Data Warehouse Preparation

Complete the following data warehouse configuration steps before connecting AtScale to your data warehouse.

Before You Begin

For Hadoop platforms, it is required that the platform be configured to use Storage-Based Authorization.

If you are connecting to a Hadoop data warehouse, ensure that your cluster meets the minimum system requirements for AtScale.

Note that you may have to tune your data warehouse size to satisfy the performance requirements specific to your work load.

About This Task

The following procedure uses terminology common to security managers such has Hive and Ranger. You may have to translate the permissions referenced below to your specific security infrastructure. See Table 1. for guidelines on how sql security manager permissions map to SQL and file system ACL permissions.

Table 1. Guidelines of Mapping Common Security Manager Permissions to SQL and ACL Permissions

Security Manager Permission	SQL Permission	ACL Permission
SELECT	SELECT	read
SELECT, UPDATE, CREATE, DROP	SELECT, UPDATE, CREATE, DROP	read, write, execute

You should consult with your data warehouse vendor's documentation on the exact permissions required to implement the equivalent permissions listed in Table 1. For example, see Hive-to-Ranger Permission Mapping and Authorization With Apache Sentry.

Steps For All Data Warehouses

Follow this procedure to prepare your data warehouse for working with AtScale. When creating permissions, use Table 1 and your vendor's security manager documentation to determine the equivalent permission for your platform:

1. Create the `atscale` data warehouse service account user. **Optional:** Set a password for this user.
2. Create the AtScale aggregate schema. AtScale will write aggregate tables to this schema.

3. Grant the 'SELECT, UPDATE, CREATE, DROP' permissions (see Table 1) on the AtScale aggregate schema to the `atscale` service account user. BI Tool user accounts should not have the SELECT permission on this schema.
4. Grant the SELECT permission on your source data to the `atscale` service account user.
5. If using Impersonation grant the select permission on your source data to your Business Intelligence Tool users.

Additional Steps For All Hadoop Data Warehouses

1. Follow all steps listed in the "Steps for All Data Warehouses" section.
2. Create an HDFS home directory for the `atscale` user.
3. Grant ownership of the HDFS atscale user home directory to the `atscale` user.
4. Create the AtScale UDAF schema. AtScale will install its User-Defined-Aggregation-Functions (UDAF) to this location. This can be the same as the AtScale Aggregate schema if desired.
5. Grant the 'SELECT, UPDATE, CREATE, DROP' permissions (see Table 1) on this schema to the `atscale` service account user.
6. If enabling AtScale impersonation then grant the SELECT permission on the AtScale UDAF schema to all Active Directory users who will run queries through AtScale (aka your Business Intelligence Tool users).
7. Create an HDFS directory for AtScale installation. AtScale refers to this as the "FILESYSTEM INSTALL PATH". The `atscale` service account must have permission to write to this location.
8. If you want AtScale to access your cluster using Kerberos:

    1. Install and configure the Kerberos client package or packages for your environment on the AtScale host(s).
    2. Obtain a `.keytab` file from your Kerberos administrator, and save it on the AtScale host(s). Remember the path because it will be needed when configuring AtScale for Kerberos operation.

Additional Steps For HDP 3.X Hadoop Data Warehouses

> **Important:** AtScale requires that Hive and SparkSQL read from the same tables in a single metastore. Additionally SparkSQL cannot read Hive ACID tables. The following steps configure HDP 3.x to disable the use of Hive ACID tables by default and to allow the Hive and the Spark Thrift servers to use the same metastore.

1. Follow all steps listed in the "Steps for All Data Warehouses" and "Additional Steps for All Hadoop Data Warehouses" sections, and ensure that Hive Metastore is configured to use storage-based authorization.
2. Using Ambari update the following Hive configuration settings:

   1. In both the 'Advanced hive-interactive-site' and 'Advanced hive-site' configurations set:

      - **hive.strict.managed.tables** = False

   2. Uncheck both of the following (effectively setting these to False):

      - **'Create Tables as ACID Insert Only'**
      - **'Create Tables as Full ACID'**

   3. Restart the Hive Service

3. Using Ambari update the following Spark2 configuration settings:

   1. In both the 'Advanced spark2-defaults' and 'Advanced spark2-thrift-sparkconf' configurations set the path of the sql data warehouse to the same value that is used for Hive (found in Ambari: `Hive Service -> CONFIGS -> ADVANCED -> General -> Hive Metastore Warehouse directory`), for example:

      - **spark.sql.warehouse.dir** = /warehouse/tablespace/managed/hive (change from /apps/spark/warehouse)

   2. Under 'spark2-hive-site-override', set:

      - **metastore.catalog.default** = hive (change from spark)

   3. Restart the Spark Service. Run the Spark Thrift Service as the Hive user. For example:

      ```
      [root@sandbox-hdp /]# su - hive
      [hive@sandbox-hdp ~]$ cd /usr/hdp/3.0.1.0-187/spark2/sbin/
      [hive@sandbox-hdp ~]$ ./start-thriftserver.sh  --master yarn-client
      ```

   4. Connect to SparkSQL and confirm that SparkSQL can read Hive table contents.

4. If necessary, convert desired transactional Hive tables using `CREATE TABLE ... TBLPROPERTIES ('transactional'='false') AS SELECT ...` .

> **Attention:** HDP 3.1.0 contains bug HIVE-21301 which causes the AtScale Data Sources panel to not displaying all query-able entities. This issue is resolved with HDP 3.2.0.

Additional Steps For MapR Hadoop Data Warehouses

1. Follow all steps listed in the "Steps for All Data Warehouses" and "Additional Steps for All Hadoop Data Warehouses" sections, and ensure that Hive Metastore is configured to use storage-based authorization.
2. You must install the MapRClient on all AtScale hosts.

3. If connecting to a secure MapR cluster you must use maprlogin to obtain a ticket for the AtScale service. For example, if you are using Kerberos with MapR:

    1. The Hadoop administrator must generate a MapR ticket with this command:

        ```
        maprlogin kerberos
        ```

        The command output includes the location of the generated ticket, for example: `/tmp/maprticket_0`. Make sure the expiration is set to the desired duration with the optional parameter `-duration"`. For more information, see maprlogin.

    2. The AtScale administer must copy the ticket file generated in step 1 to a location on the AtScale host (or hosts if installing an AtScale cluster).
    3. The AtScale system administrator must then add this environment variable to the AtScale host (or hosts) system's profile:

        ```
        export MAPR_TICKETFILE_LOCATION=<ticket file location on disk>
        ```

        For example: `export MAPR_TICKETFILE_LOCATION="/tmp/maprticket_0"`

> **Attention:** HIVE-16811 causes Query Dataset statistics queries to fail with an `IllegalArgumentException` error. As a result, aggregates may not be built from Query Datasets. This problem affects MapR distributions running Hive 2.x. This issue can be worked-around by either using AtScale User-defined Aggregates or by using the Spark SQL Engine for the System Query role. Root cause resolution requires that MapR back-port `HIVE-16811`.

What To Do Next

Perform on of the following tasks:

- Install Stand-Alone AtScale
- Upgrading Stand-Alone AtScale