HOW IT'S DONE | DELEGATED AUTHORIZATION

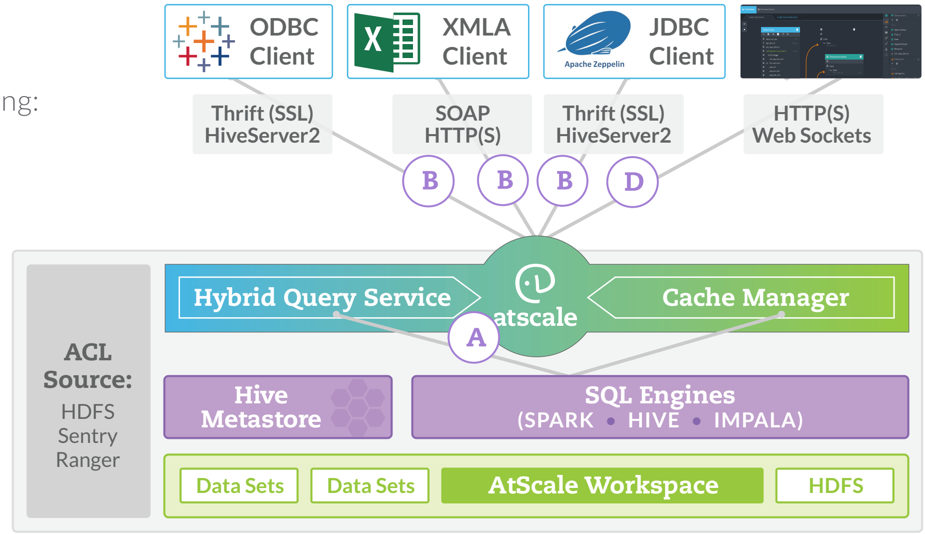
Secure Business Intelligence in the Age of Hadoop: AtScale True Delegation

IN THIS ARTICLE: See how AtScale allows you to manage authorization of user access to *Hadoop* data.

AtScale Query Types

AtScale defines query types as the following:

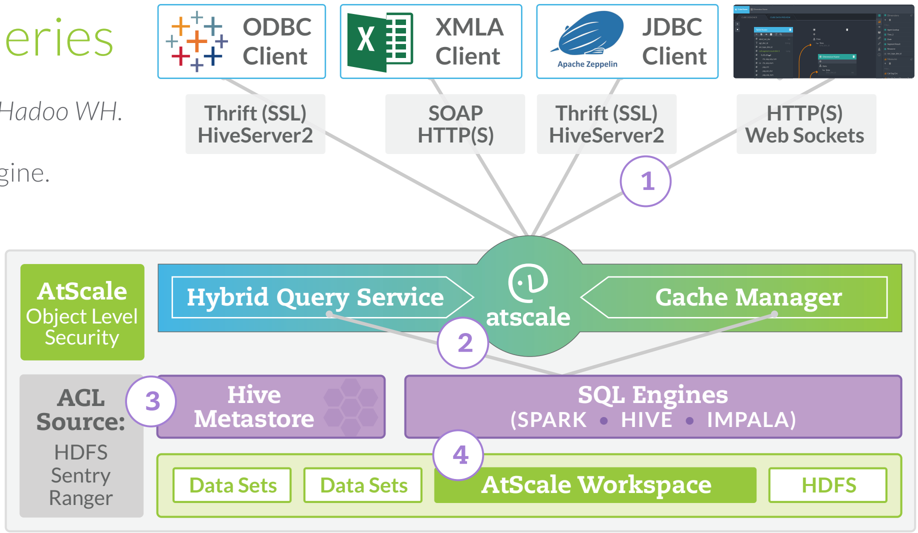
- B Runtime User Queries
- D Design Center User Queries
- A AtScale Queries



Design Center User Queries

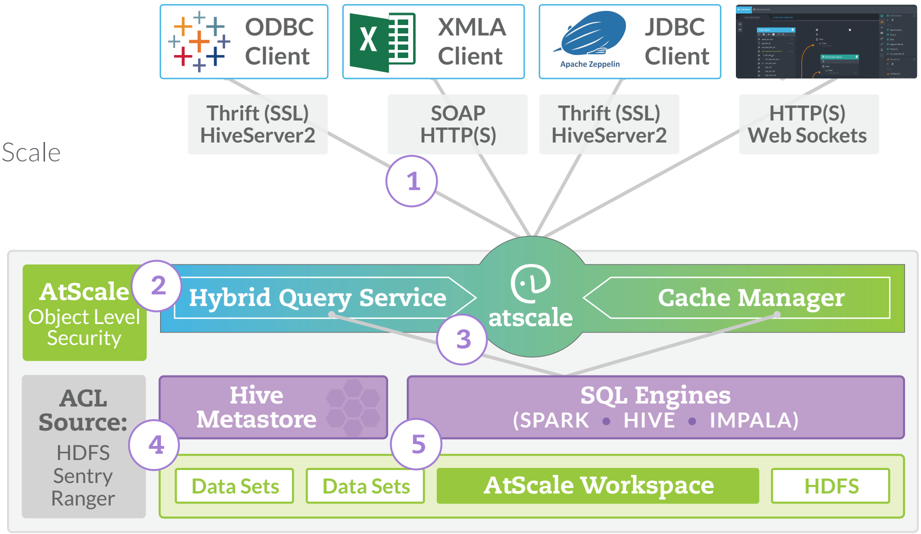
1. Cube designer connects to and queries *Hadoo WH.*
2. Query Service submits query to SQL engine.
3. SQL engine applies data-level ACLs.
4. Query is distributed to query workers.

NOTE: Depending on the user Design Center Queries may be disabled.



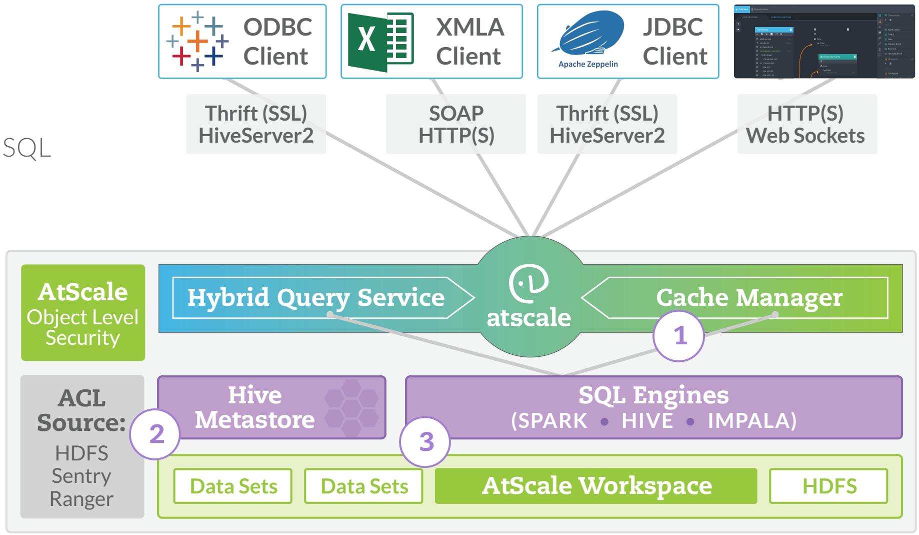
Runtime Query User: *Raw Data Query*

1. Client submits cube-scoped query to AtScale engine.
2. AtScale object security validates cube access.
3. Query service submits query to SQL engine as the delegated user.
4. SQL engine applies data-level ACLs.
5. Query is distributed to query workers.

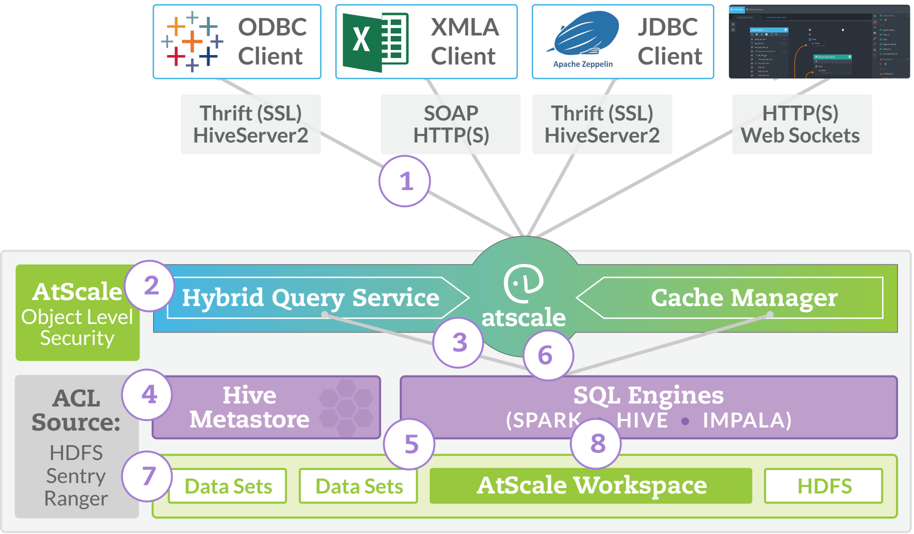


System Query: *Aggregate Creation*

1. AtScale submits table creation query to SQL engine as the AtScale user.
2. SQL engine applies data-level ACLs..
3. SQL engine executes SELECT on authorized data and CREATE on AtScale Workspace.



Runtime Query User: *Cache Query*



1. Client submits cube-scoped query to AtScale engine.
2. AtScale object security validates cube access.
3. Query service submits query to SQL engine as the delegated user.
4. SQL engine applies data-level ACLs.
5. Canary query returns failure or success with 0 rows.
6. If canary query passes, cache query submitted as AtScale user.
7. SQL engine applies data-level ACLs.
8. SQL engine query executed on AtScale workspace.

Details

BI Tool and ODBC Driver

- User credentials will be passed from client through to *Hadoop* cluster using *Kerberos* principals obtained from active directory.
- Requires client query application (Tableau, Excel, MSTR) to have access to and be able to pass end-user *Kerberos* ticket.

AtScale Engine

- Once AtScale has access to the end-user's *Kerberos* credentials, this information will be used when sending raw queries to the SQL cluster over JDBC. This approach to delegation will be supported for Impala, SparkSQL, and Hive query engines. If using SparkSQL or Hive you will need to use the AtScale Installed versions.

Data Authorization Approach

- Once the delegation approach described above has been enabled and configured, all "aggregate miss" queries that are submitted to Hadoop by AtScale will be executed as the BI client user that executed the query (using their *Kerberos* principal). As such, all aggregate miss queries will reflect the data authorizations of the querying BI client user (as configured in Sentry, Ranger, or via HDFS authorizations).
- Aggregate Data Authorization
  - Aggregate creation will be executed using the service account of the AtScale service as configured (not the BI client user). The AtScale Service Account will need read permissions on all tables it uses to create aggregates. Permissions on all aggregates created by AtScale will be restricted to the AtScale connection-ˇlevel service account that created them.

- Aggregate querying (in the case of an "aggregate hit" query) will have two components:
  - First, AtScale will execute a "canary" query that represents an equivalent "aggregate miss" query. This pre-ˇflight query will contain constraints (such as a "LIMIT 0") to limit the query execution time to 10s of milliseconds while still evaluating the data authorizations of the query. This query will be executed using the user credentials of the BI client user.
  - If the "canary" query returns a successful query (that is, passes the data authorization test) a second query (represented the aggregate hit query with no additional constraints) will be executed using the AtScale connection-level service account.

Configuration Details

In order to utilize the Delegated Authorization approach, the following configuration steps are required:

- Clients and BI Tools need to be set up to use Kerberos authentication (with ticket forwarding enabled) to connect to the AtScale engine.
- Active Directory configuration enables delegation by the AtScale account. The way to do so is:
  - Create a user account for the service.
  - The account needs to have a Service Principal Name (SPN): (Active Directory SPN Details).
  - AtScale needs a headless key tab associated with the SPN you set in the previous step.
  - Users need to have Delegation Enabled within Active Directory: (Active Directory Delegation Details).
  - Users that will be using AtScale also need to be allowed to delegate.
  - For users that will login to AtScale the login name and the UPN need to be the same case. For example, if you log in with user@DOMAIN.COM the UPN of the user in active directory should match the case, user@DOMAIN.COM.
- You will need to use the AtScale provided Spark sql and HiveServer2
- AtScale will need a Hive Metastore database and HDFS location it has read and write access to store its Aggregates.
- AtScale Service will need read and write access to the Hive Metastore database and HDFS locations of data that will be used in AtScale Cubes. (Currently this is due to Hive/Tez, we believe this is a bug and are working through it. If you don't have write access to the HDFS folder you are reading from, it will error out). (Only if using Hive/Tez)
- AtScale will need a database setup that AtScale has Read/Write/UDF Create access to, all other users using AtScale will need Read Access. The reason for this is AtScale uses UDF's and to you have to tie the UDF to a specific database which all users need to be able to Read from.
- You will need to set it up so that Hive/tez is used for large interactive and system aueries within AtScale, while Spark SQL is used for small interactive queries. (Or *Impala* for both if using *Impala*)
- When Accessing Yarn it will be using the User you have delegated as not the AtScale user when running raw queries. When running aggregate queries and aggregate builds it will be running as the AtScale user.
- If Using Hive AtScale will need the following entries added to the core-ˇsite.xml to be deployed across the cluster: (Where atscale is replaced with the user you created in Active Directory)

```
<property>
    <name>hadoop.proxyuser.atscale.group</name>
    <value>*</value>
</property>

<property>
    <name>hadoop.proxyuser.atscale.hosts</name>
    <value>*</value>
</property>
```

Examples

In this example we will be use 4 users:
User_a, User_b, User_c, Atscale.

Each user belongs to a different group.
User_a belongs to group_a.
User_b belongs to group_b.
User_c belongs to group_c.

And in this case each Group has a database it can access.
User_a, group_a has "read and write" permission for database_a.
User_b, group_b has "read and write" permission for database_b.
User_c, group_c has "read and write" permission for database_c.
Atscale has "read and write" permission for atscale_database.

In our example user_a and user_b will be users of AtScale, which means the Atscale user will look like:

AtScale belongs to group_a and group_b.
Which means it has access to both database_a and database_b, but not to database_c.

When using AtScale, user_a will be able to see any data in database_a and aggregates based on that data, but no data in database_b, database_c or atscale_database.

When using AtScale user_b will be able to see any data in database_b and aggregates based on that data, but no data in database_a, database_c or atscale_database.

User_c will not be able to see what is in database_a, database_b or the or atscale_database..