

## Schema Registry Security

Date published: 2020-06-12

Date modified: 2021-06-08



# Legal Notice

© Cloudera Inc. 2024. 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

<b>Schema Registry Authorization through Ranger Access Policies.....</b>	<b>4</b>
Pre-defined Access Policies for Schema Registry.....	4
Add the user or group to a pre-defined access policy.....	5
Create a Custom Access Policy.....	7

# Schema Registry Authorization through Ranger Access Policies

User and group access to various Schema Registry functions is controlled through Apache Ranger.

Pre-defined access policies for Schema Registry allow the administrator to quickly add a user or user group to specify:

- Who can add/evolve schemas to a schema metadata.
- Who can view and edit schemas within a schema metadata.
- Who can upload the ser/des jar files.

If a higher level of granularity is necessary, the administrator can create an access policy and add the user or user-group to this custom policy.

## Related Information

[Pre-defined Access Policies for Schema Registry](#)

[Add the user or group to a pre-defined access policy](#)

[Create a Custom Access Policy](#)

## Pre-defined Access Policies for Schema Registry

Based on a user's responsibilities, you can add users or a user group to one or more of the following pre-defined access policies for Schema Registry and you can specify the type of permission such as Create, Read, Update, and Delete.

The following image shows the pre-defined access policies for Schema Registry:

Policy ID	Policy Name	Policy Labels	Status	Audit Logging	Roles	Groups	Users	Action
1	all - export-import	--	Enabled	Enabled	--	--	streamsmgmr, schemaregistry, rangerlookup, kafka	View, Edit, Delete
2	all - serde	--	Enabled	Enabled	--	--	streamsmgmr, kafka, schemaregistry, rangerlookup	View, Edit, Delete
3	all - schema-group, schema-metadata	--	Enabled	Enabled	--	--	streamsmgmr, kafka, schemaregistry, rangerlookup	View, Edit, Delete
4	all - schema-group, schema-metadata, s...	--	Enabled	Enabled	--	--	streamsmgmr, kafka, schemaregistry, rangerlookup	View, Edit, Delete
5	all - registry-service	--	Enabled	Enabled	--	--	streamsmgmr, schemaregistry, rangerlookup	View, Edit, Delete
6	all - schema-group, schema-metadata, s...	--	Enabled	Enabled	--	--	streamsmgmr, kafka, schemaregistry, rangerlookup	View, Edit, Delete

The following table describes the pre-defined access policies for Schema Registry:

Access Policy	Description
all - export-import	<p>Allows users to import and export schemas to/from the Schema Registry service.</p> <p>For example, a user can import a .json file with schemas from a Confluent Kafka topic to Cloudera's Schema Registry.</p>

Access Policy	Description
all - serde	Allows users to store metadata for the format of how data should be read and how it should be written. Users can store JAR files for serializers and deserializers and then map the serdes to the schema.
all - schema-group, schema-metadata	Allows users to access the schema groups and schema metadata.
all - schema-group, schema-metadata, schema-branch	Allows users to access the schema groups, schema metadata, and schema branch.
all - registry-service	Allows users to access the schema registry service. If a user is added to this policy, the user can access all Schema Registry entities.
all - schema-group, schema-metadata, schema-branch, schema-version	Allows users to access the schema groups, schema metadata, schema branch, and schema version.

### Related Information

[Schema Registry Authorization through Ranger Access Policies](#)

[Add the user or group to a pre-defined access policy](#)

[Create a Custom Access Policy](#)

## Add the user or group to a pre-defined access policy

When an authenticated user attempts to view, create, edit, or delete a Schema Registry entity, the system checks whether the user has privileges to perform that action. These privileges are determined by the Ranger access policies that a user is associated with.

### Before you begin

For Ranger policies to work, you must have a user group named schemaregistry. If you use UNIX PAM, the schemaregistry user group must be on the node that hosts Schema Registry.

### About this task

Determine the permissions required by a user or user group and accordingly add the user or group to the appropriate pre-defined access policy.

Each pre-defined access policy controls access to one or more Schema Registry entities.

### Procedure

1. From the Cloudera Manager home page, click the Ranger link.  
The **Ranger** management page appears.

- Click the Ranger Admin Web UI link.

The screenshot shows the Cloudera Manager interface for Cluster 1. The left sidebar contains navigation links: Clusters, Hosts, Diagnostics, Audits, Charts, Replication, Administration, and Private Cloud. The main content area displays the Ranger-1 status, including Health Tests (3 Good), Status Summary (all services in Good Health), and a Charts section for Informational Events. The 'Ranger Admin Web UI' link is highlighted in the top navigation bar.

The **Ranger Log In** page appears.

- Enter your user name and password to log in.  
The **Ranger Service Manager** page appears.

The page is organized by service. Each cluster is listed under its respective service. For example, the Schema Registry clusters in the environment are listed under Schema Registry.

- Select a cluster from the Schema Registry section.  
The **List of Policies** page appears.

The screenshot shows the Ranger Service Manager interface. The top navigation bar includes links for Access Manager, Audit, Security Zone, and Settings. The main content area displays the 'List of Policies : cm\_schema-registry' page. A search bar is present, and a table lists the policies. The table has columns for Policy ID, Policy Name, Policy Labels, Status, Audit Logging, Roles, Groups, Users, and Action.

Policy ID	Policy Name	Policy Labels	Status	Audit Logging	Roles	Groups	Users	Action
1	all - export-import	--	Enabled	Enabled	--	--	streamsmgmr, schemaregistry, rangerlookup, kafka	[Eye] [Edit] [Delete]
2	all - serde	--	Enabled	Enabled	--	--	streamsmgmr, kafka, schemaregistry, rangerlookup	[Eye] [Edit] [Delete]
3	all - schema-group, schema-metadata	--	Enabled	Enabled	--	--	streamsmgmr, kafka, schemaregistry, rangerlookup	[Eye] [Edit] [Delete]
4	all - schema-group, schema-metadata, s...	--	Enabled	Enabled	--	--	streamsmgmr, kafka, schemaregistry, rangerlookup	[Eye] [Edit] [Delete]
5	all - registry-service	--	Enabled	Enabled	--	--	streamsmgmr, schemaregistry, rangerlookup	[Eye] [Edit] [Delete]
6	all - schema-group, schema-metadata, s...	--	Enabled	Enabled	--	--	streamsmgmr, kafka, schemaregistry, rangerlookup	[Eye] [Edit] [Delete]

- Click the ID for a policy.  
The **Edit Policy** page appears.

6. In the Allow Conditions section, add the user or group to the respective Select User or Select Group field.

Allow Conditions :

Select Role	Select Group	Select User	Policy Conditions	Permissions	Delegate Admin	
Select Roles	Select Groups	<input type="checkbox"/> streamsmgr <input checked="" type="checkbox"/> kafka <input checked="" type="checkbox"/> schemaregistry	Add Conditions <input type="button" value="+"/>	<input type="button" value="Create"/> <input checked="" type="button" value="Read"/> <input type="button" value="Update"/> <input type="button" value="Delete"/>	<input checked="" type="checkbox"/>	<input type="button" value="x"/>

7. From the Policy Conditions field, enter the appropriate IP address.  
 8. From the Permissions field, select the appropriate permission.  
 9. Click Save.

## Results

The user now has the rights according to the policy and the permission you assigned to the user. These rights apply to all objects in the entities unless you specified otherwise in the Policy Conditions field.

## Related Information

[Schema Registry Authorization through Ranger Access Policies](#)

[Pre-defined Access Policies for Schema Registry](#)

[Create a Custom Access Policy](#)

## Create a Custom Access Policy

You can create a custom access policy for a specific Schema Registry entity, specify an access type, and add a user or user-group to the policy.

### Before you begin

Determine the following information:

- The schema registry entity that the user needs access to.
- Whether the user requires all objects in the entity or specific objects.
- Whether the user needs read, view, edit, or delete permissions to the entity.
- If there are any IP addresses to include or exclude from the user's access.

### About this task

With a custom policy you can specify the Schema Registry entity and the type of access the user requires.

### Procedure

1. Go to the **Ranger List of Policies** page.

## 2. Click Add New Policy.

The screenshot shows the Ranger Access Manager interface. At the top, there's a navigation bar with 'Ranger', 'Access Manager', 'Audit', 'Security Zone', and 'Settings'. Below this, a breadcrumb trail shows 'Service Manager' > 'cm\_schema\_registry Policies'. The main heading is 'List of Policies : cm\_schema\_registry'. There's a search bar with the placeholder 'Search for your policy...'. To the right of the search bar is an 'Add New Policy' button, which is highlighted with an orange box. Below the search bar is a table listing existing policies.

Policy ID	Policy Name	Policy Labels	Status	Audit Logging	Roles	Groups	Users	Action
3	all - serde	--	Enabled	Enabled	--	--	streamsmgmr kafka schemaregistry	
5	all - schema-group, schema-metadata	--	Enabled	Enabled	--	--	streamsmgmr kafka schemaregistry	
6	all - schema-group, schema-metadata,...	--	Enabled	Enabled	--	--	streamsmgmr kafka schemaregistry	
7	all - registry-service	--	Enabled	Enabled	--	--	streamsmgmr kafka schemaregistry	
8	all - schema-group, schema-metadata,...	--	Enabled	Enabled	--	--	streamsmgmr kafka schemaregistry	

The **Create Policy** page appears.

- Enter a unique name for the policy.
- Optionally, enter a keyword in the Policy Label field to aid in searching for a policy.
- Select a Schema Registry entity. You can choose the Schema Registry service, schema group, or serde. Then, do one of the following tasks:
  - If you want the user to access all the objects in the entity, enter \*.
  - If you want to specify the objects in the entity that a user can access, enter the name of the object in the text field.
- Optionally, enter a description.
- In the Allow Conditions section, add the user or group to the respective Select User or Select Group field.

The screenshot shows the 'Allow Conditions' section of the Ranger interface. It has a 'hide' link on the right. Below the heading, there are several fields: 'Select Role' (with a 'Select Roles' dropdown), 'Select Group' (with a 'Select Groups' dropdown), and 'Select User' (with a list of selected users: 'streamsmgmr' and 'kafka', and a 'schemaregistry' entity). To the right of these are 'Policy Conditions' (with 'Add Conditions' and a '+' button), 'Permissions' (with 'Create', 'Read', 'Update', and 'Delete' buttons), and 'Delegate Admin' (with a checked checkbox and a '-' button).

- Optionally, from the Policy Conditions field, enter the appropriate IP address.
- From the Permissions field, select the appropriate permission.
- Click Save.

## Results

The user now has the rights according to the policy and the permission you assigned to the user.

## Related Information

[Schema Registry Authorization through Ranger Access Policies](#)

[Pre-defined Access Policies for Schema Registry](#)

[Add the user or group to a pre-defined access policy](#)