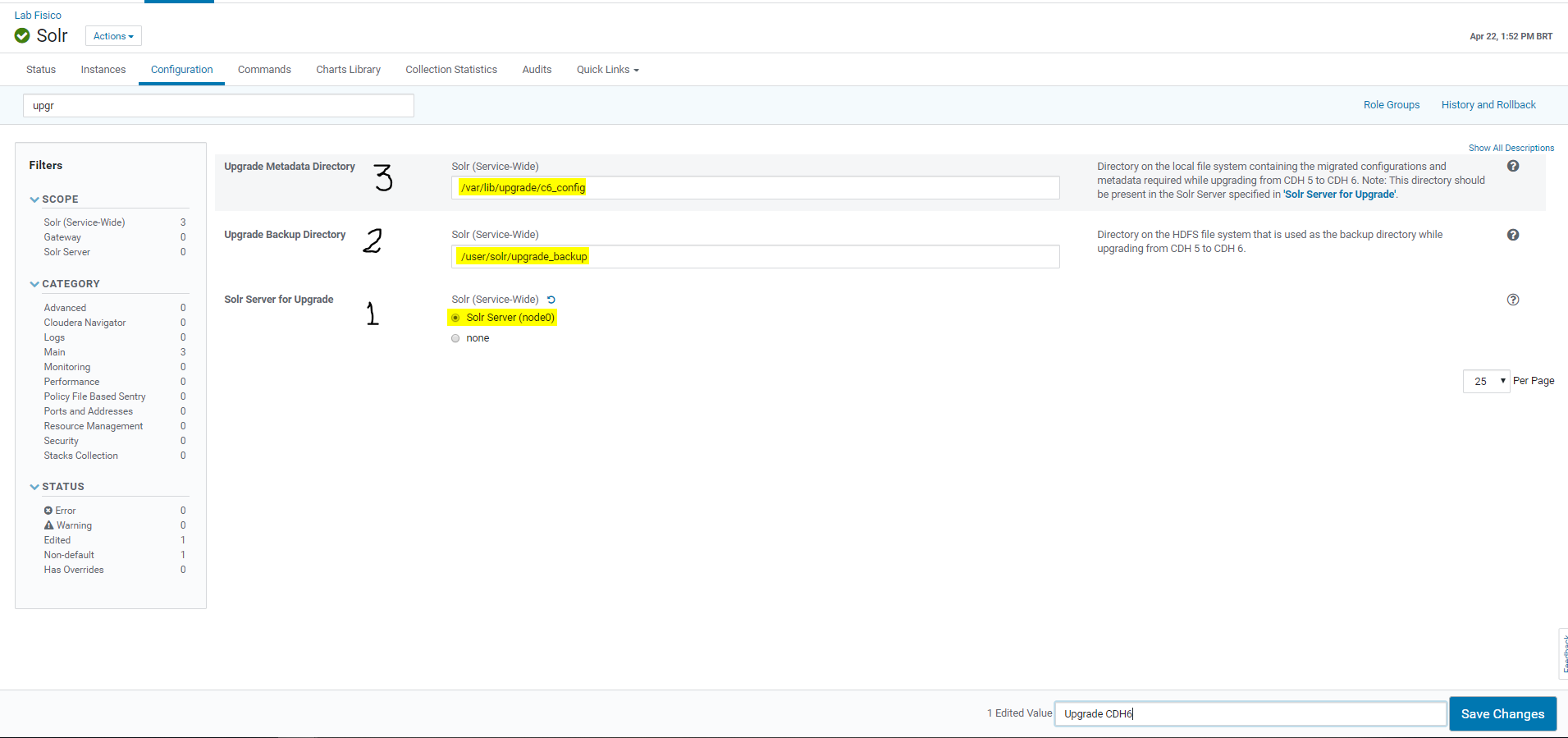
Solr > Configuration

Digite **upgrade** no campo ‘**search’**

Selecione um Solr Server host no passo 1

Define uma pasta para **Upgrade Backup Directory** (pasta no HDFS) - passo 2

Define uma pasta para **Upgrade Metadta Directory** (pasta no file system) - passo 3



Clique em Save Changes.

### **Upgrade Backup Directory**

[root@node0 c6\_config]# ls /var/lib/ | grep upgrade

upgrade

[root@node0 c6\_config]# ls /var/lib/upgrade/ | grep c6

c6\_config

### **Upgrade Metadta Directory**

[root@node0 ~]# hdfs dfs -ls /user/solr

Found 1 items

drwxr-xr-x+ - solr solr 0 2019-04-22 12:20 /user/solr/upgrade\_backup

### **Faça o Backup dos dados de configuração do Solr**

Este passo já foi realizado [aqui](#Backup_Solr)

### **Migração dos arquivos de configuração**

Defina a variável de ambiente - CDH\_SOLR\_HOME

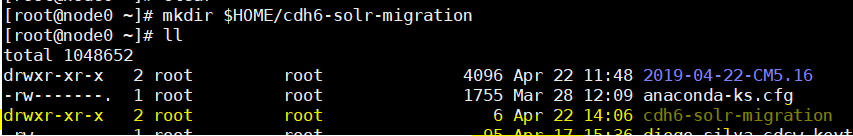
export CDH\_SOLR\_HOME=/opt/cloudera/parcels/CDH/lib/solr

Definir a variável de ambiente - JAVA\_HOME

[root@node0 ~]# echo $JAVA\_HOME

/usr/java/jdk1.8.0\_181-amd64/

Crie um diretório para a migração



Inicie o ticket do Kerberos para o Solr:

[root@node0 ~]# kinit -kt /var/run/cloudera-scm-agent/process/935-solr-SOLR\_SERVER/solr.keytab solr/node0.lab.local@KDC.LAB

[root@node0 ~]# klist

Ticket cache: FILE:/tmp/krb5cc\_0

Default principal: solr/node0.lab.local@KDC.LAB

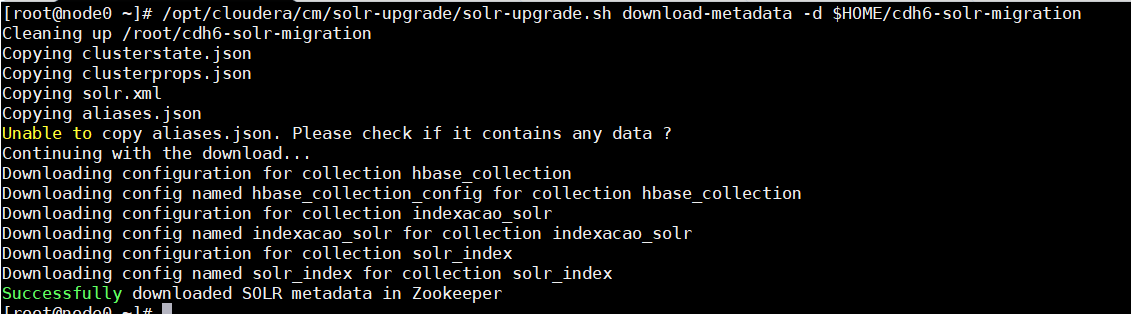
Valid starting Expires Service principal

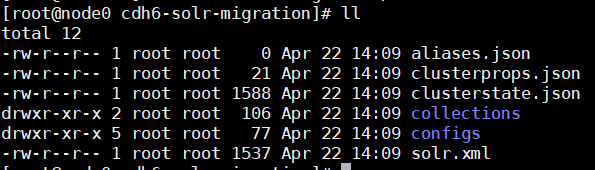
04/22/2019 14:07:20 04/23/2019 14:07:20 krbtgt/KDC.LAB@KDC.LAB

renew until 04/27/2019 14:07:20

### **Faça o download das configurações do Solr (CDH5) do Zookeeper**

/opt/cloudera/cm/solr-upgrade/solr-upgrade.sh download-metadata -d $HOME/cdh6-solr-migration



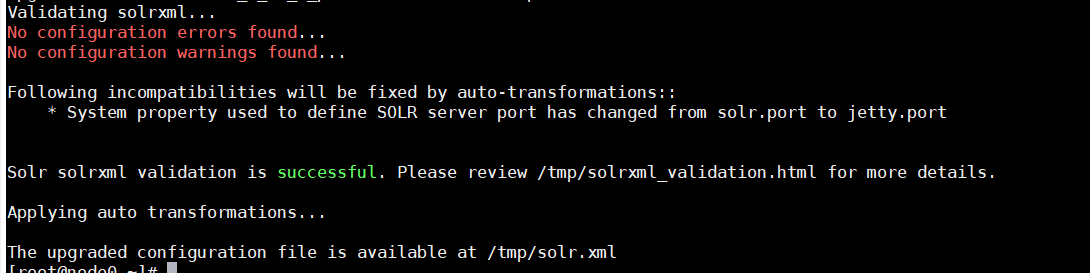


### **Faça uma cópia dos arquivos de configuração migrados**

cp -r $HOME/cdh6-solr-migration $HOME/cdh6-migrated-solr-config

### **Migre o arquivo solr.xml**

/opt/cloudera/cm/solr-upgrade/solr-upgrade.sh config-upgrade -t solrxml -c $HOME/cdh6-solr-migration/solr.xml -u /opt/cloudera/cm/solr-upgrade/validators/solr\_4\_to\_7\_processors.xml -d /tmp



### **Copie o arquivo solr.xml para a pasta de migração**

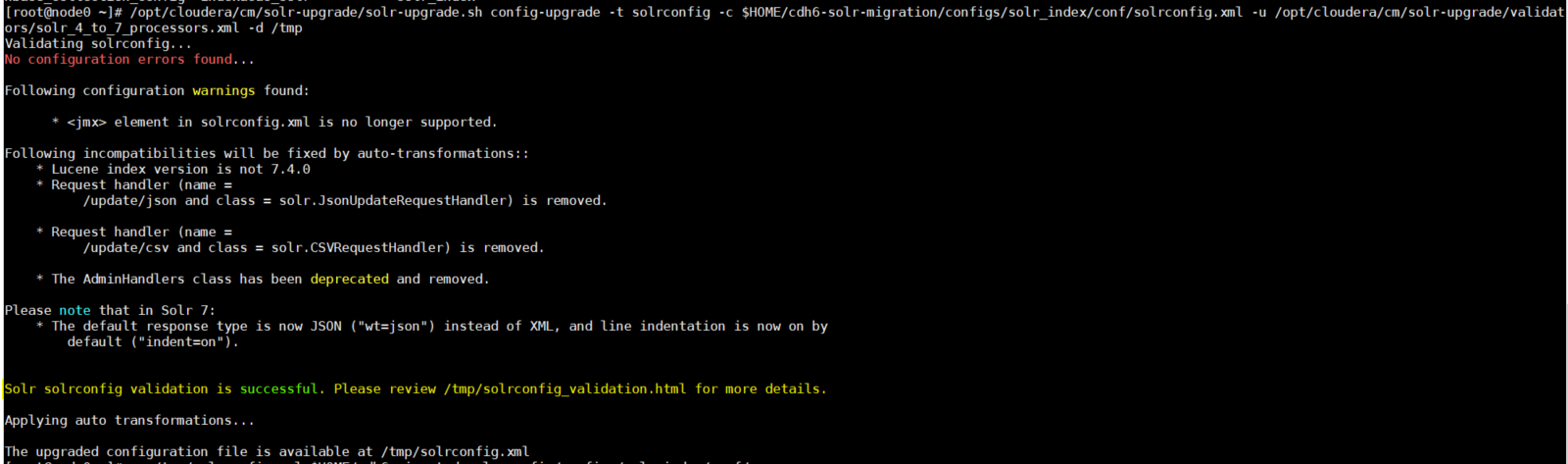
cp /tmp/solr.xml $HOME/cdh6-migrated-solr-config

Para cada uma das coleções, migre o arquivo solr.xml para a pasta que contém os arquivos de configuração já ‘migrados’

### **Arquivo solr.xml**

/opt/cloudera/cm/solr-upgrade/solr-upgrade.sh config-upgrade -t solrconfig -c $HOME/cdh6-solr-migration/configs/**NOME\_DA\_COLEÇÃO**/conf/solrconfig.xml -u /opt/cloudera/cm/solr-upgrade/validators/solr\_4\_to\_7\_processors.xml -d /tmp

O script irá identificar as incompatibilidades, abra o arquivo usando o editor de sua preferência e remova, em seguida rode o script novamente até que não haja mais incompatibilidades, como no exemplo abaixo:



Copie o arquivo solrconfig.xml gerado na pasta temporária para a pasta da coleção correspondente

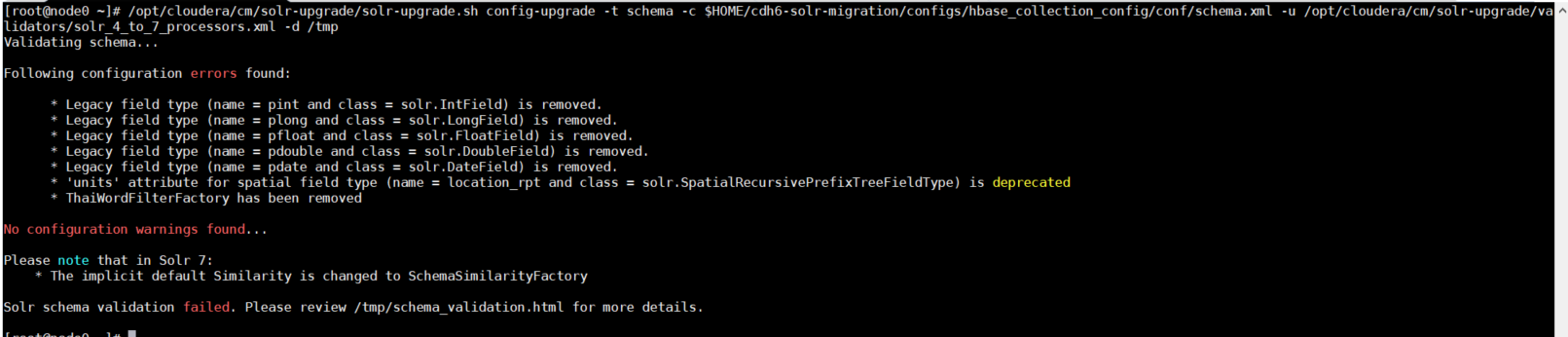
cp /tmp/solrconfig.xml $HOME/cdh6-migrated-solr-config/configs/**NOME\_DA\_COLEÇÃO**/conf/

### **Arquivo schema.xml**

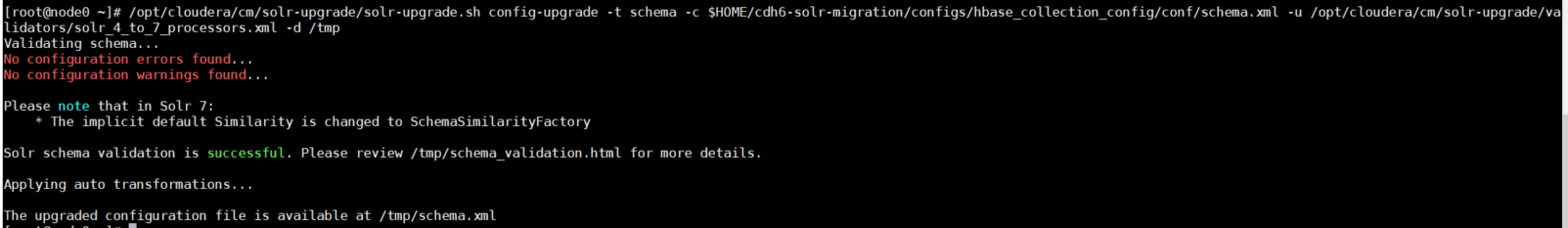
/opt/cloudera/cm/solr-upgrade/solr-upgrade.sh config-upgrade -t schema -c $HOME/cdh6-solr-migration/configs/**NOME\_DA\_COLEÇÃO**/conf/schema.xml -u /opt/cloudera/cm/solr-upgrade/validators/solr\_4\_to\_7\_processors.xml -d /tmp

O script irá identificar as incompatibilidades, abra o arquivo usando o editor de sua preferência e remova, em seguida rode o script novamente até que não haja mais incompatibilidades, como no exemplo abaixo:

**Com Incompatibilidades**



**Sem Incompatibilidades**



### **Copie o arquivo schema.xml gerado na pasta temporária para a pasta da coleção correspondente**

cp /tmp/schema.xml $HOME/cdh6-migrated-solr-config/configs/**NOME\_DA\_COLEÇÃO**/conf/

### **Faça a validação dos arquivos de configuração migrados**

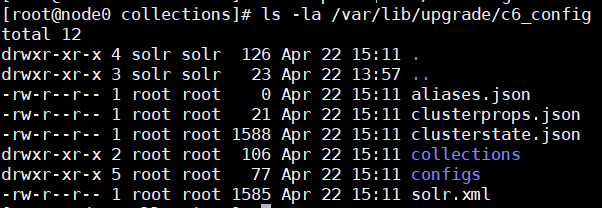
/opt/cloudera/cm/solr-upgrade/solr-upgrade.sh validate-metadata -c $HOME/cdh6-migrated-solr-config

Após a execução do script, verifique se a validação teve sucesso



### **Copie os arquivos de configuração já “migrados” para a pasta Upgrade Metadata Directory**

sudo cp -r $HOME/cdh6-migrated-solr-config/\* /var/lib/upgrade/c6\_config



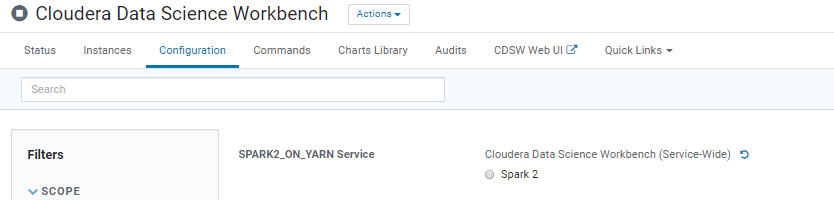
### **Remova o Apache Spark**

Pare o CDSW caso esteja instalado no cluster

  
Pare o Spark 2

  
Remova o Spark 2

Primeiro remova a dependência do CDSW



Spark 2 > Delete

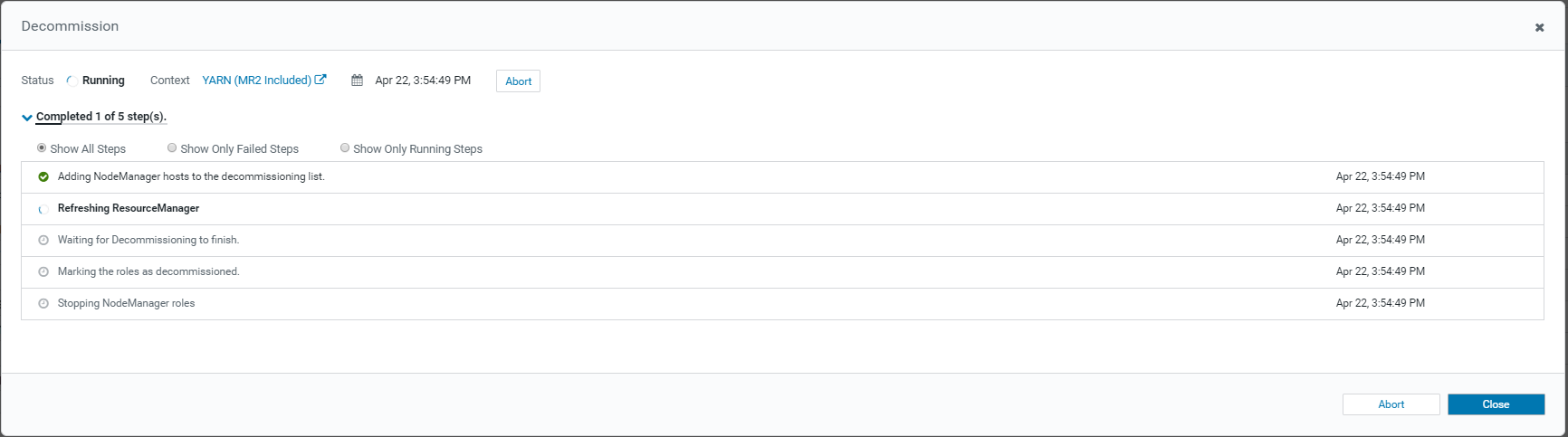
## Atualizando o Cluster

**Faça o backup do Cloudera Manager**

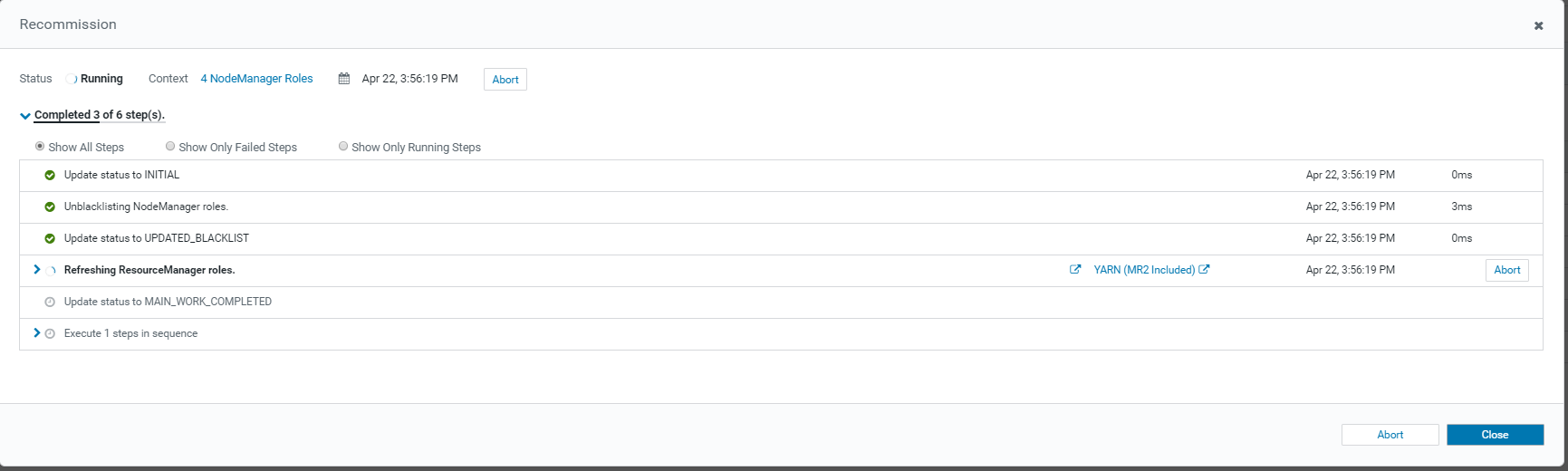
Siga os passos no manual da Cloudera<https://www.cloudera.com/documentation/enterprise/upgrade/topics/ug_cm_upgrade_backup.html>

**YARN**

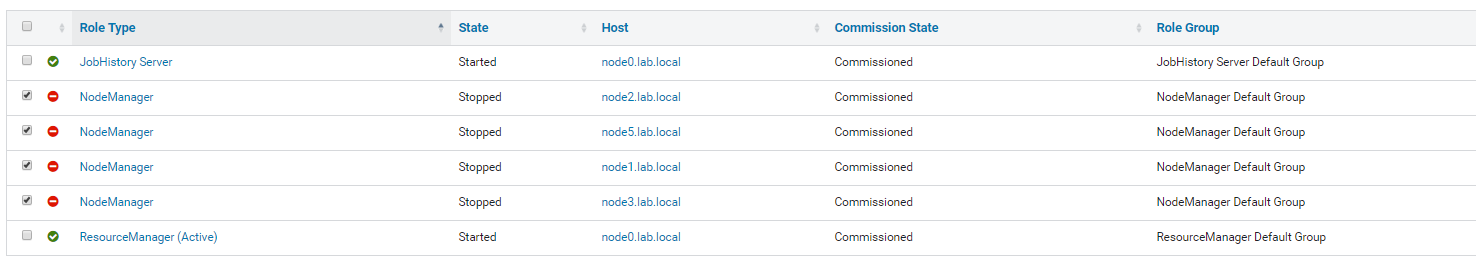
Decomissione os NodeManagers



Recomissione os NodeManagers



Mantenha os NodeManages ‘parados’



Kafka

Faça login no Cloudera Manager

Acesse o serviço do Kafka

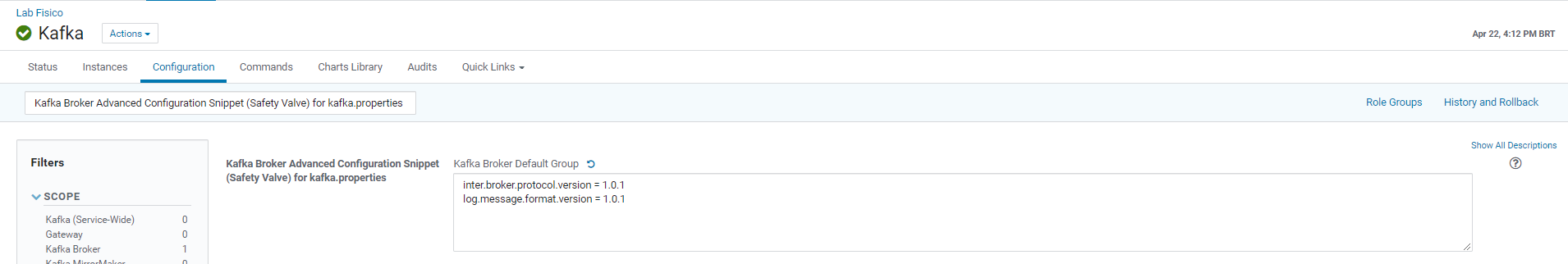
Clique em Configuration

Digite Kafka Broker Advanced Configuration Snippet (Safety Valve) for kafka.properties no campo de busca

Insira a seguinte informação no campo:

inter.broker.protocol.version = 1.0.1

log.message.format.version = 1.0.1



Hue

Verifique se alguma das tabelas abaixo possui mais de 100 mil linhas, caso contrário, este passo pode ser pulado.

select 'desktop\_document' as <table\_name>, count(\*) from desktop\_document

union

select 'desktop\_document2' as <table\_name>, count(\*) from desktop\_document2

union

select 'beeswax\_session' as <table\_name>, count(\*) from beeswax\_session

union

select 'beeswax\_savedquery' as <table\_name>, count(\*) from beeswax\_savedquery

union

select 'beeswax\_queryhistory' as <table\_name>, count(\*) from beeswax\_queryhistory

union

select 'oozie\_job' as <table\_name>, count(\*) from oozie\_job

order by 1;

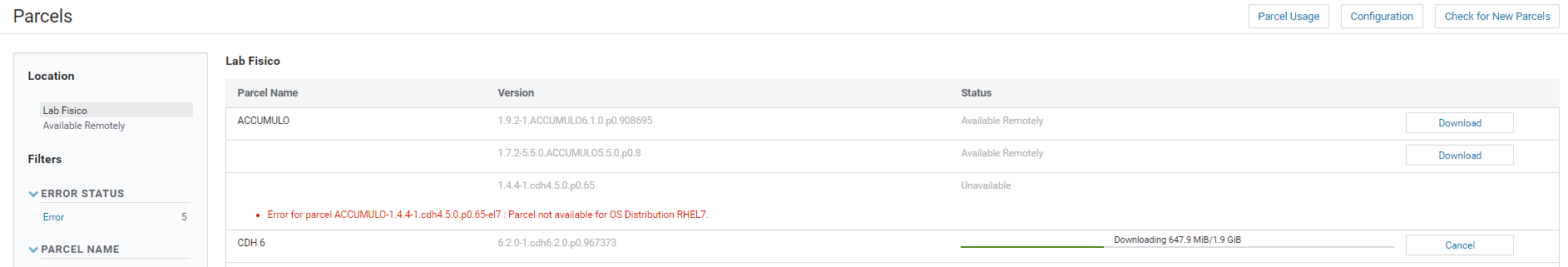
Procedimento completo encontra-se aqui:   
<https://www.cloudera.com/documentation/enterprise/upgrade/topics/ug_cdh_upgrade.html#id_zr5_ptp_sfb>

## Faça o download e distribua os parcels

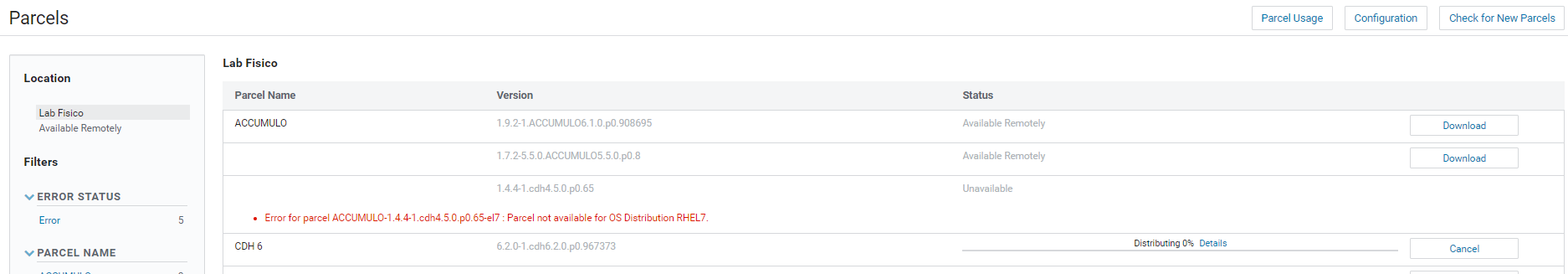
Login no Cloudera Manager

Hosts > Parcels

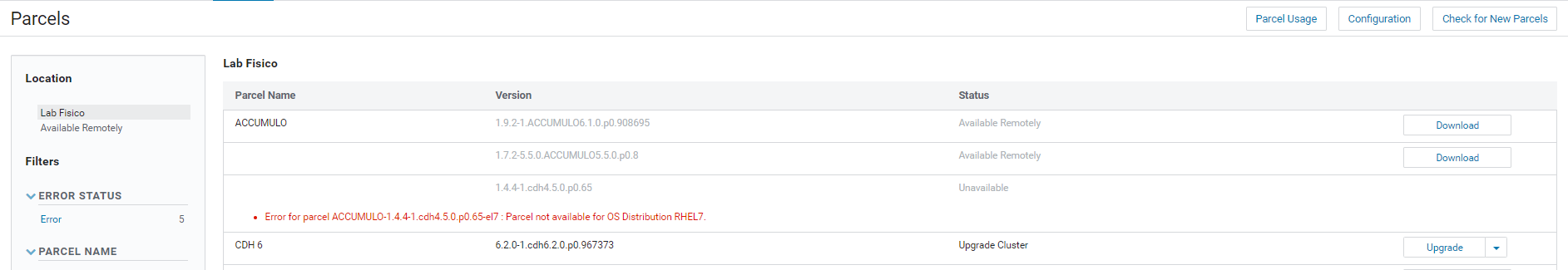
Clique em **Download**



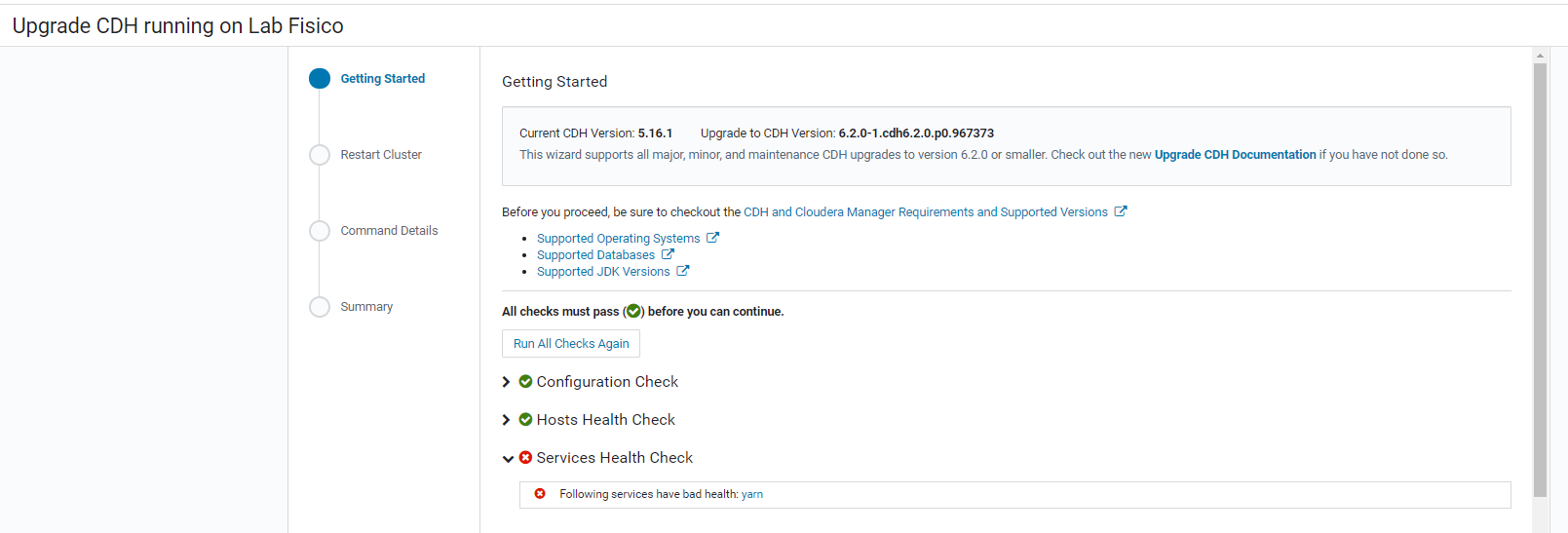
Assim que o Download terminar, clique em **Distribute**



Agora clique no botão **Upgrade**



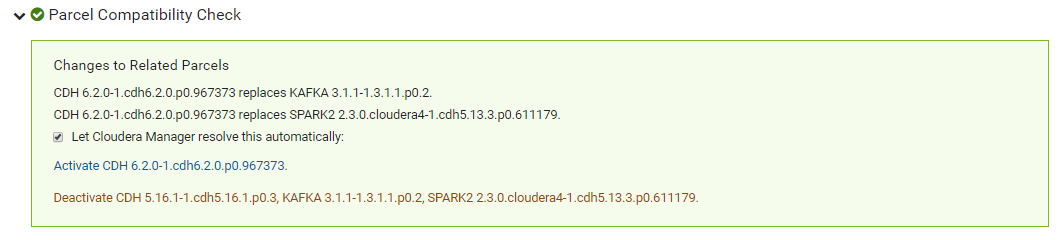
### **CDH Upgrade Wizard**



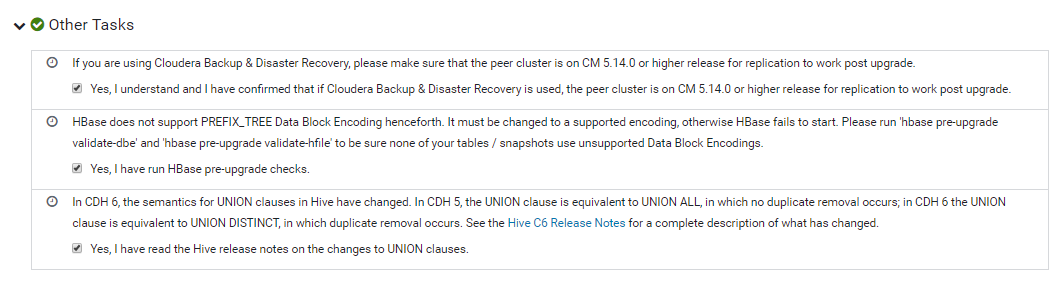
Corrija os erros apontados pelo Cloudera e em seguida clique no botão Run All Checks Again

Check de compatibilidade de parcel

Clique em **Let Cloudera Manager resolve this automatically**



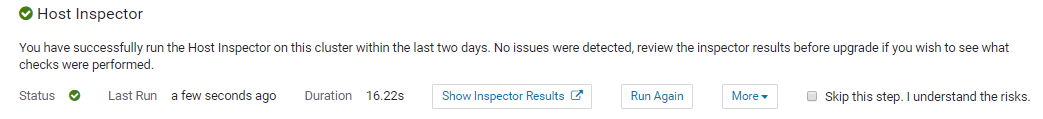
Marque as 3 opções abaixo para que seja possível prosseguir com o Upgrade do CDH



### **Inspector Checks**

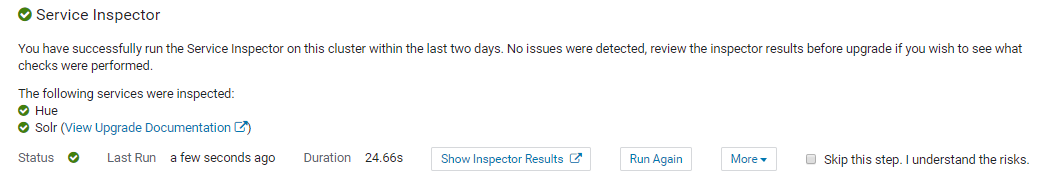
**Host Inspector**

Clique no botão Run Host Inspector

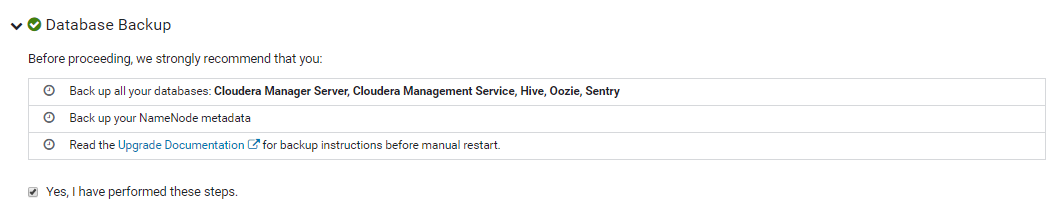


### **Service Inspector**

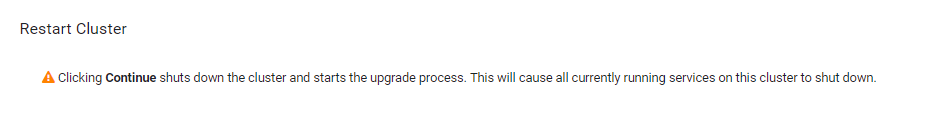
Clique no botão Run Service Inspector



### **Database Backup**

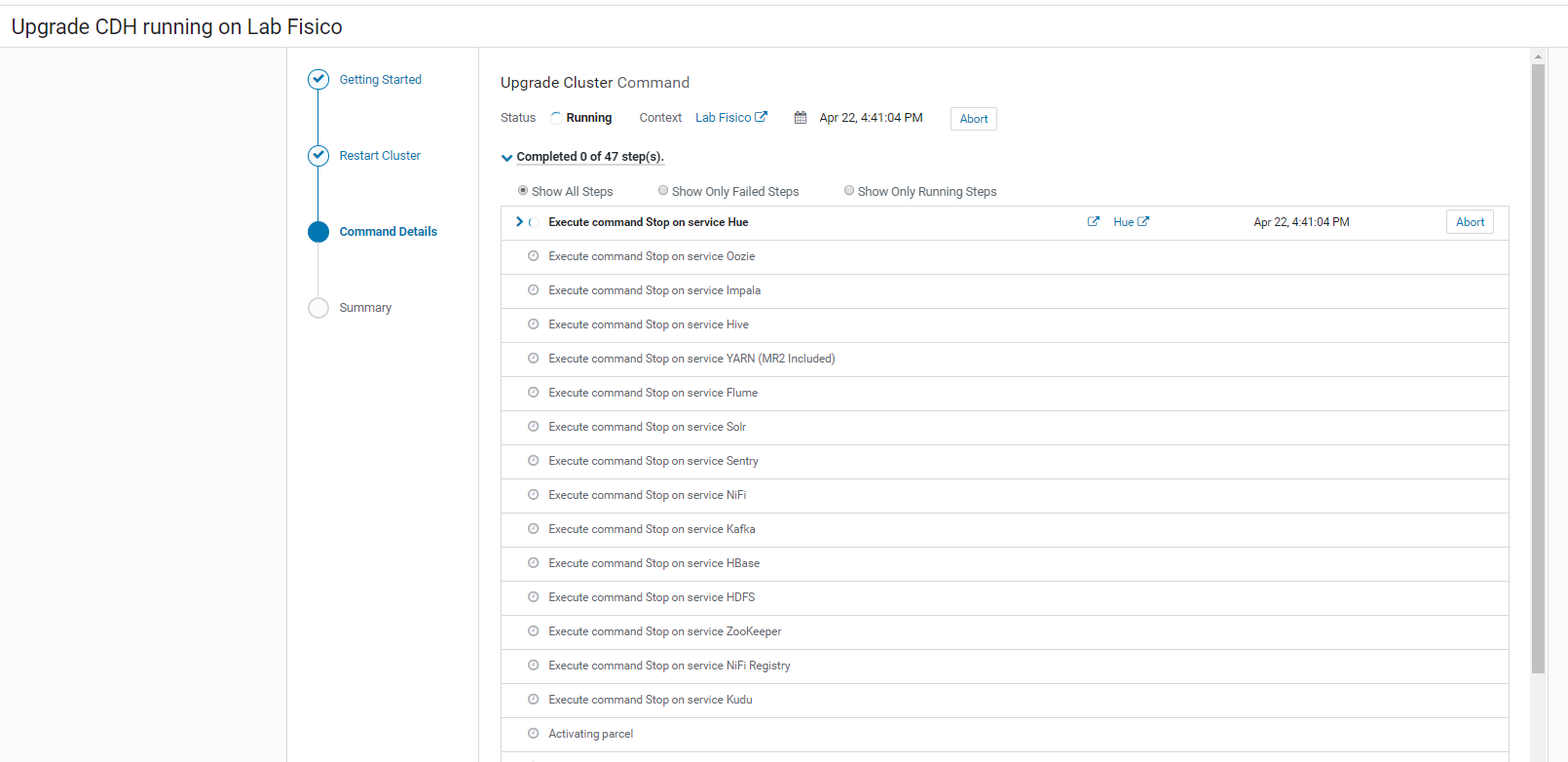


Clique em Continue

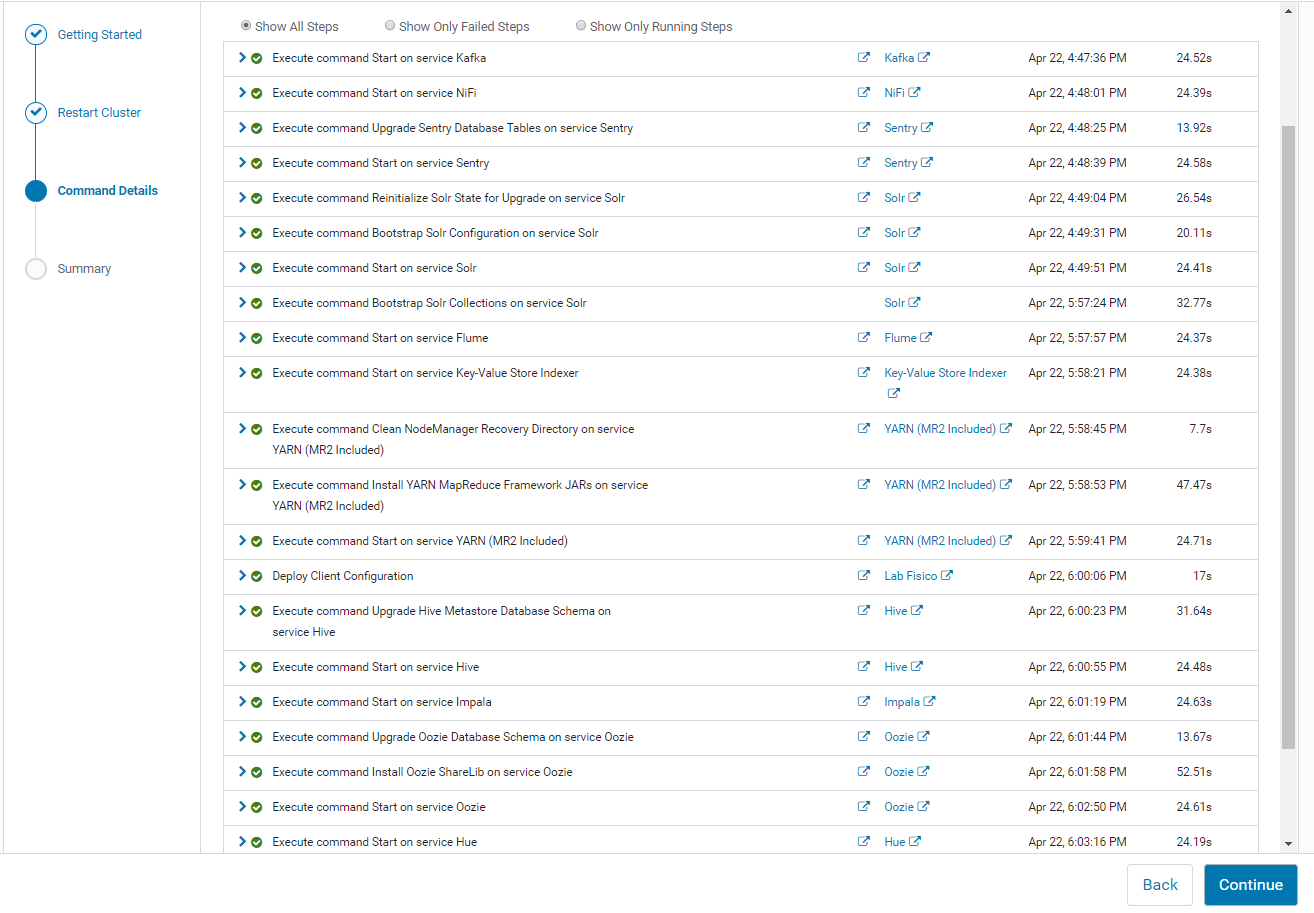


Clique em Continue para que o cluster seja reiniciado, assim como todos os serviços.

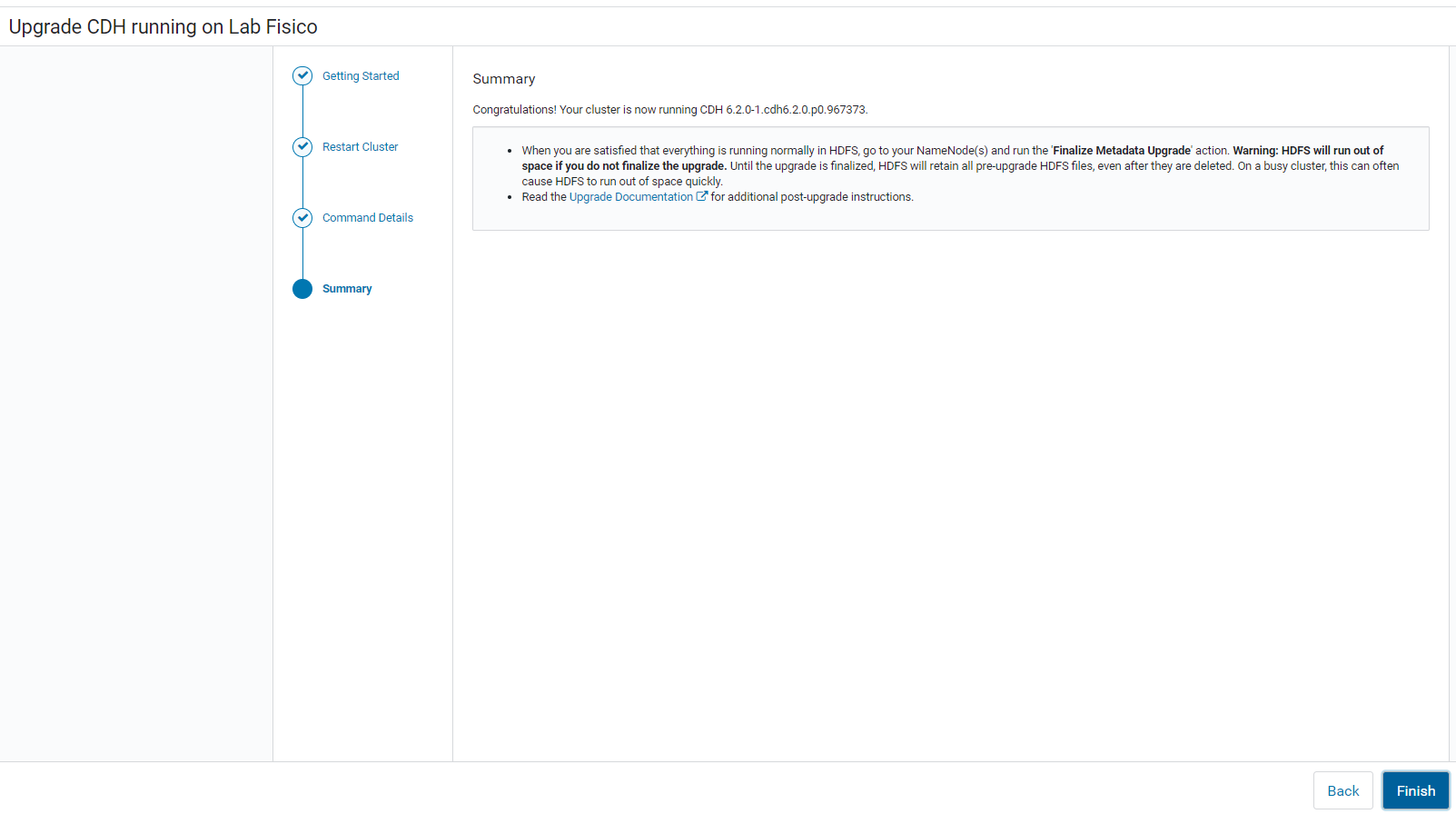
Agora basta aguardar que todos os passos sejam executados



Uma vez que todos os passos tenham sido concluídos com sucesso, clique em Continue



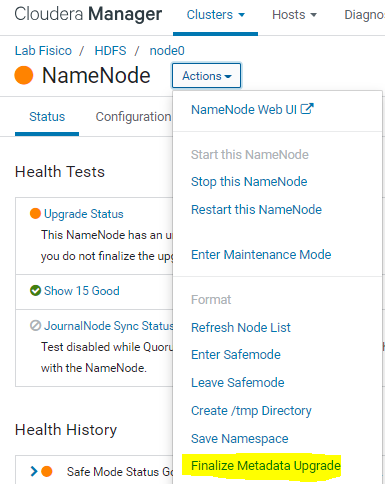
**Parabéns, seu cluster agora está rodando a versão 6.2 do CDH**



Clique em Finish.

## Finalizando o Upgrade do HDFS

HDFS > Namenode > Actions



Successfully finalized upgrade; The HDFS metadata is now permanently upgraded.

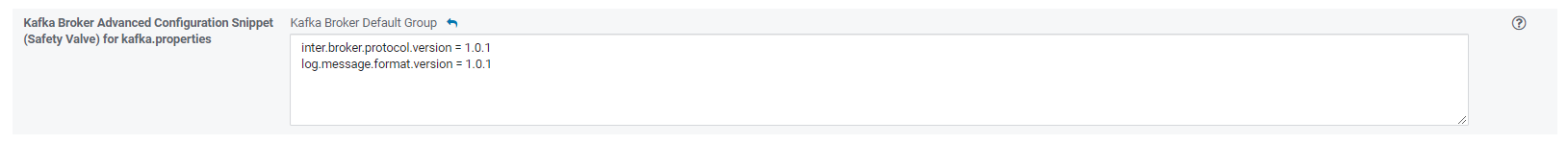


## Pós-upgrade steps

### **Kafka**

Remova as linhas inseridas na propriedade **Kafka Broker Advanced Configuration Snippet (Safety Valve) for kafka.properties**

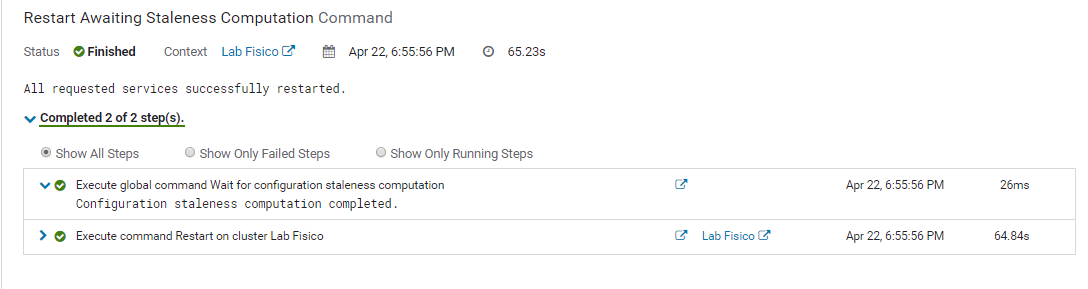
De



Para



### **Reinicie o cluster.**



## Troubleshooting

### **Solr**

Re-initialization of hbase\_collection FAILED. Run the following for details: solrctl collection --request-status restore-**hbase\_collection**-

Re-initialization of indexacao\_solr FAILED. Run the following for details: solrctl collection --request-status restore-**indexacao\_solr**-

Erros durante a conversão dos config files do Solr.

Solução

Remove as pastas de config das collections que estavam dando problema, na pasta

**/var/lib/upgrade/c6\_config/configs**

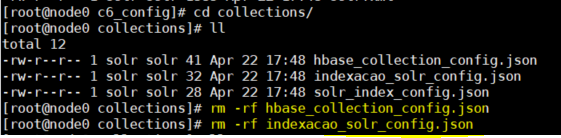
rm -rf



Remove os arquivos .json das collections que estavam dando problema, na pasta

**/var/lib/upgrade/c6\_config/collections**

rm -rf

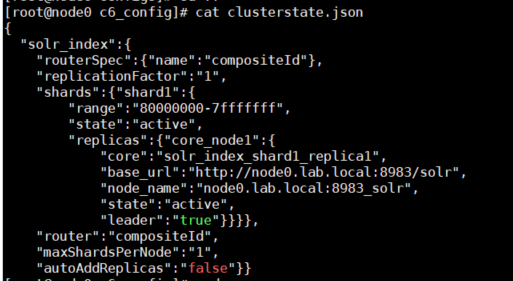


Alterar o arquivo clusterstate.json na pasta /var/lib/upgrade/c6\_config

De:

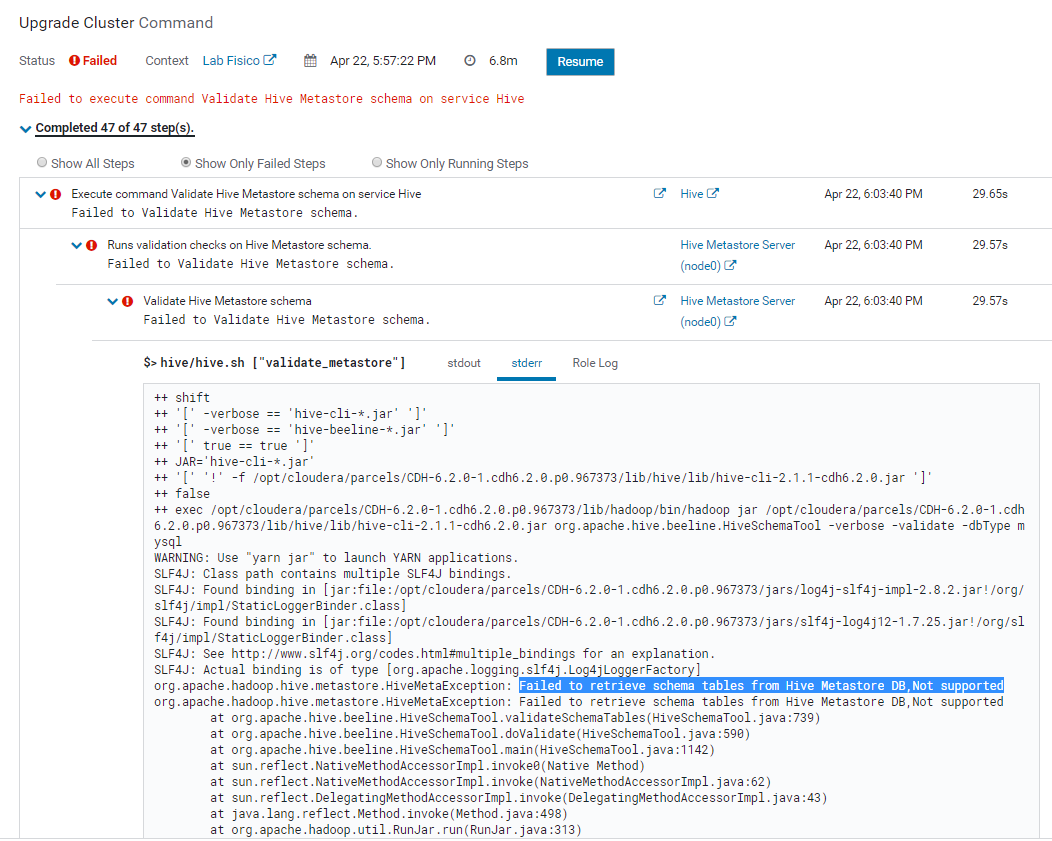


Para:



Clicar em Resume

### **Hive**



### **Faça o download do conector MYSQL**

wget <https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-java-5.1.46.tar.gz>

**Descompacte o conector**

tar zxvf mysql-connector-java-5.1.46.tar.gz

**Entre na pasta mysql-connector-java-5.1.46**

cd mysql-connector-java-5.1.46

**Copie o jar para a pasta**

cp mysql-connector-java-5.1.46-bin.jar /usr/share/java/mysql-connector-java.jar

**Faça o restart do MySQL**

systemctl restart mysql

Clique em Resume

### **Oozie**

**Ao tentar reiniciar o Oozie, o seguinte erro aparece:**

Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with existing applications not using SSL the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.

**Faça o download do conector MYSQL**

wget <https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-java-5.1.46.tar.gz>

**Descompacte o conector**

tar zxvf mysql-connector-java-5.1.46.tar.gz

**Entre na pasta mysql-connector-java-5.1.46**

cd mysql-connector-java-5.1.46

**Copie o jar para a pasta**

cp mysql-connector-java-5.1.46-bin.jar /var/lib/oozie/mysql-connector-java.jar

**Faça o restart do MySQL**

systemctl restart mysql