Cloudera Data Engineering 1.5.5

Using spark-submit drop-in migration tool for migrating Spark workloads to Cloudera Data Engineering

Date published: 2020-07-30
Date modified: 2025-11-08

CLOUDERA

https://docs.cloudera.com/

Legal Notice

Contents

Using spark-submit drop-in migration tool for migrating Spark workloads to Cloudera Data Engineering

Cloudera Data Engineering provides a command line tool cde-env to help migrate your Cloudera Spark workloads running on Cloudera Base on premises (spark-on-YARN) and Cloudera Data Hub to Cloudera Data Engineering without having to completely rewrite your existing spark-submit command-lines.

Supported platforms

You can use the migration tool in the following platforms:

- Linux
- MacOS
- Windows (Docker only)

You can use the migration tool either by installing it on a gateway host or running it as a docker container.

Using the cde-env tool

You can run the spark-submit command after installing the cde-env tool. You can use the cde-env tool even without administrator privileges.

You can activate the specific profile you want to use by running the following command:

```
$ cde-env.sh activate -p <PROFILE-NAME>
```

For example, to alternatively run the the same spark-submit command either against YARN or one of the Cloudera Data Engineering virtual cluster, you can activate the relevant profile:

1. Run spark jobs on YARN by activating the yarn profile. yarn is a reserved profile name here.

   ```
   $ cde-env.sh activate -p YARN
   ```

2. Run spark jobs on a Cloudera Data Engineering virtual cluster configured by CDE CLI profile named vc-1:

   ```
   $ cde-env.sh activate -p VC-1
   ```

Switching profiles back and forth this way lets you run the same spark-submit command either against YARN or one of the Cloudera Data Engineering virtual clusters.

Run sample spark-submit command

After you activate the profile using the cdp-env tool, you can run your spark-submit commands on Cloudera Data Engineering without completely rewriting your existing spark-on-yarn command lines.

- Sample spark-submit commands you can run on the Cloudera Data Engineering workloads.

  ```
  $ spark-submit \
  --name pt_rpt_streams \
  --master=yarn --deploy-mode=cluster \
  --driver-memory 4G \
  --executor-memory 4G --executor-cores 3 \
  --num-executors 4 \
  --files "$HOME/spark-sql.py" \
  --conf "SPARK.EXECUTOR.EXTRAJAVAOPTIONS=-
  DJAVA.SECURITY.AUTH.LOGIN.CONFIG=/HOME/HDPSPARKPRD/SPARK-
  ```

```
HDPSPARKPRDKEYTAB-JAAS.CONF -DJAVA.SECURITY.KRB5.CONF=/ETC/KRB5.CONF -
DJAVAX.SECURITY.AUTH.USESUBJECTCREDSONLY=TRUE" \
--conf "SPARK.DRIVER.EXTRAJAVAOPTIONS=-DJAVA.SECURITY.AUTH.LOGIN.CONFIG=/
HOME/HDPSPARKPRD/SPARK-HDPSPARKPRDKEYTAB-JAAS.CONF
 -DJAVA.SECURITY.KRB5.CONF=/ETC/KRB5.CONF -
DJAVAX.SECURITY.AUTH.USESUBJECTCREDSONLY=TRUE" \
--co
nf "SPARK.IO.COMPRESSION.CODEC=ORG.APACHE.SPARK.IO.LZ4COMPRESSIONCODEC" \
$HOME/spark-sql.py
```

```
$ spark3-submit \
--name pt_rpt_streams \
--master=yarn --deploy-mode=cluster \
--driver-memory 4G \
--executor-memory 4G --executor-cores 3 \
--num-executors 4 \
--files "$HOME/spark-sql.py" \
--conf "SPARK.EXECUTOR.EXTRAJAVAOPTIONS=-
DJAVA.SECURITY.AUTH.LOGIN.CONFIG=/HOME/HDPSPARKPRD/SPARK-
HDPSPARKPRDKEYTAB-JAAS.CONF -DJAVA.SECURITY.KRB5.CONF=/ETC/KRB5.CONF -
DJAVAX.SECURITY.AUTH.USESUBJECTCREDSONLY=TRUE" \
--conf "SPARK.DRIVER.EXTRAJAVAOPTIONS=-DJAVA.SECURITY.AUTH.LOGIN.CONFIG=/
HOME/HDPSPARKPRD/SPARK-HDPSPARKPRDKEYTAB-JAAS.CONF
 -DJAVA.SECURITY.KRB5.CONF=/ETC/KRB5.CONF -
DJAVAX.SECURITY.AUTH.USESUBJECTCREDSONLY=TRUE" \
--c
onf "SPARK.IO.COMPRESSION.CODEC=ORG.APACHE.SPARK.IO.LZ4COMPRESSIONCODEC" \
$HOME/spark-sql.py
```

- Sample spark-submit commands with an inline profile configuration you can run on the Cloudera Data Engineering workloads.

```
$ CDE_CONFIG_PROFILE=yarn \
spark-submit \
--name pt_rpt_streams --master=yarn \
--deploy-mode=cluster --driver-memory 4G \
--executor-memory 4G --executorcores 3 \
--num-executors 4 --files "$HOME/spark-sql.py" \
--conf "SPARK.EXECUTOR.EXTRAJAVAOPTIONS=-
DJAVA.SECURITY.AUTH.LOGIN.CONFIG=/HOME/HDPSPARKPRD/SPARK-
HDPSPARKPRDKEYTAB-JAAS.CONF -DJAVA.SECURITY.KRB5.CONF=/ETC/KRB5.CONF -
DJAVAX.SECURITY.AUTH.USESUBJECTCREDSONLY=TRUE" \
--conf "SPARK.DRIVER.EXTRAJAVAOPTIONS=-DJAVA.SECURITY.AUTH.LOGIN.CONFIG=/
HOME/HDPSPARKPRD/SPARK-HDPSPARKPRDKEYTAB-JAAS.CONF
 -DJAVA.SECURITY.KRB5.CONF=/ETC/KRB5.CONF -
DJAVAX.SECURITY.AUTH.USESUBJECTCREDSONLY=TRUE" \
--c
onf "SPARK.IO.COMPRESSION.CODEC=ORG.APACHE.SPARK.IO.LZ4COMPRESSIONCODEC" \
$HOME/spark-sql.py
```

```
$ CDE_CONFIG_PROFILE=vc-1 \
spark3-submit \
--name pt_rpt_streams \
--master=yarn --deploy-mode=cluster \
--driver-memory 4G --executor-memory 4G \
--executor-cores 3 --num-executors 4 \
--files "$HOME/spark-sql.py" \
--conf "SPARK.EXECUTOR.EXTRAJAVAOPTIONS=-
DJAVA.SECURITY.AUTH.LOGIN.CONFIG=/HOME/HDPSPARKPRD/SPARK-
HDPSPARKPRDKEYTAB-JAAS.CONF -DJAVA.SECURITY.KRB5.CONF=/ETC/KRB5.CONF -
DJAVAX.SECURITY.AUTH.USESUBJECTCREDSONLY=TRUE" \
--conf
```

```
"SPARK.DRIVER.EXTRAJAVAOPTIONS=-DJAVA.SECURITY.AUTH.LOGIN.CONFIG=/HOME/
HDPSPARKPRD/SPARK-HDPSPARKPRDKEYTAB-JAAS.CONF -DJAVA.SECURITY.KRB5.CONF=/
ETC/KRB5.CONF -DJAVAX.SECURITY.AUTH.USESUBJECTCREDSONLY=TRUE" \
--c
onf "SPARK.IO.COMPRESSION.CODEC=ORG.APACHE.SPARK.IO.LZ4COMPRESSIONCODEC" \
$HOME/spark-sql.py
```

Using the migration tool in a docker container

You can run the docker image in an interactive mode after you mount the config.yaml and credentials files into the docker container.

Run the migration tool in a docker container

Mount the config.yaml and credentials files into the docker container and run the docker image in the interactive mode. You have to activate the tool after running the tool before you run `spark-submit` commands.

Procedure

1. Run the docker tool.

```
$ docker run -it \
-v <path-to-yaml-file>/config.yaml:/home/cdpuser1/.cde/config.yaml:ro \
-v <path-to-credential-file>/credentials:/home/cdpuser1/.cde/credentials:
ro \
<customers-docker-registry-for-cdp-private-cloud>/cloudera/dex/dex-migr
ation-tool:<tag>
```

   Example:

```
$ docker run -it \
-v /Users/cdp-compute-cluster/cdpuser1/config.yaml:/home/cdpuser1/.cde/co
nfig.yaml:ro \
-v /Users/cdpuser1/credentials:/home/cdpuser1/.cde/credentials:ro \
docker-registry.example.com/cloudera/dex/dex-migration-tool:1.19.1-b185
```

2. Activate the profile.

```
$ cde-env.sh activate -p vc-1
```

Run sample spark-submit command inside the docker container

After you activate the docker image, you can run your `spark-submit` commands on Cloudera Data Engineering without completely rewriting your existing spark-on-yarn command lines inside the docker container.

- Sample spark-submit commands you can run on the Cloudera Data Engineering workloads.

```
$ spark-submit --name pt_rpt_streams5 --master=yarn --deploy-mode=cluste
r --driver-memory 4G --executor-memory 4G --executor-cores 3 --num-execu
tors 4 --conf spark.yarn.queue=hr --conf "spark.executor.extraJavaOption
s=-Djava.security.auth.login.config=/home/hdpsparkprd/spark-hdpsparkprd-
keytab-jaas.conf -Djava.security.krb5.conf=/etc/krb5.conf -Djavax.securi
ty.auth.useSubjectCredsOnly=true" --conf "spark.driver.extraJavaOptions=
-Djava.security.auth.login.config=/home/hdpsparkprd/spark-hdpsparkprd-ke
ytab-jaas.conf -Djava.security.krb5.conf=/etc/krb5.conf -Djavax.security
.auth.useSubjectCredsOnly=true" --conf "spark.io.compression.codec=org.a
```

```
pache.spark.io.LZ4CompressionCodec" --class org.apache.spark.examples.Sp
arkPi http://qe-repo.s3.amazonaws.com/dex/app-jar/spark-examples_2.11-2.
4.4.jar 22
```

Known Issues and Limitations

This page lists the current known issues and limitations that you might run into while using the cde-env tool.

- Limited to spark-submit commands and does not include spark-shell, pyspark, and sparksql.
- When activating a profile using the cde-env.sh script, there is no validation yet on whether such profile exists. However, if a profile does not exist, it will display an error when running the spark-submit command.
- The following spark-submit flags are not yet supported in Cloudera Data Engineering:
  - --archives
  - --exclude-packages
  - --driver-class-path
  - --driver-library-path
  - --driver-java-options

  You are instead suggested to create Cloudera Data Engineering jobs to handle the above mentioned scenarios.
- Using the profile yarn is not supported in the container version of the migration tool.