

Cloudera Runtime 7.1.6

# Configuring Apache Ranger Authentication with UNIX, LDAP, or AD

Date published: 2019-11-01

Date modified:

# 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

<b>Configuring Ranger Authentication with UNIX, LDAP, AD, or PAM.....</b>	<b>4</b>
Configure Ranger authentication for UNIX.....	4
Configure Ranger authentication for AD.....	6
Configure Ranger authentication for LDAP.....	8
Configure Ranger authentication for PAM.....	10
<b>Ranger AD Integration.....</b>	<b>12</b>
Ranger UI authentication.....	16
Ranger UI authorization.....	19
Ranger Usersync.....	20
Ranger user management.....	29
<b>Configure Ranger Usersync for Deleted Users and Groups.....</b>	<b>29</b>

# Configuring Ranger Authentication with UNIX, LDAP, AD, or PAM

This section describes how to configure the authentication method that determines who is allowed to log in to the Ranger web UI. The options are local UNIX, LDAP, AD, or PAM.



**Note:** In CDP Public Cloud, identity management is provided by FreeIPA, and configured using the Management Console. Therefore for CDP Public Cloud you should leave the Admin Authentication Method set to the UNIX authentication settings. For more information on FreeIPA, see [Managing FreeIPA](#) in the [Identify Management](#) documentation.

## Related Information

[Cloudera Management Console](#)

[CDP Cloud Management Console: Managing user access and authorization](#)

[Managing FreeIPA](#)

## Configure Ranger authentication for UNIX

How to configure Ranger to use UNIX for user authentication.

About this task



**Note:** In CDP Public Cloud, identity management is provided by FreeIPA, and configured using the Management Console. Therefore for CDP Public Cloud you should leave the Admin Authentication Method set to the UNIX authentication settings. For more information on FreeIPA, see Managing FreeIPA in the Identify Management documentation.

Procedure

1. In Cloudera Manager, select Ranger, then click the Configuration tab.
2. To display the UNIX authentication settings, type "authentication unix" in the Search box.

The screenshot shows the Cloudera Manager interface for configuring Ranger-1. The search bar contains "authentication unix". The configuration page is divided into filters and settings. The filters section shows counts for SCOPE, CATEGORY, and STATUS. The settings section includes:

- Admin Authentication Method:** Set to UNIX (selected).
- Admin UNIX Auth Remote Login:** Checked.
- Admin UNIX Auth Service Hostname:** Set to {{(RANGER\_USERSYNC\_HOST)}}.
- Unix Auth Service Hostname:** Set to 5151.
- Admin Unix Auth Service Port:** Set to 5151.

3. Configure the following settings for UNIX authentication, then click Save Changes.

Table 1: UNIX Authentication Settings

Configuration Property	Description	Default Value	Example Value	Required
Admin Authentication Method	The Ranger authentication method.	UNIX	UNIX	Yes, to auther
Allow remote Login	Flag to enable/disable remote login. Only used if the Authentication method is UNIX.	TRUE	TRUE	No.

Configuration Property	Description	Default Value	Example Value	Required
ranger.unixauth.service.hostname	The FQDN of the host where the UNIX authentication service is running. Only used if the Authentication method is UNIX. {{RANGER_USERSYNC_HOST}} is a placeholder value that is replaced with the host where Ranger Usersync is installed in the cluster.	localhost	myunixhost.domain.com	Yes, if selected
ranger.unixauth.service.port	The port number where the ranger-usersync module is running the UNIX Authentication Service.	5151	5151	Yes, if selected

### Related Information

[Cloudera Management Console](#)

## Configure Ranger authentication for AD

How to configure Ranger to use Active Directory (AD) for user authentication.

### About this task



**Note:** In CDP Public Cloud, identity management is provided by FreeIPA, and configured using the Management Console. Therefore for CDP Public Cloud you should leave the Admin Authentication Method set to the UNIX authentication settings. For more information on FreeIPA, see [Managing FreeIPA](#) in the Identify Management documentation.

## Procedure

1. Select Cloudera Manager Ranger Configuration, type authentication in Search. Ranger authentication property settings display. You may need to scroll down to see the AD settings.

The screenshot shows the Cloudera Manager interface for configuring Ranger authentication. The left sidebar contains navigation options like Clusters, Hosts, Diagnostics, Audits, Charts, Backup, and Administration. The main content area is titled 'RANGER-1' and shows the 'Configuration' tab for 'authentication'. A search filter 'authentication' is applied. The 'Filters' section on the left shows counts for SCOPE (RANGER-1: 0, Ranger Admin: 19, Ranger Tagsync: 1, Ranger Usersync: 2) and STATUS (Error: 0, Warning: 0, Edited: 2, Non-default: 2, Has Overrides: 0). The main configuration area includes several sections:

- Admin Authentication Method:** Method is set to `ranger.authentication.method` with radio buttons for UNIX, LDAP, **ACTIVE\_DIRECTORY** (selected), PAM, and NONE.
- Admin UNIX Auth Remote Login:** Login is set to `ranger.unixauth.remote.login.enabled` with a checkbox for Ranger Admin Default Group.
- Admin UNIX Auth Service Hostname:** Hostname is set to `ranger.unixauth.service.hostname` with a text input field containing `{{(RANGER_USERSYNC_HOST)}}`. A help icon indicates: 'Host where unix authentication service is running. Only used if Authentication method is UNIX. {{(RANGER\_USERSYNC\_HOST)}} is a placeholder value which will be replaced with the host where Ranger Usersync will be installed in the current cluster.'
- Admin LDAP Auth User DN Pattern:** Pattern is set to `ranger.ldap.user.dn.pattern` with a text input field.
- Admin LDAP Auth User Search Filter:** Search Filter is set to `ranger.ldap.user.searchfilter` with a text input field.
- Admin LDAP Auth Group Search Base:** Search Base is set to `ranger.ldap.group.searchbase` with a text input field.

2. Configure the following settings for AD authentication, then click Save Changes.

Property	Description	Default value	Sample values
Admin Authentication Method	The Ranger authentication method.	UNIX	ACTIVE_DIRECTORY
Admin AD Auth Base DN <code>ranger.ldap.ad.base.dn</code>	The Distinguished Name (DN) of the starting point for directory server searches.	N/A	dc=example,dc=com
Admin AD Auth Bind DN <code>ranger.ldap.ad.bind.dn</code>	The full Distinguished Name (DN), including Common Name (CN) of an LDAP user account that has privileges to search for users.	N/A	cn=adadmin,cn=Users,dc=example,dc=com
Admin AD Auth Bind Password <code>ranger.ldap.ad.bind.password</code>	Password for the bind.dn.	N/A	Secret123!
Admin AD Auth Domain Name <code>ranger.ldap.ad.domain</code>	The domain name of the AD Authentication service.	N/A	dc=example,dc=com

Property	Description	Default value	Sample values
Admin AD Auth Referral ranger.ldap.ad.referral*	See below.	ignore	follow   ignore   throw
Admin AD Auth URL ranger.ldap.ad.url	The AD server URL, for example: ldap://<AD-Servername>Port	N/A	ldap://<AD-Servername>Port
Admin AD Auth User Search Filter ranger.ldap.ad.user.searchfilter	AD user search filter.	N/A	

\* There are three possible values for `ranger.ldap.ad.referral`:

- follow
- throw
- ignore

The recommended setting is: follow.

When searching a directory, the server might return several search results, along with a few continuation references that show where to obtain further results. These results and references might be interleaved at the protocol level.

**When `ranger.ldap.ad.referral` is set to follow:**

The AD service provider processes all of the normal entries first, and then follows the continuation references.

**When `ranger.ldap.ad.referral` is set to throw:**

All of the normal entries are returned in the enumeration first, before the `ReferralException` is thrown.

By contrast, a referral error response is processed immediately when this property is set to follow or throw.

**When `ranger.ldap.ad.referral` is set to ignore:**

The server should return referral entries as ordinary entries (or plain text). This might return partial results for the search. In the case of AD, a `PartialResultException` is returned when referrals are encountered while search results are processed.

### Related Information

[Cloudera Management Console](#)

## Configure Ranger authentication for LDAP

How to configure Ranger to use LDAP for user authentication.

### About this task



**Note:** In CDP Public Cloud, identity management is provided by FreeIPA, and configured using the Management Console. Therefore for CDP Public Cloud you should leave the Admin Authentication Method set to the UNIX authentication settings. For more information on FreeIPA, see [Managing FreeIPA](#) in the Identify Management documentation.

### Procedure

1. In Cloudera Manager, select Ranger, then click the Configuration tab.

- To display the authentication settings, type "authentication" in the Search box. You may need to scroll down to see all of the LDAP settings.

- Configure the following settings for LDAP authentication, then click Save Changes.

Property	Required ?	Description	Default value	Sample values
Admin Authentication Method	Required	The Ranger authentication method.	UNIX	LDAP
Admin LDAP Auth Group Search Base ranger.ldap.group.searchbase	Optional	The LDAP group search base.	N/A	((CN=Hdp_users)(CN=Hdp_admins))
Admin LDAP Auth Group Search Filter ranger.ldap.group.searchfilter	Optional	The LDAP group search filter.	N/A	
Admin LDAP Auth URL ranger.ldap.url	Required	The LDAP server URL	N/A	ldap://localhost:389 or ldaps://localhost:636

Property	Required ?	Description	Default value	Sample values
Admin LDAP Auth Bind User ranger.ldap.bind.dn	Required	Full distinguished name (DN), including common name (CN), of an LDAP user account that has privileges to search for users. This user is used for searching the users. This could be a read-only LDAP user.	N/A	cn=admin,dc=example,dc=com
Admin LDAP Auth Bind User Password ranger.ldap.bind.password	Required	Password for the account that can search for users.	N/A	Secret123!
Admin LDAP Auth User Search Filter ranger.ldap.user.searchfilter	Required	The LDAP user search filter.	N/A	
Admin LDAP Auth Base DN ranger.ldap.base.dn	Required	The Distinguished Name (DN) of the starting point for directory server searches.	N/A	dc=example,dc=com
Admin LDAP Auth Group Role Attribute ranger.ldap.group.roleattribute	Optional	The LDAP group role attribute.	N/A	cn
Admin LDAP Auth Referral ranger.ldap.referral*	Required	See below.	ignore	follow   ignore   throw
Admin LDAP Auth User DN Pattern ranger.ldap.user.dnpattern	Required	The LDAP user DN.	N/A	uid={0},ou=users,dc=xasecure,dc=net

\* There are three possible values for `ranger.ldap.ad.referral`: `follow`, `throw`, and `ignore`. The recommended setting is `follow`.

When searching a directory, the server might return several search results, along with a few continuation references that show where to obtain further results. These results and references might be interleaved at the protocol level.

- When this property is set to `follow`, the AD service provider processes all of the normal entries first, and then follows the continuation references.
- When this property is set to `throw`, all of the normal entries are returned in the enumeration first, before the `ReferralException` is thrown. By contrast, a "referral" error response is processed immediately when this property is set to `follow` or `throw`.
- When this property is set to `ignore`, it indicates that the server should return referral entries as ordinary entries (or plain text). This might return partial results for the search. In the case of AD, a `PartialResultException` is returned when referrals are encountered while search results are processed.

### Related Information

[Cloudera Management Console](#)

## Configure Ranger authentication for PAM

How to configure Ranger to use PAM for user authentication.

## About this task



**Note:** In CDP Public Cloud, identity management is provided by FreeIPA, and configured using the Management Console. Therefore for CDP Public Cloud you should leave the Admin Authentication Method set to the UNIX authentication settings. For more information on FreeIPA, see [Managing FreeIPA](#) in the [Identify Management](#) documentation.

## Procedure

1. In Cloudera Manager, select Ranger, then click the Configuration tab.
2. Under Admin Authentication Method, select PAM, then click Save Changes.

The screenshot shows the Cloudera Manager interface for configuring Ranger. The left sidebar contains navigation options: Clusters, Hosts, Diagnostics, Audits, Charts, Replication, and Administration. The main content area displays various configuration settings for Ranger, including Solr Collection of Ranger Audits (Shards, Maximum Shards, Replicas) and Admin Authentication Method. The Admin Authentication Method is currently set to PAM, with radio buttons for UNIX, LDAP, ACTIVE\_DIRECTORY, PAM (selected), and NONE. Below this, there are settings for Admin UNIX Auth Remote Login, Admin UNIX Auth Service Hostname, Admin LDAP Auth URL, Admin LDAP Auth Bind User, Admin LDAP Auth Bind User Password, Admin LDAP Auth User DN Pattern, and Admin LDAP Auth User Search Filter. A status bar at the bottom indicates '1 Edited Value' and 'Reason for change: Modified Admin Authentication Method'. A blue 'Save Changes (CTRL+S)' button is visible at the bottom right.

3. Create the following two PAM files:

- `/etc/pam.d/ranger-admin` with the following content:

```

#%PAM-1.0
auth sufficient pam_unix.so
auth sufficient pam_sss.so
account sufficient pam_unix.so
account sufficient pam_sss.so

```

- `/etc/pam.d/ranger-remote` with the following content:

```

#%PAM-1.0
auth sufficient pam_unix.so
auth sufficient pam_sss.so
account sufficient pam_unix.so

```

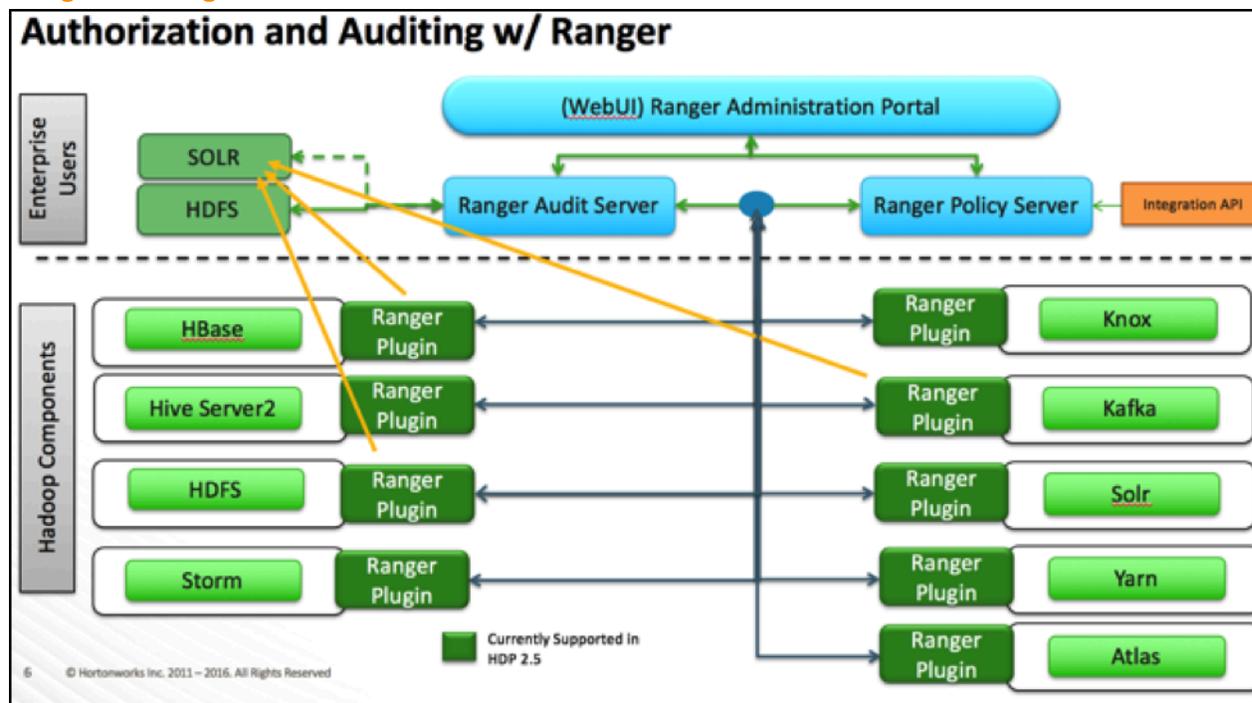
```
account sufficient pam_sss.so
```

4. Confirm that the /etc/shadow file has 444 permissions.
5. Select Actions > Restart to restart Ranger.

## Ranger AD Integration

A conceptual overview of Ranger-AD integration architecture.

### Ranger AD Integration: Architecture Overview



When a Ranger plugin for a component (such as HBase or HDFS) is activated, Ranger is in full control of any access. There is two-way communication between the Ranger plugin and the Ranger (Admin) Policy Server (RPS):

1. **Plugins to RPS:** Ranger plugins regularly call the RPS to see if new policies were defined in the Ranger Administration Portal (RAP). Generally it takes approximately 30 seconds for a policy to be updated.
2. **RPS to components:** The RPS queries the component for meta objects that live on the component to base policies upon (this provides the autocomplete and drop-down list when defining policies).

The first communication channel (Plugin to RPS) is essential for the plugin to function, whereas the second (RPS to components) is optional. It would still be possible to define and enforce policies without the second channel, but you would not have autocomplete during policy definition.

Configuration details on both communication channels are configured in both Cloudera Manager and in the Ranger Administration Portal.

Example for HDFS plugin on a kerberized cluster:

The screenshot shows the Cloudera Manager interface for Cluster 1. The left sidebar contains the Clusters menu, with 'Clusters' selected. The main area displays the configuration for HDFS-1. The configuration parameters are as follows:

Parameter	Value
Superuser Group	supergroup
Kerberos Principal	hdfs
HDFS User to Impersonate	
Hue's Kerberos Principal Short Name	
DataNode Data Transfer Protection	Authentication

The Kerberos principal short name for the HDFS service, "hdfs", is the one that is involved the second communication channel (RPS to components) for getting metadata from HDFS (such as HDFS folders) across. The settings on the HDFS configuration must match those set in Ranger (by selecting Access > Manager > Resource Based Policies, then selecting the Edit icon for the HDFS service):

Select Tag Service:

**Config Properties :**

Username \*

Password \*

Namemode URL \*

Authorization Enabled:

Authentication Type \*

hadoop.security.auth\_to\_local:

dfs.datanode.kerberos.principal:

dfs.namenode.kerberos.principal:

dfs.secondary.namenode.kerberos.principal:

RPC Protection Type:

Common Name for Certificate:

Add New Configurations

Name	Value
tag.download.auth.users	<input type="text" value="hdfs"/>
policy.download.auth.users	<input type="text" value="hdfs"/>

To verify the second communication channel (RPS to components) click Test Connection for the applicable service (as shown above for the HDFS service). A confirmation message appears if the connection works successfully.

To verify if the paramount first communication channel (Plugins to RPS) works, select Audit > Plugins in Ranger:

The screenshot shows the Ranger interface with the 'Plugins' tab selected. A search bar is at the top. Below it, a table lists plugin sync events. The table has columns for Export Date, Service Name, Plugin Id, Plugin IP, Cluster Name, Http Response Code, and Status. All entries show a 200 response code and 'Policies synced to plugin' status.

Export Date ( Eastern Daylight Time )	Service Name	Plugin Id	Plugin IP	Cluster Name	Http Response Code	Status
08/13/2019 11:49:39 AM	cm_hive	hiveServer2@...	10.65.30.5	Cluster 1	200	Policies synced to plugin
08/13/2019 11:49:27 AM	cm_hive	impala@...	10.65.30.5	Cluster 1	200	Policies synced to plugin
08/13/2019 11:49:22 AM	cm_hive	impala@...	10.65.30.5	Cluster 1	200	Policies synced to plugin
08/13/2019 11:49:17 AM	cm_hive	impala@...	10.65.49.144	Cluster 1	200	Policies synced to plugin
08/13/2019 11:49:17 AM	cm_hive	impala@...	10.65.50.67	Cluster 1	200	Policies synced to plugin
08/13/2019 11:46:39 AM	cm_hive	hiveServer2@...	10.65.30.5	Cluster 1	200	Policies synced to plugin
08/13/2019 11:46:27 AM	cm_hive	impala@...	10.65.30.5	Cluster 1	200	Policies synced to plugin
08/13/2019 11:46:22 AM	cm_hive	impala@...	10.65.30.5	Cluster 1	200	Policies synced to plugin
08/13/2019 11:46:17 AM	cm_hive	impala@...	10.65.49.144	Cluster 1	200	Policies synced to plugin
08/13/2019 11:46:17 AM	cm_hive	impala@...	10.65.50.67	Cluster 1	200	Policies synced to plugin
08/05/2019 02:51:20 PM	cm_atlas	atlas@...	10.65.30.5	Cluster 1	200	Policies synced to plugin

### Ranger AD Integration: Ranger Audit

Ranger plugins furthermore send their audit event (whether access was granted or not and based on which policy) directly to the configured sink for audits, which can be HDFS, Solr or both. This is indicated by the yellow arrows in the architectural graph.

The audit access tab on the RAP (Audit > Access) is only populated if Solr is used as the sink.

The screenshot shows the Ranger interface with the 'Audit' > 'Access' tab selected. A search bar with 'START DATE: 08/14/2019' is at the top. Below it, a table lists audit events. The table has columns for Policy ID, Policy Version, Event Time, Application, User, Service Name / Type, Resource Name / Type, Access Type, Result, Access Enforcer, and Agent Host Name. All results are 'Allowed'.

Policy ID	Policy Version	Event Time	Application	User	Service Name / Type	Resource Name / Type	Access Type	Result	Access Enforcer	Agent Host Name
5	1	08/14/2019 03:15:02 PM	hbaseRegional	atlas	cm_hbase hbase	atlas_janus/m column-family	get	Allowed	ranger-acl	...
15	1	08/14/2019 03:15:02 PM	kafka	kafka	cm_kafka kafka	kafka-cluster cluster	kafka_admin	Allowed	ranger-acl	...
15	1	08/14/2019 03:15:02 PM	kafka	kafka	cm_kafka kafka	kafka-cluster cluster	kafka_admin	Allowed	ranger-acl	...
15	1	08/14/2019 03:15:00 PM	kafka	kafka	cm_kafka kafka	kafka-cluster cluster	kafka_admin	Allowed	ranger-acl	...
20	1	08/14/2019 03:15:00 PM	kafka	atlas	cm_kafka kafka	ATLAS_SPARK_... topic	consume	Allowed	ranger-acl	...
15	1	08/14/2019 03:15:00 PM	kafka	kafka	cm_kafka kafka	kafka-cluster cluster	kafka_admin	Allowed	ranger-acl	...
18	1	08/14/2019 03:15:00 PM	kafka	atlas	cm_kafka kafka	ATLAS_HOOK topic	consume	Allowed	ranger-acl	...
15	1	08/14/2019 03:14:58 PM	kafka	kafka	cm_kafka kafka	kafka-cluster cluster	kafka_admin	Allowed	ranger-acl	...
15	1	08/14/2019 03:14:58 PM	kafka	kafka	cm_kafka kafka	kafka-cluster cluster	kafka_admin	Allowed	ranger-acl	...
5	1	08/14/2019 03:14:57 PM	hbaseRegional	atlas	cm_hbase hbase	atlas_janus/m column-family	get	Allowed	ranger-acl	...

This screen points out an important Ranger feature. When the plugin is enabled AND no specific policy is in place for access to some object, the plugin will fall back to enforcing the standard component-level Access Control Lists (ACLs). For HDFS that would be the user : rwx / group : rwx / other : rwx ACLs on folders and files.

Once this defaulting to component ACLs happens, the audit events list a " - " in the Policy ID column instead of a policy number. If a Ranger policy was in control of allowing/denying access, the policy number is shown.

### Ranger AD Integration: Overview

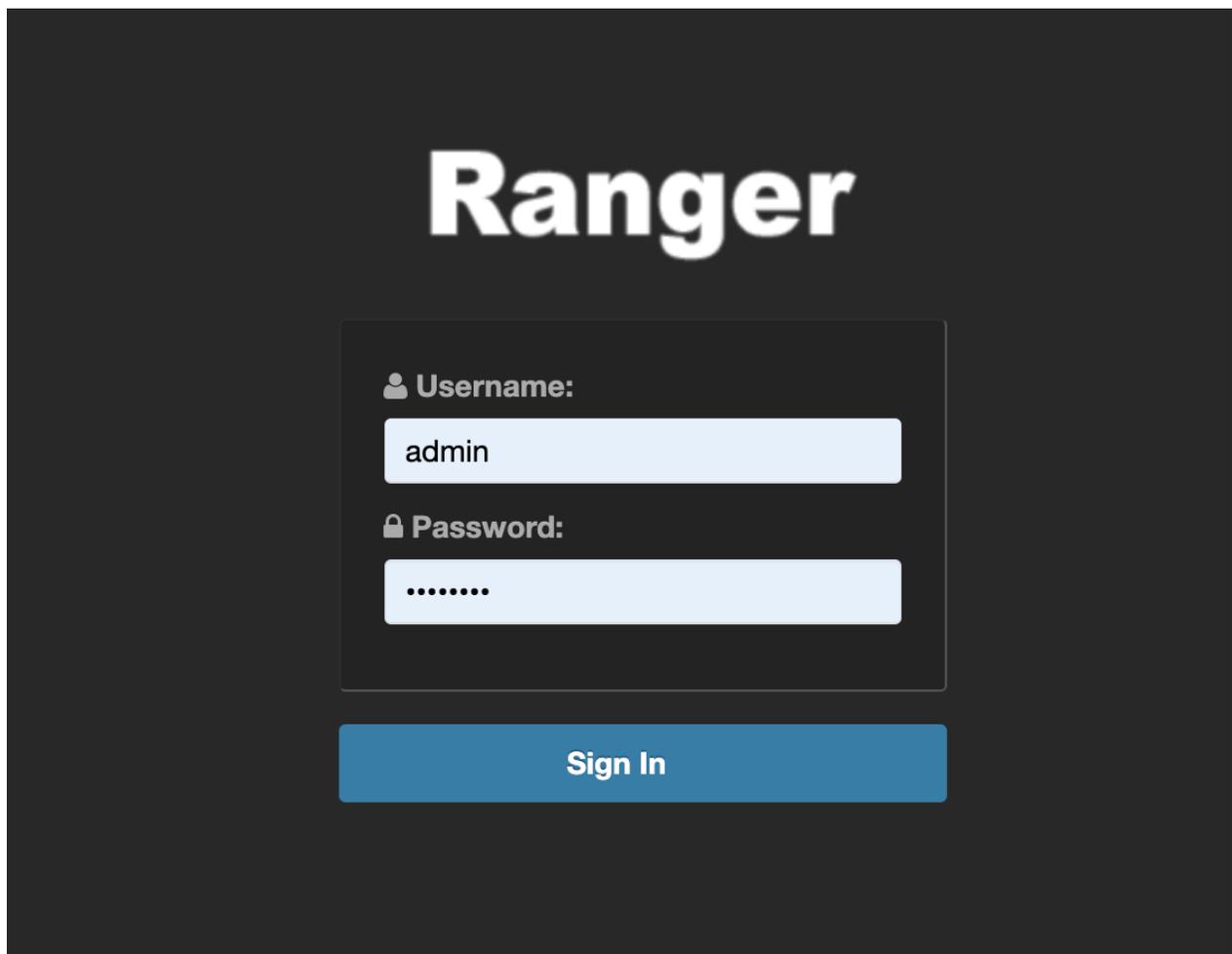
Rangers AD Integration has 2 levels:

1. Ranger UI authentication (which users can log in to Ranger itself).
2. Ranger user/group sync (which users/groups to define policies for)

## Ranger UI authentication

Reference information on Ranger UI authentication, when configuring Ranger AD integration.

This is an extra AD level filter option on top of Kerberos authentication that maps to:



For AD there are two options for defining who can access the Ranger UI: LDAP or ACTIVE\_DIRECTORY. There is not a huge amount of difference between them, but they are separate sets of properties.

ACTIVE\_DIRECTORY

In Cloudera Manager, select Ranger, then click the Configuration tab. To display the authentication settings, type "authentication" in the Search box. You may need to scroll down to see the AD settings.

The screenshot shows the Cloudera Manager interface for configuring Ranger-1. The search box is set to "authentication". The left sidebar lists navigation options: Clusters, Hosts, Diagnostics, Audits, Charts, Backup, and Administration. The main content area displays the configuration for Ranger-1, with the "Configuration" tab selected. The search results show a list of filters and a detailed view of the authentication settings.

**Filters**

- SCOPE**
  - RANGER-1 (Service-Wide) 0
  - Ranger Admin 19
  - Ranger Tagsync 1
  - Ranger Usersync 2
- CATEGORY**
  - Advanced 0
  - Logs 0
  - Main 21
  - Monitoring 0
  - Performance 0
  - Ports and Addresses 1
  - Resource Management 0
  - Security 0
  - Stacks Collection 0
- STATUS**
  - Error 0
  - Warning 0
  - Edited 2
  - Non-default 2
  - Has Overrides 0

**Authentication Settings:**

- Admin Authentication Method:** Ranger Admin Default Group. Options: UNIX, LDAP, ACTIVE\_DIRECTORY (selected), PAM, NONE.
- Admin UNIX Auth Remote Login:** Ranger Admin Default Group.
- Admin UNIX Auth Service Hostname:** {{RANGER\_USERSYNC\_HOST}}. Description: Host where unix authentication service is running. Only used if Authentication method is UNIX. {{RANGER\_USERSYNC\_HOST}} is a placeholder value which will be replaced with the host where Ranger Usersync will be installed in the current cluster.
- Admin LDAP Auth User DN Pattern:** Ranger Admin Default Group.
- Admin LDAP Auth User Search Filter:** Ranger Admin Default Group.
- Admin LDAP Auth Group Search Base:** Ranger Admin Default Group.

The `ranger.ldap.ad.base.dn` property determines the base of any search, so users not on this OU tree path can not be authenticated.

The `ranger.ldap.ad.user.searchfilter` property is a dynamic filter that maps the user name in the Ranger web UI login screen to `sAMAccountName`. For example, the AD `sAMAccountName` property has example values like `k.reshi` and `d.alora` so make sure to enter a matching value for 'Username' in the logon dialogue.

## LDAP

The LDAP properties allow for more fine tuning.

In Cloudera Manager, select Ranger, then click the Configuration tab. To display the authentication settings, type "authentication" in the Search box. You may need to scroll down to see all of the LDAP settings.

The screenshot shows the Cloudera Manager interface for configuring Ranger-1 authentication. The left sidebar contains navigation options like Clusters, Hosts, Diagnostics, Audits, Charts, Backup, and Administration. The main content area is titled 'Cluster 1' and 'RANGER-1'. The 'authentication' configuration is selected, showing a list of filters (SCOPE, CATEGORY, STATUS) and a detailed configuration panel. The configuration panel includes settings for Admin Authentication Method (LDAP), Admin UNIX Auth Remote Login, Admin UNIX Auth Service Hostname, Admin LDAP Auth User DN Pattern, Admin LDAP Auth User Search Filter, Admin LDAP Auth Group Search Base, and Admin LDAP Auth Group Search Filter.

There is one catch: the `ranger.ldap.user.dnpattern` is evaluated first. Consider the following example value:

```
CN={0},OU=London,OU=Company,OU=User    Accounts,OU=CorpUsers,DC=field,DC=hortonworks,DC=com
```

This would work, but has two side effects:

- Users would have to log on with their 'long username' (like 'Kvothe Reshi / Denna Alora'), which would also mean that policies would have to be updated using that long name instead of the `k.reshi` short name variant.
- Traversing AD by DN patterns does not allow for applying group filters at all. In the syntax above, only users directly in `OU=London` would be able to log on.

This adverse behavior can be avoided by intentionally putting a DN pattern (`DC=intentionally,DC=wrong`) in the `ranger.ldap.user.dnpattern` property, AND a valid filter in User Search Filter:

```
(&(objectclass=user)(memberOf=CN=Hdp_admins,OU=Company,OU=User    Accounts,OU=CorpUsers,DC=field,DC=hortonworks,DC=com)(sAMAccountName={0}))
```

This works because the filter is only applied after the DN pattern query on AD does not return anything. If it does, the User Search Filter is not applied.

Ranger has a very simple approach to the internal user list that is kept in a relational schema. This list contains all users that were synced with AD ever, and all those users can potentially log in to the Ranger UI. But only Admin users can really do any policy-related things in the Ranger UI (see next section).

Be aware that all of this is only about authentication to Ranger. Someone from the ‘Hdp\_admins’ group would still not have a Ranger admin role.

**Related Information**

[Configure Ranger authentication for LDAP](#)

## Ranger UI authorization

Reference information on Ranger UI authorization, when configuring Ranger AD integration.

To configure the users, groups, and roles that can access the Ranger portal or its services, select Settings > Users/Groups/Roles in the top menu.

The screenshot shows the Ranger UI interface for managing users. The top navigation bar includes 'Ranger', 'Access Manager', 'Audit', 'Security Zone', 'Settings', and a user profile 'admin'. The 'Settings' menu is open, showing 'Users/Groups/Roles' and 'Permissions'. Below the navigation, there are tabs for 'Users', 'Groups', and 'Roles'. The main content area is titled 'User List' and features a search bar and buttons for 'Add New User', 'Set Visibility', and a trash icon. A table lists the following users:

<input type="checkbox"/>	User Name	Email Address	Role	User Source	Groups	Visibility
<input type="checkbox"/>	admin		Admin	Internal	--	Visible
<input type="checkbox"/>	rangerusersync		Admin	Internal	--	Visible
<input type="checkbox"/>	rangertagsync		Admin	Internal	--	Visible
<input type="checkbox"/>	hive		User	External	hive	Visible
<input type="checkbox"/>	cloudera-scm		User	External	wheel cloudera-scm	Visible
<input type="checkbox"/>	https		User	External	https	Visible
<input type="checkbox"/>	superset		User	External	superset	Visible
<input type="checkbox"/>	atlas		User	External	hadoop atlas	Visible
<input type="checkbox"/>	ranger		User	External	hadoop ranger	Visible
<input type="checkbox"/>	kudu		User	External	kudu	Visible
<input type="checkbox"/>	kms		User	External	kms	Visible
<input type="checkbox"/>	accumulo		User	External	accumulo	Visible
<input type="checkbox"/>	polkitd		User	External	polkitd	Visible
<input type="checkbox"/>	nfsnobody		User	External	nfsnobody	Visible
<input type="checkbox"/>	spark		User	External	spark	Visible
<input type="checkbox"/>	flume		User	External	flume	Visible
<input type="checkbox"/>	solr		User	External	solr	Visible
<input type="checkbox"/>	jenkins		User	External	jenkins	Visible

A user can be a User, Admin, or Auditor:

Only users with the Admin role can edit Ranger policies.

## Ranger Usersync

How to configure Ranger Usersync to sync users and groups from AD/LDAP

### Overview

The Ranger usersync service syncs users, groups, and group memberships from various sources, such as Unix, File, or AD/LDAP into Ranger. Ranger usersync provides a set of rich and flexible configuration properties to sync users, groups, and group memberships from AD/LDAP supporting a wide variety of use cases.

As a Ranger administrator, you will work with users and groups to configure policies in Ranger and administer access to the Ranger UI. You will use group memberships only to administer access to the Ranger UI. You must first understand the specific use-case before syncing users, groups, and group memberships from AD/LDAP. For example, if you want to configure only group-level policies, then you must sync groups to Ranger, but syncing users and group memberships to Ranger is not required.

Determining the users and groups to sync to Ranger:

Typically, you must complete a three-step process to define the complete set of users and groups that you will sync to Ranger:

1. Define the customer use-case.

3 common use cases:

- A customer Admin or Data Admin wants to configure only group-level policies and restrict access to the Ranger UI to only a few users.
- A customer's Admin or Data Admin wants to configure only group-level policies and restrict access to the Ranger UI to only members of a group.
- A customer's Admin or Data Admin wants to configure mostly group-level policies and a few user-level policies.

2. Define all relevant sync source details. For every use-case, at least four key questions must be answered:

- What groups will sync to Ranger?
- Which organizational units (OUs) in AD/LDAP contain these groups?
- What users will sync to Ranger?
- Which organizational units (OUs) in AD/LDAP contain these users?

3. Configure Usersync properties.

This topic describes an example set of Usersync configuration properties and values, based on a simple use-case and example AD source repository.

Example Use Case:

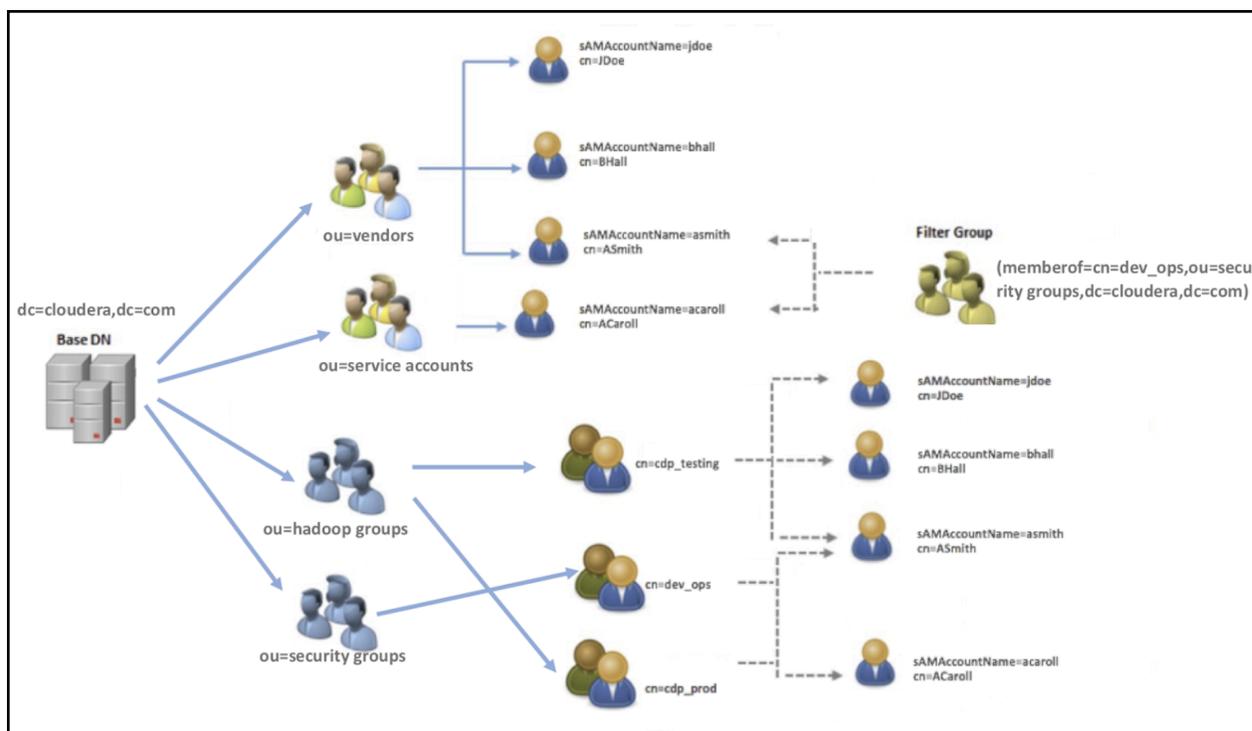
First, consider the following use-case, in order to better understand how to configure Usersync properties:

A customer's Admin or Data Admin wants to configure only group-level policies and restrict access to the Ranger UI to only members of a group.

Example AD environment:

Configuring Ranger Usersync with AD/LDAP depends highly on the customer environment. You must understand the organization of users and groups in the customer environment. This illustration shows users and groups organized in an Active Directory environment.

**Figure 1: Example Active Directory Structure**



Answering the key user and group questions, based on the example AD structure:

In this example, the customer wants to configure group-level policies for groups `cdp_testing` and `cdp_prod` and wants to provide admin access to the Ranger UI only for users in the `dev_ops` group.

Based on the example Active Directory structure, answers to the four key user/group questions are:

**Q1: What groups will be synced to Ranger?**

A1: `cdp_testing`, `cdp_prod`, and `dev_ops`

**Q2: What OUs contain these groups in AD?**

A2: `hadoop groups` and `security groups`

**Q3: What users will be synced to Ranger?**

A3: asmith and acaroll (these users are dev\_ops group members)

**Q4: What OUs contain these users in AD?**

A4: vendors and service accounts

To find the specific answers to these questions in a particular environment, use a tool such as Ldapsearch, as shown in the following examples.

- Example: Ldapsearch command to search a particular group cdp\_testing and determine what attributes are available for the group.

**Figure 2: Using Ldapsearch to find a specific group**

```
ldapsearch -x -LLL -h 10.10.10.10:389 -D 'cn=administrator,CN=Users,dc=cloudera,dc=com' -W
-b 'ou=Hadoop Groups,dc=cloudera,dc=com' 'cn=cdp_testing'
Enter LDAP Password:
dn: CN=cdp_testing,ou=Hadoop Groups,dc=cloudera,dc=com
objectClass: top
objectClass: group
cn: cdp_testing
member: CN=ASmith,ou=Hadoop Users,dc=cloudera,dc=com
member: CN=BHall,ou=Hadoop Users,dc=cloudera,dc=com
member: CN=JDoe,ou=Hadoop Users,dc=cloudera,dc=com
distinguishedName: CN=cdp_testing,ou=Hadoop Groups,dc=cloudera,dc=com
instanceType: 4
name: cdp_testing
sAMAccountName: cdp_testing
```

Above output shows all the available attributes for cn=cdp\_testing. The highlighted attributes are those of interest for usersync configuration. In this case, cdp\_testing has three “member” attributes: ASmith, BHall, and JDoe.

- Example: Ldapsearch command to search a particular user ASmith and determine what attributes are available for the user.

**Figure 3: Using Ldapsearch to find a specific user**

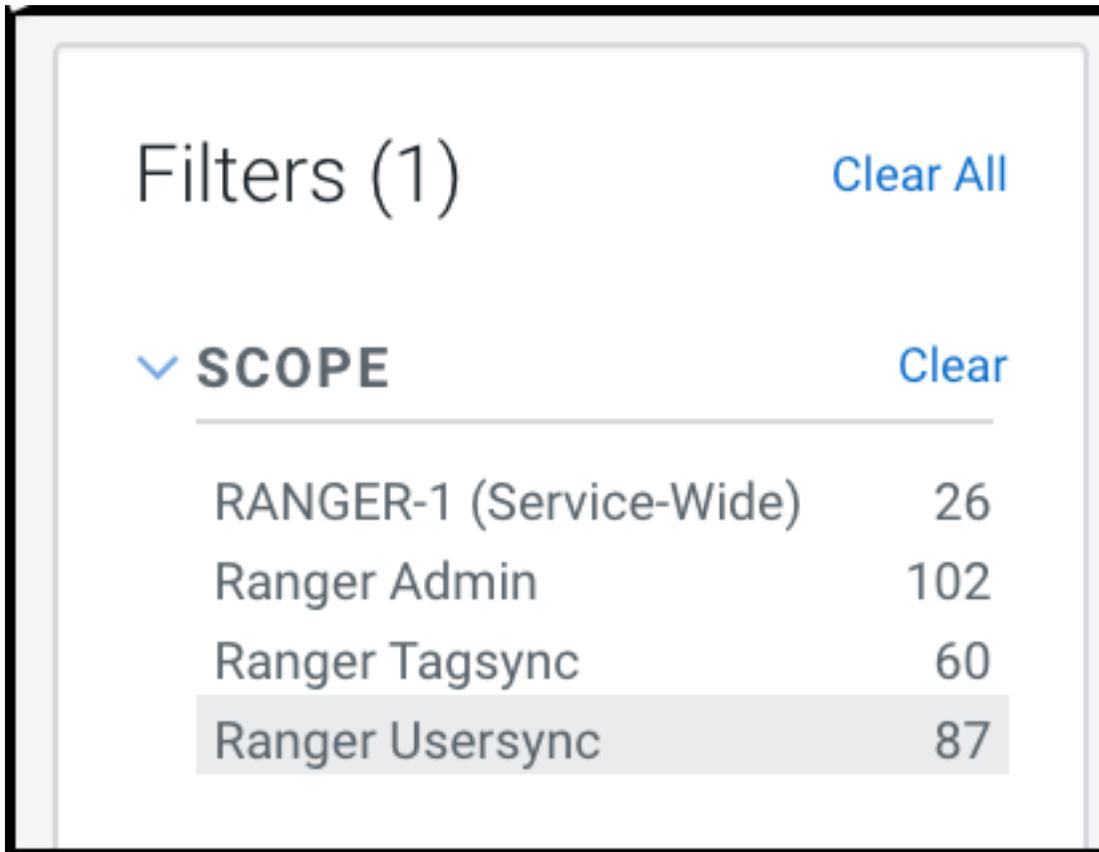
```
ldapsearch -x -LLL -h 10.10.10.10:389 -D 'cn=administrator,CN=Users,dc=cloudera,dc=com'
-W -b 'ou=Hadoop Users,dc=cloudera,dc=com' 'samaccountname=ASmith
Enter LDAP Password:
dn: CN=ASmith,ou=Hadoop Users,dc=cloudera,dc=com
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: user
cn: ASmith
sn: Smith
givenName: Andy
distinguishedName: CN=ASmith,ou=Hadoop Users,dc=cloudera,dc=com
instanceType: 4
memberOf: CN=cdp_testing,ou=Hadoop Groups,dc=cloudera,dc=com
memberOf: CN=dev_ops,ou=Hadoop Groups,dc=cloudera,dc=com
memberOf: CN=cdp_prod,ou=Hadoop Groups,dc=cloudera,dc=com
primaryGroupID: 513
logonCount: 0
sAMAccountName: ASmith
```

Above output shows all the available attributes for a user. The highlighted attributes are those of interest for usersync configuration. In this case, ASmith is a “memberof” 3 groups - cdp\_testing, dev\_ops, and cdp\_prod.

How to configure Usersync, based on the illustrated AD environment example:

In Cloudera Manager Ranger Configuration select the Ranger Usersync filter scope.

**Figure 4: Filtering the Ranger Configuration Properties for Usersync**



The screenshot shows a filter menu with the following structure:

- Filters (1) [Clear All](#)
- ▼ **SCOPE** [Clear](#)
- RANGER-1 (Service-Wide) 26
- Ranger Admin 102
- Ranger Tagsync 60
- Ranger Usersync 87**

Scope	Count
RANGER-1 (Service-Wide)	26
Ranger Admin	102
Ranger Tagsync	60
<b>Ranger Usersync</b>	<b>87</b>

Filtering narrows the list to 87 configuration properties specific to Usersync.

- To define the common configuration properties that control LDAP URL and bind credentials, scroll to Source for Syncing Users and Groups, then define the configurations properties appropriate for the environment. Configurations shown here match the Example AD environment.

**Figure 5: Ranger Usersync common configuration settings**

<b>Source for Syncing User and Groups</b> ranger.usersync.source.impl.class <a href="#">ranger.usersync.source.impl.class</a>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ <input type="radio"/> org.apache.ranger.unixusersync.process.UnixUserGroupBuilder <input type="radio"/> org.apache.ranger.unixusersync.process.FileSourceUserGroupBuilder <input checked="" type="radio"/> org.apache.ranger.ldapusersync.process.LdapUserGroupBuilder
<b>Usersync LDAP/AD URL</b> ranger.usersync.ldap.url <a href="#">ranger.usersync.ldap.url</a>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ <input type="text" value="ldap://ad01.cloudera.com:389"/>
<b>Usersync Bind User</b> ranger.usersync.ldap.binddn <a href="#">ranger.usersync.ldap.binddn</a>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ <input type="text" value="cn=administrator,ou=service accounts,dc=cloudera,dc=com"/>
<b>Usersync Bind User Password</b> ranger.usersync.ldap.ldapbindpassword <a href="#">ranger_usersync_ldap_ldapbindpassword</a>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ <input type="password" value="....."/>
<b>Usersync Incremental Sync</b> ranger.usersync.ldap.deltasync <a href="#">ranger.usersync.ldap.deltasync</a>	<input checked="" type="checkbox"/> Ranger Usersync Default Group ⓘ

Bind credentials are for the user to query Ldap service for users and groups. Bind credentials contain two configuration properties:

- Usersync Bind User (or bind dn) - specify the username as complete DN (Distinguished Name)
- Usersync Bind User Password

- To define the required configuration properties that control group synchronization from AD, scroll to Usersync Enable User Search, then define the configurations properties appropriate for the environment. Configurations shown here match the Example AD environment.

**Figure 6: Ranger Usersync group configuration settings**

<b>Usersync Groupname Case Conversion</b> <small>ranger.usersync.ldap.groupname.caseconversion</small> <small><a href="#">ranger.usersync.ldap.groupname.caseconversion</a></small>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ <input type="radio"/> none <input checked="" type="radio"/> lower <input type="radio"/> upper
<b>Usersync Enable User Search</b> <small>ranger.usersync.user.searchenabled</small> <small><a href="#">ranger.usersync.user.searchenabled</a></small>	<input checked="" type="checkbox"/> Ranger Usersync Default Group ⓘ
<b>Usersync Group Search Base</b> <small>ranger.usersync.group.searchbase</small> <small><a href="#">ranger.usersync.group.searchbase</a></small>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ ou=hadoop groups,dc=cloudera,dc=com,ou=security groups,dc=cloudera,dc=com
<b>Usersync Group Search Scope</b> <small>ranger.usersync.group.searchscope</small> <small><a href="#">ranger.usersync.group.searchscope</a></small>	Ranger Usersync Default Group <input checked="" type="radio"/> sub <input type="radio"/> base <input type="radio"/> one
<b>Usersync Group Object Class</b> <small>ranger.usersync.group.objectclass</small> <small><a href="#">ranger.usersync.group.objectclass</a></small>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ group
<b>Usersync Group Search Filter</b> <small>ranger.usersync.group.searchfilter</small> <small><a href="#">ranger.usersync.group.searchfilter</a></small>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ ((cn=cdp*)(cn=dev_ops))
<b>Usersync Group Name Attribute</b> <small>ranger.usersync.group.nameattribute</small> <small><a href="#">ranger.usersync.group.nameattribute</a></small>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ cn
<b>Usersync Group Member Attribute</b> <small>ranger.usersync.group.memberattributename</small> <small><a href="#">ranger.usersync.group.memberattributename</a></small>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ member

A few specific points to consider about group config settings:

- `ranger.usersync.ldap.groupname.caseconversion` - Used for converting the case of the groupname. Three possible options are:
  - None - Group names are synced to ranger as is from AD/LDAP. This is the default setting.
  - Lower - All the group names are converted to lowercase while syncing to ranger. This is the recommended setting.
  - Upper - All the group names are converted to uppercase while syncing to ranger



**Note:** Policy authorization is case sensitive. Therefore, usernames and groups names synced to ranger must match the exact case of the users and groups resolved by the services such as hdfs, hive, hbase, etc. For example, consider `dev_ops` (all in lower case). Ranger does not treat this as the same value as `Dev_Ops` which may have been synced from AD and applied to some policies.

`ranger.usersync.group.searchbase` - Used to search a particular OU in AD for groups. Multiple OUs can be specified with ; separated. For example, the example AD shows two OUs that must be searched for groups:

- `ou=hadoop groups,dc=cloudera,dc=com` (complete DN for ou=hadoop groups)
- `ou=security groups,dc=cloudera,dc=com` (complete DN for ou=security groups)

- `ranger.usersync.group.searchfilter` - In this example, since only 3 groups exist in hadoop groups OU and security groups OU and since all 3 require sync to Ranger, you can specify the filter as `cn=*` . The value for this property follows standard ldap search query filter format.
  -  **Note:** Later, if a new group is added in AD under these OUs and if the customer wants those groups to be sync'd to ranger, no configuration change to usersync is required.
- `ranger.usersync.user.searchenabled` - In this example, since the customer wants to sync users from `dev_ops` groups to provide admin access to Ranger UI, this property is set to `true` .

- To define the required configuration properties that control user synchronization from AD, scroll to Usersync User Search Base, then define the configurations properties appropriate for the environment. Configurations shown here match the Example AD environment.

**Figure 7: Ranger Usersync user configuration settings**

<b>Usersync User Search Base</b> ranger.usersync.ldap.user.searchbase <a href="#">ranger.usersync.ldap.user.searchbase</a>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ <input type="text" value="ou=vendors,dc=cloudera,dc=com,ou=service accounts,dc=cloudera,dc=com"/>
<b>Usersync User Search Scope</b> ranger.usersync.ldap.user.searchscope <a href="#">ranger.usersync.ldap.user.searchscope</a>	Ranger Usersync Default Group ⓘ <input checked="" type="radio"/> sub <input type="radio"/> base <input type="radio"/> one
<b>Usersync User Object Class</b> ranger.usersync.ldap.user.objectclass <a href="#">ranger.usersync.ldap.user.objectclass</a>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ <input type="text" value="user"/>
<b>Usersync User Search Filter</b> ranger.usersync.ldap.user.searchfilter <a href="#">ranger.usersync.ldap.user.searchfilter</a>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ <input type="text" value="(memberof=cn=dev_ops,ou=security groups,dc=cloudera,dc=com)"/>
<b>Usersync User Name Attribute</b> ranger.usersync.ldap.user.nameattribute <a href="#">ranger.usersync.ldap.user.nameattribute</a>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ <input type="text" value="sameaccountname"/>
<b>Usersync Referral</b> ranger.usersync.ldap.referral <a href="#">ranger.usersync.ldap.referral</a>	Ranger Usersync Default Group ⓘ <input checked="" type="radio"/> ignore <input type="radio"/> follow <input type="radio"/> throw
<b>Usersync Username Case Conversion</b> ranger.usersync.ldap.username.caseconversion <a href="#">ranger.usersync.ldap.username.caseconversion</a>	Ranger Usersync Default Group <a href="#">Undo</a> ⓘ <input type="radio"/> none <input checked="" type="radio"/> lower <input type="radio"/> upper

A few specific points to consider about user config settings:

- ranger.usersync.ldap.user.searchbase - This configuration is used to search a particular location in AD for users. Specify multiple OUs with ; separated.
  -  **Note:** If users are distributed across several OUs, specifying a base directory, for example, dc=cloudera,dc=com might be convenient and is highly recommended to restrict the search with proper filters.
- ranger.usersync.ldap.user.searchfilter - In this example, since the customer wants to sync only the users that belong to dev\_ops, the value for this property is (memberof=cn=dev\_ops,ou=security groups,dc=cloudera,dc=com) .
  -  **Note:** Wildcards are not supported only when the memberof attribute is used for searching. If you use attributes such as cn or samaccountname for filtering, you can specify wildcards. For example, ((cn=asm\*)(samaccountname=acar\*))
- ranger.usersync.ldap.username.caseconversion - Used for converting the case of the username. Three possible options are:
  - None - Usernames are synced to ranger as is from AD/LDAP. This is the default setting.
  - Lower - All the usernames are converted to lowercase while syncing to ranger. This is the recommended setting.
  - Upper - All the usernames are converted to uppercase while syncing to ranger



**Note:** Policy authorization is case sensitive. Therefore, usernames and groups names synced to ranger must match the exact case of the users and groups resolved by the services such as hdfs, hive, hbase, etc. For example, consider asmith (all in lower case). Ranger does not treat this as the same value as ASmith which may have been synced from AD and applied to some policies.

## Ranger user management

Reference information on Ranger user management, when configuring Ranger AD integration.

To delete a user, select the check box for the user in the User Name list, then click the red Delete button. Ranger removes the user from all policies.

The screenshot shows the Ranger web interface with the 'Users' tab selected. The 'User List' table contains the following data:

	User Name	Email Address	Role	User Source	Groups	Visibility
<input type="checkbox"/>	hdfs		User	External	hadoop hdfs	Visible
<input type="checkbox"/>	rangerlookup		User	External	--	Visible
<input type="checkbox"/>	livy		User	External	livy	Visible
<input type="checkbox"/>	chrony		User	External	chrony	Visible
<input type="checkbox"/>	druid		User	External	hadoop druid	Visible
<input type="checkbox"/>	kafka		User	External	kafka	Visible
<input type="checkbox"/>	knoxui		User	External	knoxui	Visible
<input type="checkbox"/>	yarn		User	External	hadoop yarn	Visible
<input type="checkbox"/>	hue		User	External	hue	Visible
<input type="checkbox"/>	sqoop		User	External	sqoop	Visible
<input type="checkbox"/>	centos		User	External	systemd-journal wheel adm centos	Visible
<input type="checkbox"/>	storm		User	External	--	Visible
<input type="checkbox"/>	knox		User	External	hadoop knox	Visible
<input type="checkbox"/>	mapred		User	External	hadoop mapred	Visible
<input type="checkbox"/>	nifi		User	External	--	Visible
<input type="checkbox"/>	tez		User	External	tez	Visible
<input type="checkbox"/>	auditor1		Auditor	Internal	--	Visible
<input type="checkbox"/>	new-user1		Admin	Internal	--	Visible
<input checked="" type="checkbox"/>	asmith		Admin	Internal	public	Visible

The 'asmith' row is highlighted with a red box, and the 'Delete' button (a red trash icon) is also highlighted with a red box. The interface includes a search bar, 'Add New User' and 'Set Visibility' buttons, and a pagination control showing page 2 of 2.

## Configure Ranger Usersync for Deleted Users and Groups

How to configure Ranger Usersync for users and groups that have been deleted from the sync source.

### About this task

You can configure Ranger Usersync to update Ranger when users and groups have been deleted from the sync source (UNIX, LDAP, AD or PAM). This ensures that users and groups – and their associated access permissions – do not remain in Ranger when they are deleted from sync source.

## Procedure

1. In Cloudera Manager, select Ranger > Configuration, then use the Search box to search for Ranger Usersync Advanced Configuration Snippet (Safety Valve) for conf/ranger-ugsync-site.xml. Use the Add (+) icons to add the following properties, then click Save Changes.

Name	Value	Description
ranger.usersync.deletes.enabled	true	Enables deleted users and groups synchronization. The default setting is false (disabled).
ranger.usersync.deletes.frequency	10	Sets the frequency of delete synchronization. The default setting is 10, or once every 10 Usersync cycles. Delete synchronization consumes cluster resources, so a lower (more frequent) setting may affect performance.

Cluster 1

RANGER-1

Jun 4, 8:57 PM UTC

Status Instances **Configuration** Commands Charts Library Audits Ranger Admin Web UI Quick Links

Q ranger-ugsync-site.xml Filters (1) Role Groups History & Rollback

Filters (1) Clear All

SCOPE Clear

- RANGER-1 (Service-Wide) 0
- Ranger Admin 0
- Ranger Tagsync 0
- Ranger Usersync 1

CATEGORY

- Advanced 1
- Database 0
- Logs 0
- Main 0
- Monitoring 0
- Performance 0
- Ports and Addresses 0
- Resource Management 0
- Security 0
- Stacks Collection 0

STATUS

- Error 0
- Warning 0
- Edited 1
- Non-Default 1
- Include Overrides 0

Ranger Usersync Advanced Configuration Snippet (Safety Valve) for conf/ranger-ugsync-site.xml

Ranger Usersync Default Group Undo

View as XML

Name ranger.usersync.deletes.enabled

Value true

Description

Final

Name ranger.usersync.deletes.frequency

Value 10

Description

Final

1 - 1 of 1

1 Edited Value Reason for change: Modified Ranger Usersync Advanced Configuration Snippet (Safety Valve) for c Save Changes(CTRL+S)

2. Click the Ranger Restart icon.

The screenshot shows the Cloudera Manager interface for a cluster named 'RANGER-1'. The 'Actions' menu is open, and the 'Restart' icon (a circular arrow) is highlighted with a red box. A tooltip above the icon reads 'Stale Configuration: Restart Command needed'. The main configuration area shows a search for 'ranger-ugsync-site.xml' with one filter applied. The configuration details for 'Ranger Usersync Default Group' are displayed, showing two configuration items:

- Item 1:** Name: `ranger.usersync.deletes.enabled`, Value: `true`. Description: `conf/ranger-ugsync-site.xml_role_safety_valve`. There is a 'Final' checkbox which is unchecked.
- Item 2:** Name: `ranger.usersync.deletes.frequen`, Value: `10`. Description: (empty). There is a 'Final' checkbox which is unchecked.

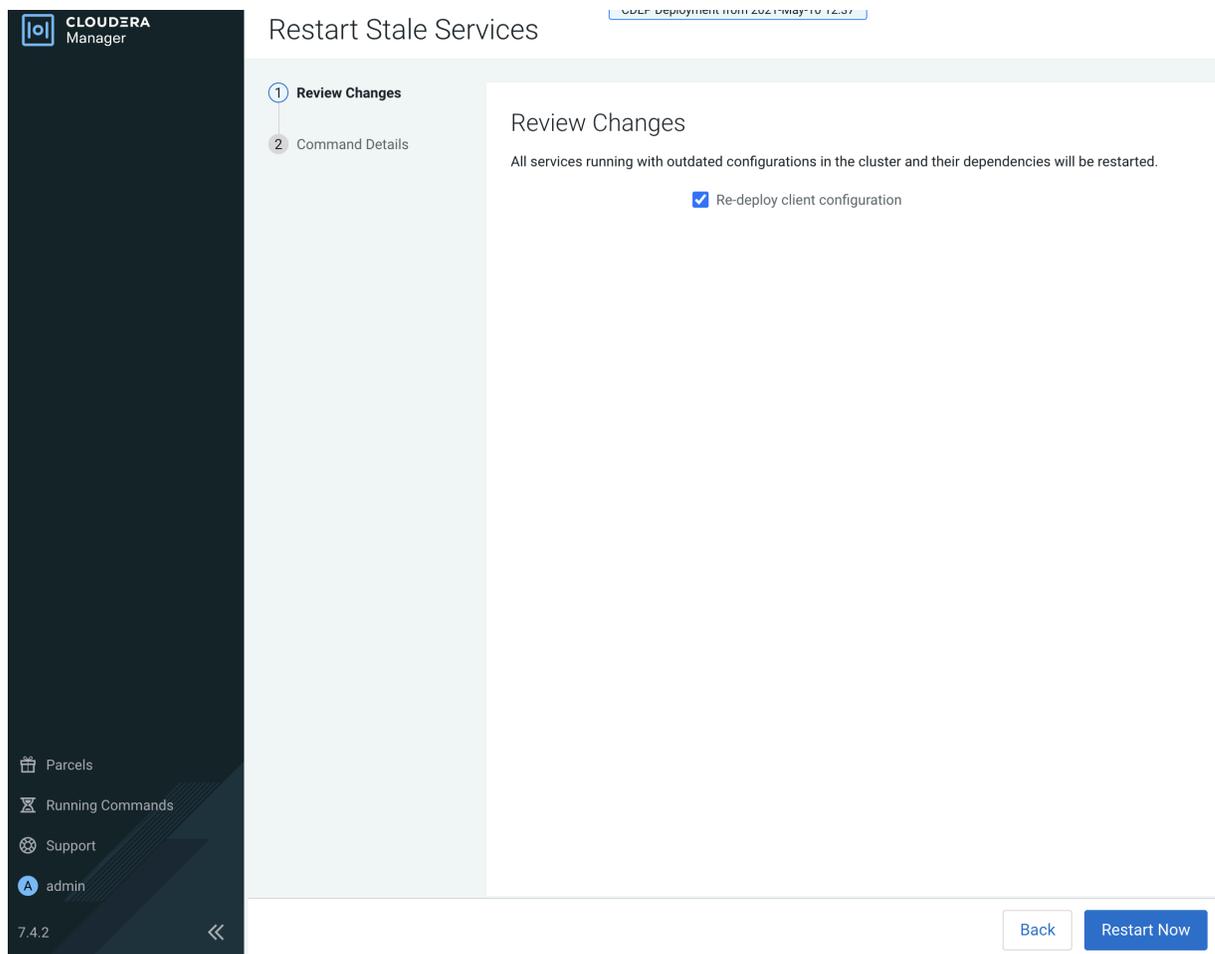
3. On the Stale Configurations page, click Restart Stale Services.

The screenshot displays the Cloudera Manager interface for a cluster named 'Cluster 1'. The main section is titled 'Stale Configurations'. On the left, there is a sidebar with navigation options: Clusters, Hosts, Diagnostics, Audits, Charts, Replication, Administration, Private Cloud (New), Parcels, Running Commands, Support, and a user profile for 'admin' (7.4.2).

The main content area is divided into several sections:

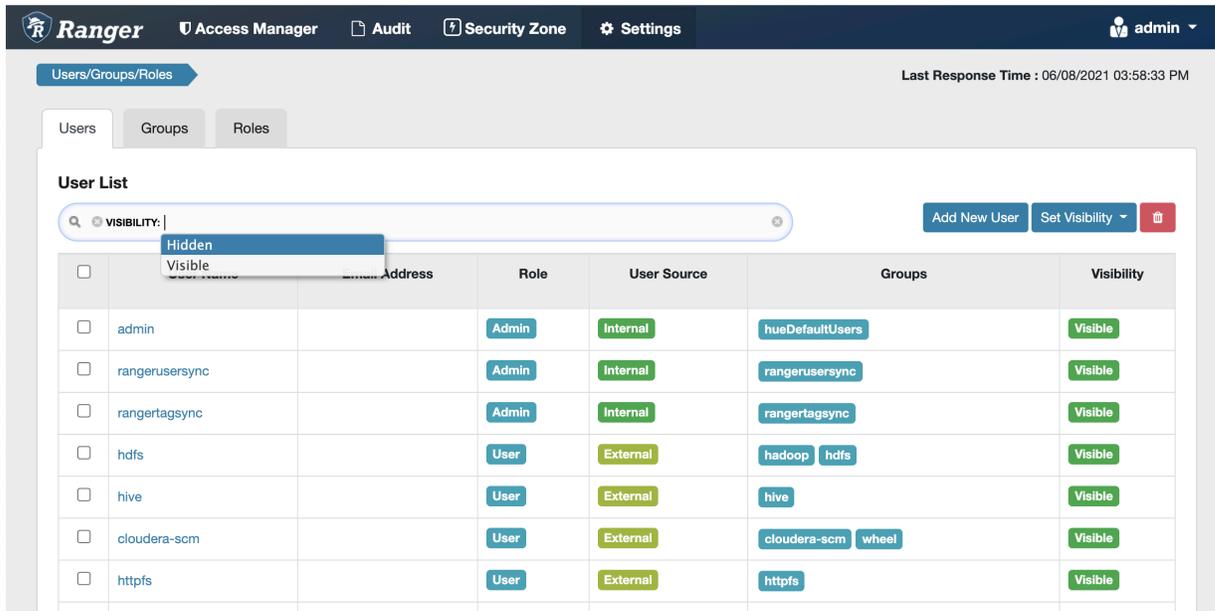
- Filters:** Includes 'Clear All' and sections for 'FILE', 'SERVICE', and 'ROLE TYPE'. Under 'SERVICE', 'RANGER-1' is highlighted with a count of 3.
- Configuration Files:** Three files are shown with their contents:
  - File: conf/ranger-ugsync-site.xml (RANGER-1(1)):** Shows XML configuration for usersync, including properties for kerberos principal, deleted users, and frequency.
  - File: conf/rangeradmin.properties (RANGER-1(1)):** Shows properties for the ranger service, including external URL, http/https ports, and SSL settings.
  - File: conf/rangeradmin.properties (RANGER-1(2)):** Shows similar properties for a second instance of the ranger service.
- Restart Stale Services:** A prominent blue button at the bottom right of the configuration view.

4. On the Restart Stale Services page, select the Re-deploy client configuration check box, then click Restart Now.



5. A progress indicator page appears while the services are being restarted. When the services have restarted, click Continue.

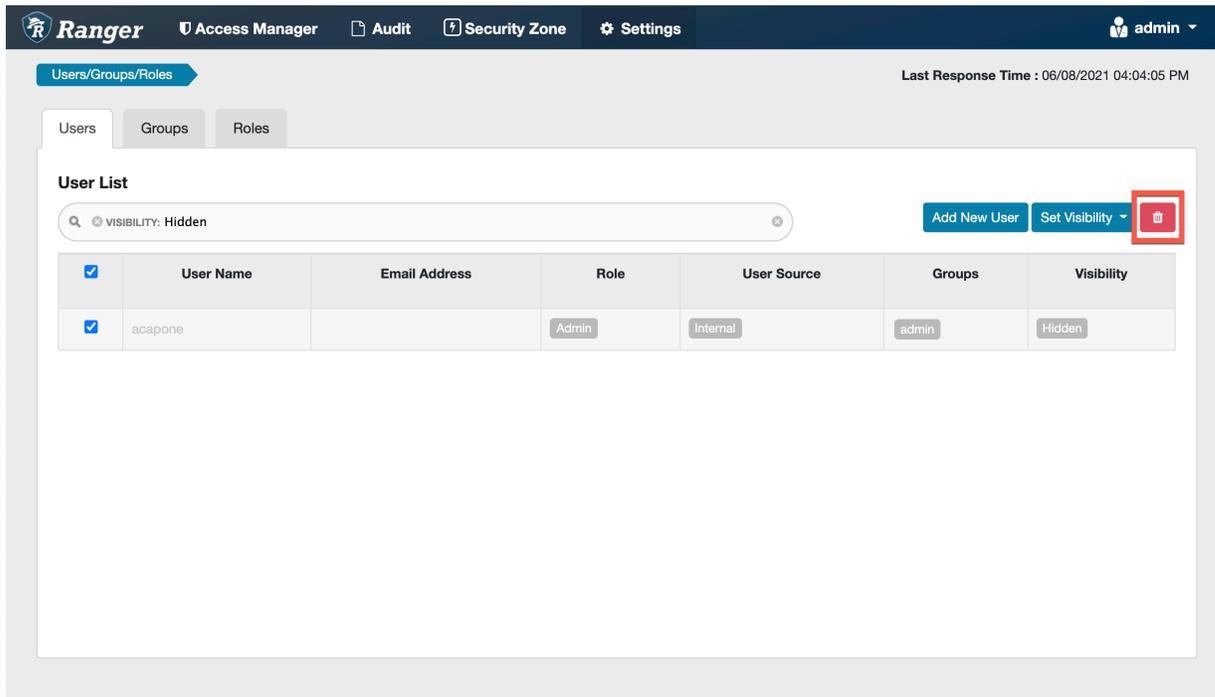
6. Users that have been deleted in sync source are not automatically deleted in Ranger – they are marked as Hidden and must be manually deleted by the Ranger Admin user, and then Ranger Usersync must be restarted. In the Ranger Admin Web UI, select Settings > Users/Groups/Roles. Click in the User List text box, then select Visibility > Hidden.



The screenshot shows the Ranger Admin Web UI interface. The top navigation bar includes 'Ranger', 'Access Manager', 'Audit', 'Security Zone', and 'Settings'. The user 'admin' is logged in. The main content area is titled 'Users/Groups/Roles' and shows a 'User List' table. A search filter 'visibility:' is applied, and a dropdown menu is open showing 'Hidden' and 'Visible' options. The table lists several users, including 'admin', 'rangerusersync', 'rangertagsync', 'hdfs', 'hive', 'cloudera-scm', and 'httpfs', all with a 'Visible' status.

	User Name	Email Address	Role	User Source	Groups	Visibility
<input type="checkbox"/>	admin		Admin	Internal	hueDefaultUsers	Visible
<input type="checkbox"/>	rangerusersync		Admin	Internal	rangerusersync	Visible
<input type="checkbox"/>	rangertagsync		Admin	Internal	rangertagsync	Visible
<input type="checkbox"/>	hdfs		User	External	hadoop hdfs	Visible
<input type="checkbox"/>	hive		User	External	hive	Visible
<input type="checkbox"/>	cloudera-scm		User	External	cloudera-scm wheel	Visible
<input type="checkbox"/>	httpfs		User	External	httpfs	Visible

7. To delete hidden users and groups, select the applicable check boxes, then click the red Delete icon.



The screenshot shows the Ranger Admin Web UI interface. The top navigation bar includes 'Ranger', 'Access Manager', 'Audit', 'Security Zone', and 'Settings'. The user 'admin' is logged in. The main content area is titled 'Users/Groups/Roles' and shows a 'User List' table. A search filter 'visibility: Hidden' is applied. The table lists one user, 'acapone', with a 'Hidden' status. The 'Delete' icon (a red trash can) is highlighted with a red box.

	User Name	Email Address	Role	User Source	Groups	Visibility
<input checked="" type="checkbox"/>	acapone		Admin	Internal	admin	Hidden

8. In Cloudera Manager, select Ranger > Ranger Usersync, then select Actions > Restart this Ranger Usersync.

The screenshot shows the Cloudera Manager interface for the Ranger Usersync service. The 'Actions' menu is open, with 'Restart this Ranger Usersync' selected. The 'Health Tests' section shows 4 Good and 2 Disabled tests. The 'Health History' section shows 2 tests becoming Good, 2 becoming Disabled, 1 becoming Good, and 1 becoming Unknown. A warning message is displayed, and two charts are visible at the bottom.



#### Note:

- Sync source is tracked when processing Ranger users and groups for deletion. If the same user name for a separate sync source already exists in Ranger DB, that user will not be updated or marked as hidden.
- For AD/LDAP sync:
  - Once a user is marked as deleted in Ranger, the user status will not be changed automatically until the user is manually deleted and Usersync is restarted to reflect any changes to the same user name in the source.
  - For example, a user (Bob) from one OU (say Engineering) is deleted from the source and is marked as deleted in Ranger admin. If the same user name (Bob) is subsequently added back to the same OU, the user status will not be automatically enabled. The user must be manually deleted and Usersync must be restarted to implement the changes.
  - If an identical user name (say Bob) is deleted from one OU (say Engineering) and added to a different OU (say Finance) between the sync cycles, user Bob is marked as hidden/deleted only when the delete cycle is triggered. Until then there is a security risk that user Bob from Finance will be granted the permissions for Bob from Engineering.