

Cloudera Runtime 7.3.1

Data Sharing Reference

Date published: 2020-07-28

Date modified: 2024-12-10

CLOUDERA

<https://docs.cloudera.com/>

Legal Notice

© Cloudera Inc. 2025. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

Contents

Knox topologies.....	4
Sample Spark workload to access data.....	7
REST Catalog API calls throughput.....	8

Knox topologies

Learn about the Knox topologies needed for proxying the authorization requests from external users for Cloudera Data Sharing.

Topology name: cdp-share-access.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<topology>
  <uri>https://[***KNOX-HOST***]:8443/gateway/cdp-share-access</uri>
  <name>cdp-share-access</name>
  <timestamp>1714251374810</timestamp>
  <generated>false</generated>
  <redeployTime>0</redeployTime>
  <gateway>
    <provider>
      <role>federation</role>
      <name>JWTProvider</name>
      <enabled>true</enabled>
      <param>
        <name>knox.token.exp.server-managed</name>
        <value>true</value>
      </param>
    </provider>
    <provider>
      <role>identity-assertion</role>
      <name>Default</name>
      <enabled>true</enabled>
      <param>
        <name>group.mapping.$PRIMARY_GROUP</name>
        <value>(not (member username))</value>
      </param>
    </provider>
  </gateway>
  <service>
    <role>KNOXTOKEN</role>
    <param>
      <name>knox.token.ttl</name>
      <value>36000000</value>
    </param>
    <param>
      <name>knox.token.exp.server-managed</name>
      <value>true</value>
    </param>
    <param>
      <name>gateway.knox.token.limit.per.user</name>
      <value>-1</value>
    </param>
  </service>
  <service>
    <role>HMS-API</role>
    <url>http://[***HMS-HOST***]:8090</url>
  </service>
</topology>
```

Topology name: cdp-share-management.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<topology>  <uri>http://[***KNOX-HOST***]:8443/gateway/cdp-share-manageme
nt</uri>
  <name>cdp-share-management</name>
  <timestamp>1711399642000</timestamp>
  <generated>>false</generated>
  <redeployTime>0</redeployTime>
  <gateway>
    <provider>
      <role>authentication</role>
      <name>ShiroProvider</name>
      <enabled>>true</enabled>
      <param>
        <name>main.invalidRequest</name>
        <value>org.apache.shiro.web.filter.InvalidRequestFilter</value>
      </param>
      <param>
        <name>main.invalidRequest.blockBackslash</name>
        <value>>false</value>
      </param>
      <param>
        <name>main.invalidRequest.blockNonAscii</name>
        <value>>false</value>
      </param>
      <param>
        <name>main.invalidRequest.blockSemicolon</name>
        <value>>false</value>
      </param>
      <param>
        <name>main.pamRealm</name>
        <value>org.apache.knox.gateway.shirorealm.KnoxPamRealm</value>
      </param>
      <param>
        <name>main.knoxAnonFilter</name>
        <value>org.apache.knox.gateway.filter.AnonymousAuthFilter</val
ue>
      </param>
      <param>
        <name>urls./knoxtoken/api/v1/jwks.json</name>
        <value>knoxAnonFilter</value>
      </param>
      <param>
        <name>main.pamRealm.service</name>
        <value>login</value>
      </param>
      <param>
        <name>sessionTimeout</name>
        <value>30</value>
      </param>
      <param>
        <name>urls./**</name>
        <value>authcBasic</value>
      </param>
    </provider>
    <provider>
      <role>identity-assertion</role>
      <name>HadoopGroupProvider</name>
      <enabled>>true</enabled>
      <param>
        <name>hadoop.proxyuser.impersonation.enabled</name>
        <value>>true</value>

```

```

    </param>
    <param>
      <name>hadoop.proxyuser.{user-who-runs-the-script}.users</name>
      <value>*</value>
    </param>
    <param>
      <name>hadoop.proxyuser.{user-who-runs-the-script}.groups</name>
      <value>*</value>
    </param>
    <param>
      <name>hadoop.proxyuser.{user-who-runs-the-script}.hosts</name>
      <value>*</value>
    </param>
    <param>
      <name>CENTRAL_GROUP_CONFIG_PREFIX</name>
      <value>gateway.group.config.</value>
    </param>
  </provider>
</provider>
  <role>authorization</role>
  <name>XASecurePDPKnox</name>
  <enabled>false</enabled>
</provider>
<provider>
  <role>ha</role>
  <name>HaProvider</name>
  <enabled>true</enabled>
  <param>
    <name>RANGER</name>
    <value>enableStickySession=false;noFallback=false;enableLoadB
alancing=true</value>
  </param>
</provider>
</gateway>
<service>
  <role>RANGER</role>
  <url>https://[***RANGER-HOST***]:6182</url>
</service>
<service>
  <role>KNOXTOKEN</role>
  <param>
    <name>knox.token.ttl</name>
    <value>-1</value>
  </param>
  <param>
    <name>knox.token.type</name>
    <value>JWT</value>
  </param>
  <param>
    <name>knox.token.target.url</name>
    <value>cdp-proxy-token</value>
  </param>
  <param>
    <name>knox.token.audiences</name>
    <value>cdp-proxy-token</value>
  </param>
  <param>
    <name>knox.token.client.data</name>
    <value>homepage_url=homepage/home?profile=token&topologies=cdp-
proxy-token</value>
  </param>
  <param>
    <name>knox.token.exp.tokengen.allowed.tss.backends</name>
    <value>JDBCTokenStateService,AliasBasedTokenStateService</value>

```

```

    </param>
    <param>
      <name>knox.token.lifespan.input.enabled</name>
      <value>true</value>
    </param>
    <param>
      <name>knox.token.user.limit.exceeded.action</name>
      <value>RETURN_ERROR</value>
    </param>
    <param>
      <name>knox.token.exp.server-managed</name>
      <value>true</value>
    </param>
  </service>
</topology>

```

Related Information

[Declaring Knox topologies](#)

Sample Spark workload to access data

See an example of an end to end sample workload flow that describes the process for enabling a Spark session, connecting to the HMS REST Catalog server, and running a Spark engine query to access data.



Note: Based on the type of client that is accessing the REST Catalog, refer to the appropriate client-side documentation to configure your client. For example, see the [Tabular documentation](#).

PySpark workload example

The PySpark workload connects to Cloudera HMS REST Catalog with CLIENT_ID and CLIENT_SECRET as credentials and runs workloads from an external system like Databricks, Snowflake, Standalone Spark in Docker, and so on.

```

# © 2024 by Cloudera, Inc. All rights reserved.
# Scripts and sample code are licensed under the Apache License,
# Version 2.0
import pyspark
from pyspark.sql import SparkSession
import argparse

parser = argparse.ArgumentParser(description="Spark WorkLoad Script")
parser.add_argument("--credential", help="ClientId:Secret")
args = parser.parse_args()

conf = (
    pyspark.SparkConf()
    .setAppName('Fetch Employees')
    .setMaster('local[*]')
    .set('spark.jars', '/opt/spark/jars/iceberg-spark-runtime-3.5_2.13-1.5.2.jar')
    .set('spark.files', '/opt/spark/conf/log4j2.properties')
    #packages
    .set('spark.jars.packages', 'org.apache.iceberg:iceberg-spark-runtime-3.5_2.13-1.5.2.jar')
    #SQL Extensions
    .set('spark.sql.extensions', 'org.apache.iceberg.spark.extensions.IcebergSparkSessionExtensions')
)

```

```

#Configuring Catalog
.set('spark.sql.defaultCatalog', 'demo')
.set('spark.sql.catalog.demo', 'org.apache.iceberg.spark.SparkCatalog')

.set('spark.sql.catalog.demo.type', 'rest')
.set('spark.sql.catalog.demo.uri', 'https://<datalake-hostname>/<datalake-name>/cdp-share-access/hms-api/icecli')
.set('spark.sql.catalog.demo.io-impl', 'org.apache.iceberg.aws.s3.S3FileIO')
.set('spark.sql.catalog.demo.s3.client-factory-impl', 'org.apache.iceberg.aws.s3.DefaultS3FileIOAwsClientFactory')
.set('spark.sql.catalog.demo.credential', args.credential)
.set('spark.sql.catalog.demo.default-namespace', 'emp_data')
)

## Start Spark Session
spark = SparkSession.builder.config(conf=conf).getOrCreate()
spark.sparkContext.setLogLevel("DEBUG")
print("Spark Job Running...")
print("##### Credential: #####")
print(args.credential)

## list databases
dblist=spark.catalog.listDatabases()
print("##### List Databases #####")
print(dblist)
spark.sparkContext.parallelize([dblist]).coalesce(1).saveAsTextFile("file:///opt/spark/out/databases")

## list tables
tableList=spark.catalog.listTables("demo.emp_data")
print("##### List Tables #####")
print(tableList)
spark.sparkContext.parallelize([tableList]).coalesce(1).saveAsTextFile("file:///opt/spark/out/tables")

## Run a Query
print("##### Query: fetch all the employees of 'department -> d006' #####")
results=spark.sql("select emp_data.employees.first_name, emp_data.employees.last_name, emp_data.departments.dept_name "
                  "from emp_data.employees, emp_data.departments, emp_data.dept_emp "
                  "where emp_data.employees.emp_no=emp_data.dept_emp.emp_no "
                  "and emp_data.dept_emp.dept_no=emp_data.departments.dept_no "
                  "and emp_data.departments.dept_no='d006'")
print(results.show())
results.coalesce(1).write.option("header", "true").csv("file:///opt/spark/out/query_results")

```

Related Information

[Running Apache Spark Applications](#)

REST Catalog API calls throughput

Based on the stress test done on the REST Catalog it is recommended that as the throughput on REST Catalog increases, heap requirement for REST Catalog service has to be increased.

As the throughput increase in REST Catalog, increase the heap size accordingly to avoid timed out API requests.

Runs	Heap	API Calls Throughput	Run Time	Total requests	Response: Success	Response: Time Out	Error
1	8 GB	~30/s	~17hrs	1401698	1325580	76118	5.742
2	16 GB	~30/s	~26hrs	2756130	2709228	46902	1.731
3	16 GB	~15/s	~52hrs	2755525	2731253	24272	0.888
4	32 GB	~15/s	~40hrs	2162994	2158488	4506	0.208

Related Information

[Supported REST Catalog APIs for accessing the data](#)