Data Governance 3

Configuring Apache Atlas

Date of Publish: 2018-04-01

http://docs.hortonworks.com

Contents

Configure Atlas High Availability

For information about configuring High Availability (HA) for Apache Atlas, see "Apache Atlas High Availability" in the Ambari Managing and Monitoring a Cluster guide.

Related Information

Configuring Atlas high availability

Configuring Apache Atlas Security

This section provides information on how to configure Apache Atlas security.

Enable the Ranger plugin

The Ranger Atlas plugin enables you to establish and enforce global security policies based on data classifications.

For more information, see "Enabling the Ranger Atlas Plugin" in the HDP Security guide.

Related Information

Enable the Atlas Ranger Plugin

Configure Atlas Tagsync in Ranger

You can use Ranger TagSync to synchronize the Ranger tag store with Apache Atlas metadata.

Note:

Before configuring Atlas Tagsync in Ranger, you must enable Ranger Authorization in Atlas by enabling the Ranger Atlas plugin in Ambari.

For information about configuring Atlas Tagsync in Ranger, see "Customize Services: Tagsync" in the HDP Security guide.

Related Information

Enable the Atlas Ranger Plugin

Configure Ranger Tagsync

Additional requirements for Atlas with Ranger and Kerberos

Currently additional configuration steps are required for Atlas with Ranger and in Kerberized environments.

Additional Requirements for Atlas with Ranger

When Atlas is used with Ranger, perform the following additional configuration steps:

Important: These steps are not required for Ambari-2.4.x and higher versions. For Ambari-2.4.x and higher, these steps will be performed automatically when Atlas is restarted.

- Create the following HBase policy:
  - table: atlas_titan, ATLAS_ENTITY_AUDIT_EVENTS

    user: atlas

permission: Read, Write, Create, Admin

- Create following Kafka policies:
  - topic=ATLAS_HOOK

    permission=publish, create; group=public

    permission=consume, create; user=atlas (for non-kerberized environments, set group=public)
  - topic=ATLAS_ENTITIES

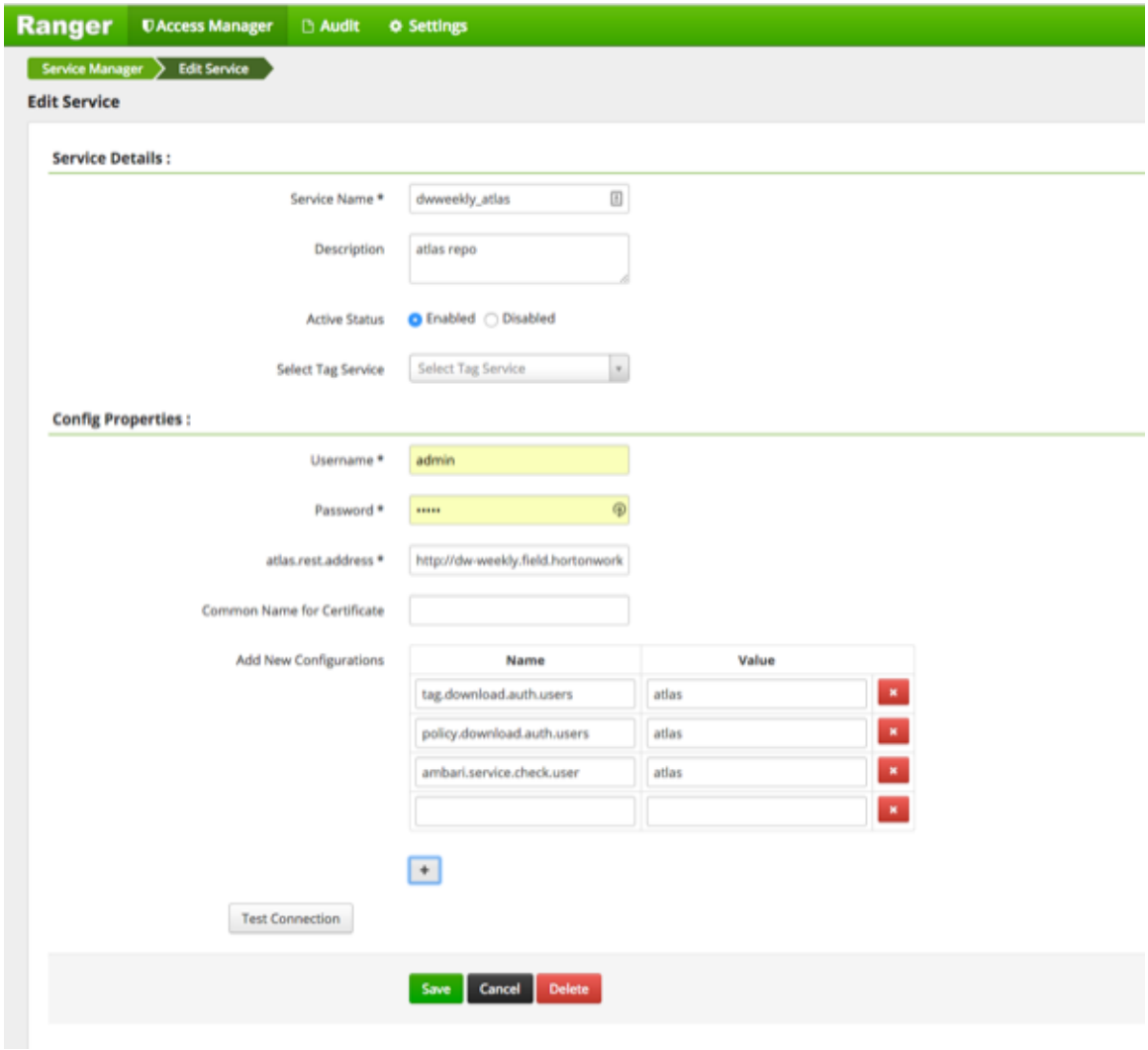
    permission=publish, create; user=atlas (for non-kerberized environments, set group=public)

    permission=consume, create; group=public

You should also ensure that an Atlas service is created in Ranger, and that the Atlas service includes the following configuration properties:

Table 2.4. Ranger Atlas Service Kerberos Properties

Property	Value
tag.download.auth.users	atlas
policy.download.auth.users	atlas
ambari.service.check.user	atlas



Note:

If the Ranger Atlas service is not created after enabling the plugin and restarting Atlas, that indicates that either there is already a policy JSON on the Atlas host (in the /etc/ranger/<service_name>/policycache/ directory), or Ambari was unable to connect to Ranger Admin during the Atlas restart. The solution for the first issue is to delete or move the policycache file, then restart Atlas.

- You can click the Test Connection button on the Ranger Atlas Service Details page to verify the configuration settings.
- You can also select Audit > Plugins in the Ranger Admin UI to check for the latest Atlas service entry.

Additional Requirements for Atlas with Kerberos without Ranger

When Atlas is used in a Kerberized environment without Ranger, perform the following additional configuration steps:

- Start the HBase shell with the user identity of the HBase admin user ('hbase')
- Execute the following command in HBase shell, to enable Atlas to create necessary HBase tables:
  - grant 'atlas', 'RWXCA'
- Start (or restart) Atlas, so that Atlas would create above HBase tables
- Execute the following commands in HBase shell, to enable Atlas to access necessary HBase tables:
  - grant 'atlas', 'RWXCA', 'atlas_titan'
  - grant 'atlas', 'RWXCA', 'ATLAS_ENTITY_AUDIT_EVENTS'
- Kafka – To grant permissions to a Kafka topic, run the following commands as the Kafka user:

```
/usr/hdp/current/kafka-broker/bin/kafka-acls.sh --authorizer
 kafka.security.auth.SimpleAclAuthorizer
--authorizer-properties zookeeper.connect=hostname:2181 --add --operation
 All --allow-principal User:atlas
--topic ATLAS_HOOK
/usr/hdp/current/kafka-broker/bin/kafka-acls.sh --authorizer
 kafka.security.auth.SimpleAclAuthorizer
--authorizer-properties zookeeper.connect=hostname:2181 --add --operation
 All --allow-principal User:atlas
--topic ATLAS_ENTITIES
```

Related Information

Create an HBase Policy
Create a Kafka Policy
Create an Atlas Service

Enable Atlas HTTPS

HTTPS encryption protects Apache Atlas data as it into, through, and out of a Hadoop cluster.

For information about enabling HTTPS for Apache Atlas, see "Enable SSL for Apache Atlas" in the HDP Security guide.

Related Information

Enable SSL for Apache Atlas

Hive CLI security

If you have Oozie, Storm, or Sqoop Atlas hooks enabled, the Hive CLI can be used with these components.

You should be aware that the Hive CLI may not be secure without taking additional measures.

Configure the Knox proxy for Atlas

You can avoid exposing Atlas hosts and ports by using Apache Knox as a proxy.

Procedure

1. On the Ambari Dashboard, select Knox > Configs > Advanced Topology, then add the following services:

```
<service>
  <role>ATLAS-API</role>
  <url><atlas-server-host>:21000</url>
</service>

<service>
  <role>ATLAS</role>
  <url><atlas-server-host>:21000</url>
</service>
```

2. Click Save to save the new configuration, then click Restart > Restart All Affected to restart Knox.
3. With the Knox proxy enabled, use the following URL format to access the Atlas Dashboard:

```
https://<knox-gateway-host>:<knox-gateway-port>/<gateway-path>/<topology>/
atlas/index.html
```

For example:

```
https://<knox-gateway-host>:8443/gateway/ui/atlas/index.html
```

Use the following format to access the Atlas REST API:

```
https://<knox-gateway-host>:<knox-gateway-port>/<gateway-path>/<topology>/
atlas/
```

For example:

```
curl -i -k -L -u admin:admin -X GET \
'https://<knox-gateway-host>:8443/gateway/{topology}/atlas/api/atlas/v2/
types/typedefs?type=classification&_=1495442879421'
```

Note:

- Apache Atlas HA (High Availability) is not supported with the Atlas Knox proxy.
- Knox SSO is supported with the Atlas Knox proxy, but is not required.

Configuring Atlas Authorization

You can use Apache Atlas authorization to restrict access to Apache Atlas.

About this task

More specifically, you can restrict access to Atlas types and entities, and to administrative operations.

Access to Types

- Create
- Update
- Delete

Access to Entities

- Read
- Create
- Update
- Delete
- Add classification
- Update classification
- Remove classification

Admin Operations

- admin-export
- admin-import

You can use simple or Ranger authorization:

- Simple authorization – Authorization policies are specified using a JSON file.
- Ranger authorization – Authorization policies are specified using Apache Ranger.

Configure Simple Authorization

1. Select Atlas > Configs > Advanced on the Ambari dashboard, then click Advanced application-properties.
2. Set the value of the atlas.authorizer.impl property to simple.
3. Confirm that the value of the atlas.simple.authz.policy.file property is set to {{conf_dir}}/atlas-simple-authz-policy.json.
4. Edit the {{conf_dir}}/atlas-simple-authz-policy.json file to specify authorization settings:

```
vi /usr/hdp/current/atlas-server/conf/atlas-simple-authz-policy.json
```

The following is a sample atlas-simple-authz-policy.json file:

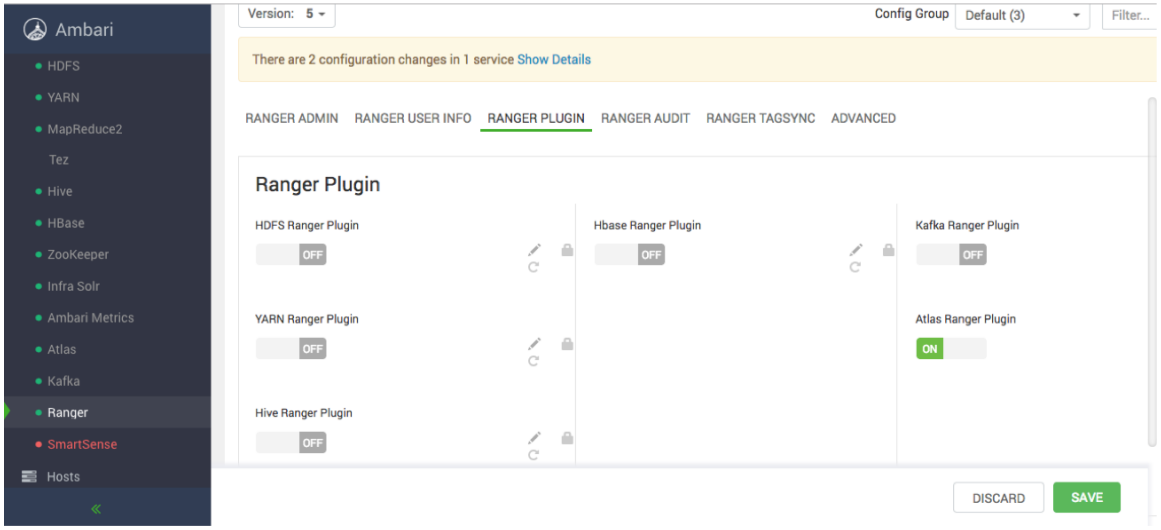
```
{
  "roles": {
    "ROLE_ADMIN": {
      "adminPermissions": [
        {
          "privileges": [ ".*" ]
        }
      ],

      "entityPermissions": [
        {
          "privileges":      [ ".*" ],
          "entityTypes":     [ ".*" ],
          "entityIds":       [ ".*" ],
          "classifications": [ ".*" ]
        }
      ],

      "typePermissions": [
        {
          "privileges":     [ ".*" ],
          "typeCategories": [ ".*" ],
          "typeNames":      [ ".*" ]
```

```
        }
      ]
    },

    "DATA_SCIENTIST": {
      "entityPermissions": [
        {
          "privileges":      [ "entity-read"],
          "entityTypes":     [ ".*" ],
          "entityIds":       [ ".*" ],
          "classifications": [ ".*" ]
        }
      ]
    },

    "DATA_STEWARD": {
      "entityPermissions": [
        {
          "privileges":      [ "entity-read", "entity-create", "entity-
update", "entity-add-classification", "entity-update-classification",
 "entity-remove-classification" ],
          "entityTypes":     [ ".*" ],
          "entityIds":       [ ".*" ],
          "classifications": [ ".*" ]
        }
      ]
    }
  },

  "userRoles": {
    "admin": [ "ROLE_ADMIN" ]
  },

  "groupRoles": {
    "ROLE_ADMIN":      [ "ROLE_ADMIN" ],
    "hadoop":          [ "DATA_STEWARD" ],
    "DATA_STEWARD":    [ "DATA_STEWARD" ],
    "RANGER_TAG_SYNC": [ "DATA_SCIENTIST" ]
  }
}
```

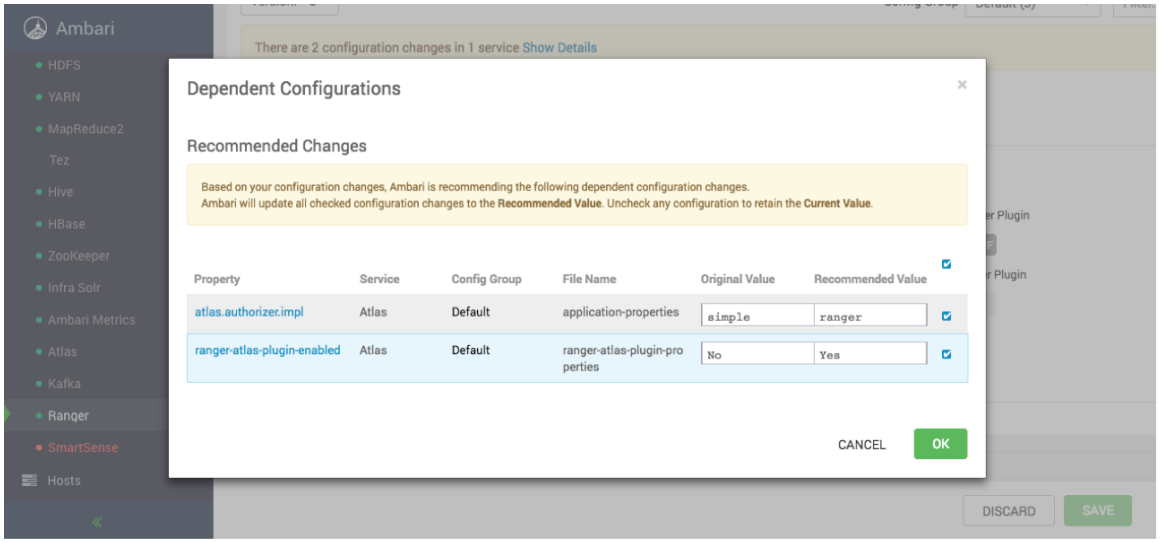
5. To apply these permissions, restart Atlas and any other services that require a restart.

Configure Ranger Authorization

1. On the Ranger Configs page, select the Ranger Plugin tab.
2. Under Atlas Ranger Plugin, select On, then click Save in the black menu bar.

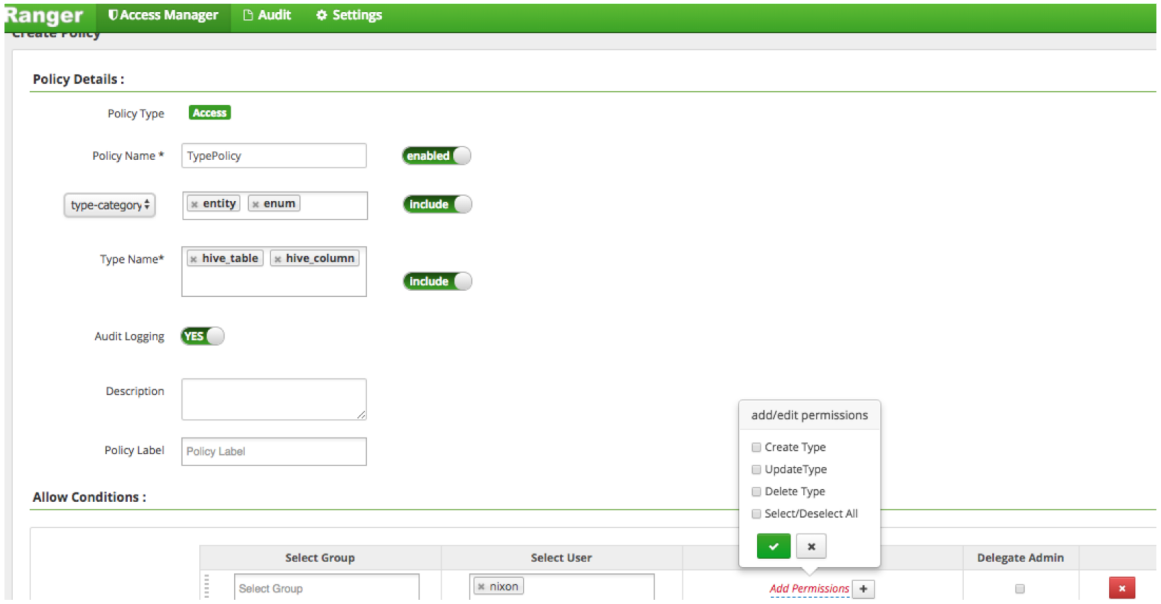


3. A Save Configuration pop-up appears. Type in a note describing the changes you just made, then click Save.
4. A Dependent Configurations pop-up appears. Click OK to confirm the recommended configuration updates.

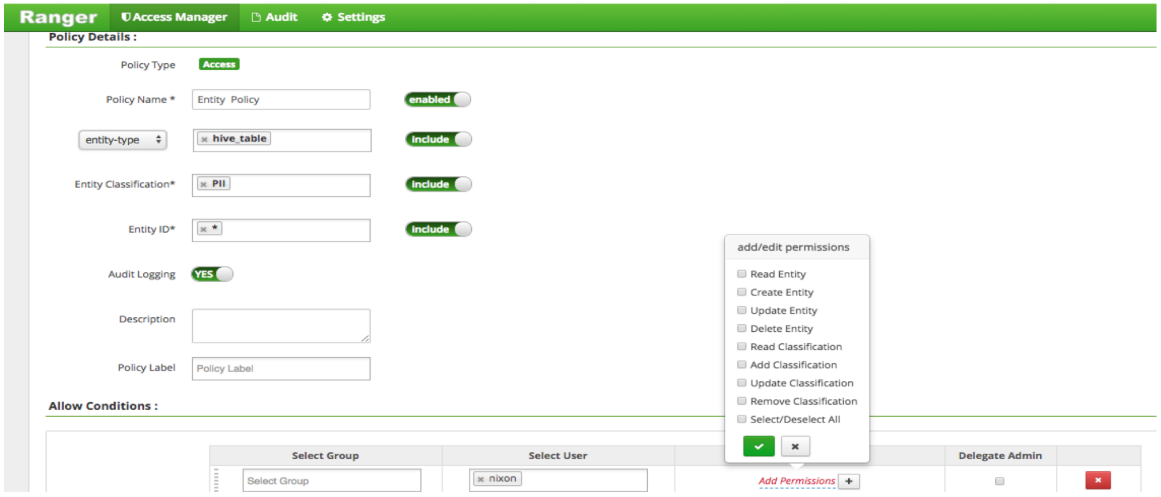


5. Click OK on the Save Configuration Changes pop-up.
6. Select Actions in the navigation menu, then select Restart All Required to restart all services that require a restart.
7. Click Confirm Restart All on the confirmation pop-up to confirm the Storm restart. After the services restart, the Ranger plugin for Atlas is enabled.
8. Log in to the Apache Ranger Web UI.
9. On the Apache Ranger Service Manager page, select an existing service under Atlas.
10. On the List of Policies page, click Add New Policy.
11. Use the Create Policy page to specify the Atlas authorization policy, then click Add. The following figures show sample Ranger policy pages for Atlas authorization.

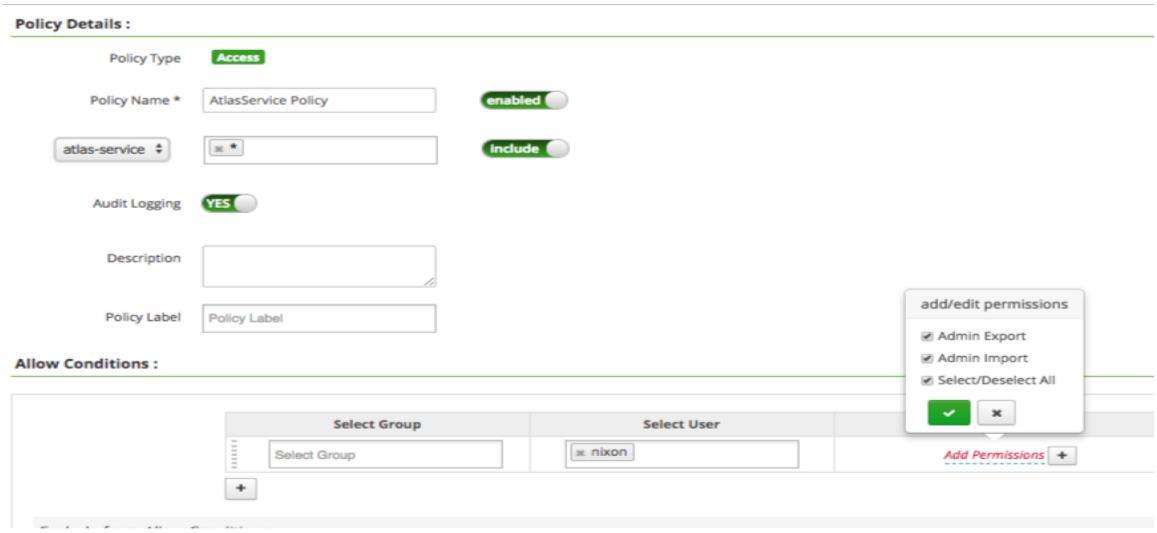
    Authorize Access to Atlas Types



Authorize Access to Atlas Entities



Authorize Access to Atlas Admin Operations



Authorization in the Apache Atlas Web UI

An error message appears in the Atlas Web UI if a user attempts to perform an unauthorized operation.

You should also note that Atlas Search displays all results, but if a user does not have Read permissions for some of the search results, links will not be available to access those entities.