

Configuring Apache Ranger Authentication with UNIX, LDAP, or AD

Date published: 2019-08-21

Date modified:



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

Configuring Ranger Authentication with UNIX, LDAP, or AD.....	4
Configure Ranger authentication for UNIX.....	6
Configure Ranger authentication for AD.....	9
Configure Ranger authentication for LDAP.....	11
 Ranger AD Integration.....	 15
Ranger UI authentication.....	25
Ranger UI authorization.....	31
Ranger Usersync.....	35
Ranger user management.....	46
Known issue: Ranger group mapping.....	48

Configuring Ranger Authentication with UNIX, LDAP, or AD

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



CLUSTERA
Manager

 Clusters

 Hosts

 Diagnostics

 Audits

 Charts

 Backup

 Administration

Cluster 1

 RANGER-1

Actions ▾

Status Instances Configuration

authentication unix

Filters

▼ SCOPE	
RANGER-1 (Service-Wide)	0
Ranger Admin	4
Ranger Tagsync	0
Ranger Usersync	1
▼ CATEGORY	
Advanced	0
Logs	0
Main	4
Monitoring	0
Performance	0
Ports and Addresses	1
Resource Management	0
Security	0
Stacks Collection	0
▼ STATUS	
✖ Error	0
⚠ Warning	0

Related Information

[Cloudera Management Console](#)

Configure Ranger authentication for UNIX

How to configure Ranger to use UNIX for user authentication.

About this task



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

**CLUSTERA**
Manager

 Clusters

 Hosts

 Diagnostics

 Audits

 Charts

 Backup

 Administration

Cluster 1

 RANGER-1

Actions

Status

Instances

Configuration

authentication unix

Filters

▼ SCOPE

RANGER-1 (Service-Wide)

0

Ranger Admin

4

Ranger Tagsync

0

Ranger Usersync

1

▼ CATEGORY

Advanced

0

Logs

0

Main

4

Monitoring

0

Performance

0

Ports and Addresses

1

Resource Management

0

Security

0

Stacks Collection

0

▼ STATUS

 Error

0

 Warning

0

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 authen
Allow remote Login	Flag to enable/disable remote login. Only used if the Authentication method is UNIX.	TRUE	TRUE	No.
ranger.unixauth.service.hostname	The FQDN of the host where the UNIX authentication service is running. Only used if the Authentication method is UNIX. <code>{{RANGER_USERSYNC_HOST}}</code> is a placeholder value that is replaced with the host where Ranger Usersync is installed in the cluster.	localhost	myunixhost.domain.com	Yes, if selecte
ranger.unixauth.service.port	The port number where the ranger-usersync module is running the UNIX Authentication Service.	5151	5151	Yes, if selecte

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 Cloud, identity management is provided by FreeIPA, and configured using the Management Console. Therefore for CDP Cloud you should leave the Admin Authentication Method set to the UNIX authentication settings. For more information on FreeIPA, see the CDP Management Console 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 the AD settings.

The screenshot shows the Cloudera Manager interface for configuring Ranger-1. The search bar at the top contains the word "authentication". The left sidebar shows the navigation menu with "Clusters" selected. The main panel displays the configuration for "RANGER-1" under the "Configuration" tab. The search results show filters for SCOPE, CATEGORY, and STATUS. The configuration details for "Admin Authentication Method" are visible, showing "ACTIVE_DIRECTORY" selected. Other settings like "Admin UNIX Auth Remote Login", "Admin UNIX Auth Service Hostname", "Admin LDAP Auth User DN Pattern", "Admin LDAP Auth User Search Filter", and "Admin LDAP Auth Group Search Base" are also shown.

- 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 ranger.ldap.ad.base.dn	The Distinguished Name (DN) of the starting point for directory server searches.	N/A	dc=example,dc=com
Admin AD Auth Bind DN ranger.ldap.ad.bind.dn	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 ranger.ldap.ad.bind.password	Password for the bind.dn.	N/A	Secret123!
Admin AD Auth Domain Name ranger.ldap.ad.domain	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.	N/A	
Admin AD Auth User Search Filter ranger.ldap.ad.user.searchfilter	The search filter used for Bind Authentication.	N/A	

* 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 LDAP

How to configure Ranger to use LDAP for user authentication.

About this task



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

Procedure

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

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



CLOUDERA
Manager

 Clusters

 Hosts

 Diagnostics

 Audits

 Charts

 Backup

 Administration

Cluster 1

 RANGER-1

Actions

- Status
- Instances
- Configuration

authentication

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

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

Property	Description	Default value	Sample values
Admin Authentication Method	The Ranger authentication method.	UNIX	LDAP
Admin LDAP Auth Group Search Base ranger.ldap.group.searchbase	The LDAP group search base.	N/A	((CN=Hdp_users)(CN=Hdp_admins))
Admin LDAP Auth Group Search Filter ranger.ldap.group.searchfilter	The LDAP group search filter.	N/A	
Admin LDAP Auth URL ranger.ldap.url	The LDAP server URL	N/A	ldap://localhost:389 or ldaps://localhost:636
Admin LDAP Auth Bind User ranger.ldap.bind.dn	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	Password for the account that can search for users.	N/A	Secret123!
Admin LDAP Auth User Search Filter ranger.ldap.user.searchfilter	The LDAP user search filter.	N/A	
Admin LDAP Auth Base DN ranger.ldap.base.dn	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	The LDAP group role attribute.	N/A	cn
Admin LDAP Auth Referral ranger.ldap.referral*	See below.	ignore	follow ignore throw
Admin LDAP Auth User DN Pattern ranger.ldap.user.dnpattern	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.

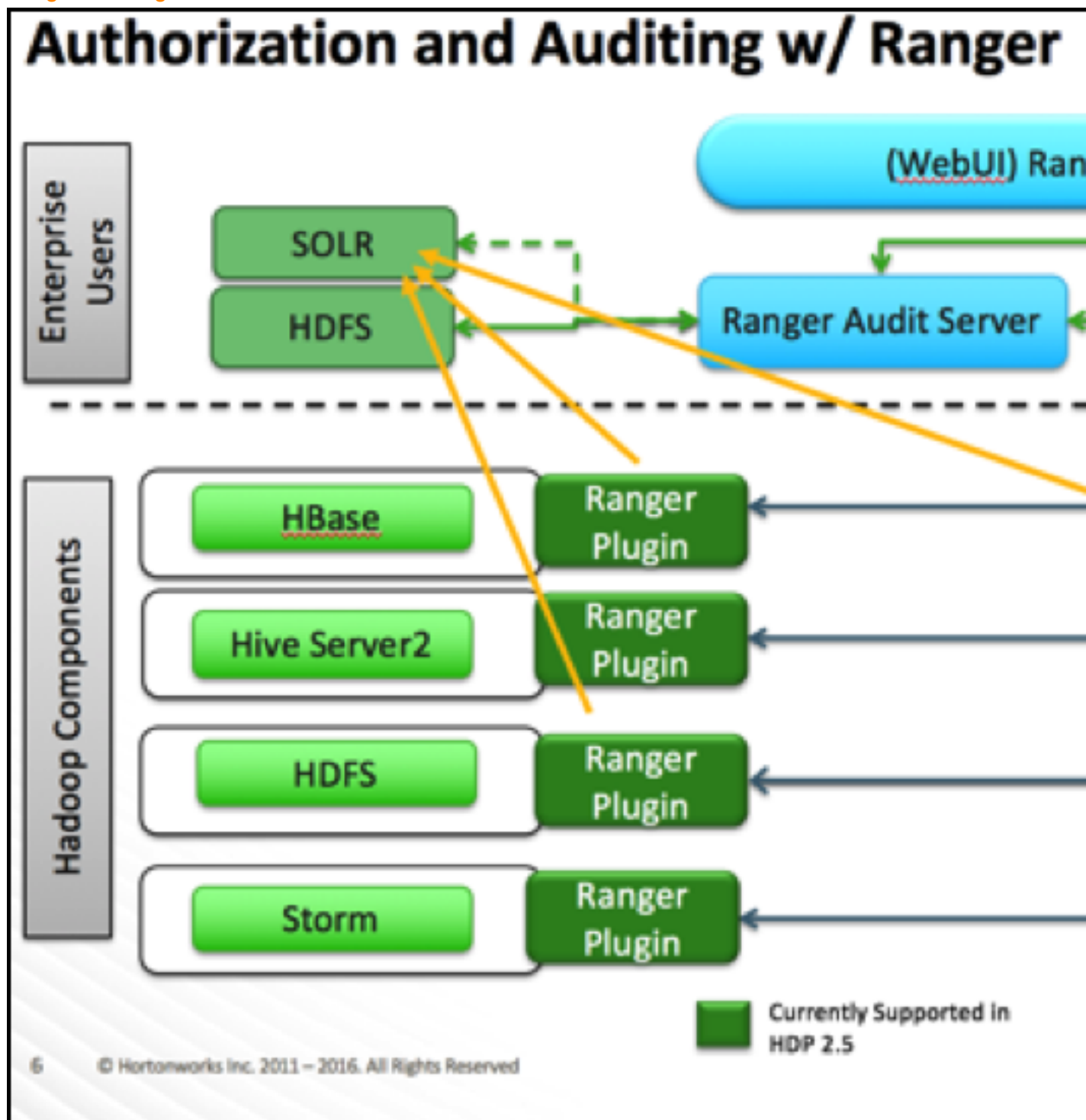
- When this property is set to ignore, it indicates that the server should return referral entries as ordinary entries (or plain text). This might return partial results for the search. In the case of AD, a `PartialResultException` is returned when referrals are encountered while search results are processed.

Related Information[Cloudera Management Console](#)

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:



CLUSTERA
Manager

Clusters

Hosts

Diagnostics

Audits

Charts

Backup

Administration

CDEP Deployment from 2019-Aug-05
11:11

Parcels

Recent Commands

Cluster 1



HDFS-1

Actions ▾

Status

Instances

Configuration

Filters

Clear All

▼ SCOPE

HDFS-1 (Service-Wide)51

Balancer0

DataNode1

Gateway0

HttpFS7

JournalNode0

NFS Gateway0

NameNode2

SecondaryNameNode0

Failover Controller0

▼ CATEGORY

Clear

Advanced95

Checkpointing2

Cloudera Navigator4

Erasure Coding4

High Availability5

Logs37

Main44

Monitoring100

18

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:

Ranger

Access Manager

[Service Manager](#)[Edit Service](#)

Config Properties :

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:

Ranger

Access Manag

Access

Admin

 Search for your plugins...

Export Date (Eastern Daylight Time)

08/13/2019 11:49:39 AM

08/13/2019 11:49:27 AM

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.

Ranger

Access Management

Access

Admin



START DATE: 08/14/201

Exclude Service Users : ☐

Policy ID	Policy Version	Ev
5	1	08/14/2

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:

Ra

 **Username:**

admin

 **Password:**

.....

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.

 **CLOUDERA**
Manager

Clusters

Hosts

Diagnostics

Audits

Charts

Backup

Administration

Cluster 1

 **RANGER-1**

Actions ▾

Status

Instances

Configuration

Filters

▼ SCOPE

RANGER-1 (Service-Wide)0

Ranger Admin19

Ranger Tagsync1

Ranger Usersync2

▼ CATEGORY

Advanced0

Logs0

Main21

Monitoring0

Performance0

Ports and Addresses1

Resource Management0

Security0

Stacks Collection0

▼ STATUS

Error0

28

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.

With `ACTIVE_DIRECTORY` it is not possible to limit the scope of users that can access the Ranger UI any further by refining the value of the `ranger.ldap.ad.user.searchfilter` property even further to :

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

This does NOT work with the `ACTIVE_DIRECTORY` option.

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.

 **CLOUDERA**
Manager

Clusters

Hosts

Diagnostics

Audits

Charts

Backup

Administration

Cluster 1

 **RANGER-1**

Actions ▾

Status

Instances

Configuration

Filters

▼ SCOPE

RANGER-1 (Service-Wide)0

Ranger Admin19

Ranger Tagsync1

Ranger Usersync2

▼ CATEGORY

Advanced0

Logs0

Main21

Monitoring0

Performance0

Ports and Addresses1

Resource Management0

Security0

Stacks Collection0

▼ STATUS

Error0

30

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.

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.

Ranger

Access Manager

Audit

Security Zone

Users/Groups/Roles

Users

Groups

Roles

User List

Search for your users...

☐	User Name	Email Address	
☐	admin		Ad
☐	rangerusersync		Ad
☐	rangertagsync		Ad
☐	hive		Us
☐	cloudera-scm		Us
☐	httpfs		Us
☐	superset		Us
☐	atlas		Us
☐	ranger		Us
☐	kudu		Us
☐	kms		Us
☐	accumulo		Us
☐	polkitd		Us
☐	nfsnobody		Us
☐	spark		Us

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

Ranger

Access Manager

Users/Groups/Roles

User Edit

User Detail

Basic Info

User Name *

ranger

First Name *

ranger

Last Name

Email Address

34

Only users with the Admin role can edit Ranger policies.

Ranger Usersync

Reference information on Ranger usersync, when configuring Ranger AD integration.

A vital part of the Ranger architecture is the ability to get users and groups from the corporate AD to use in policy definitions.

Ranger usersync runs as separate daemon:

 **CLOUDERA**
Manager

 **Clusters**

 Hosts

 Diagnostics

 Audits

 Charts

 Backup

 Administration

Cluster 1

 **RANGER-1**

Actions ▾

- Status
- Instances
- Configuration

Health Tests

 [Show 3 Good](#)

Status Summary

Ranger Admin	 1 Good Health
Ranger Tagsync	 1 Good Health
Ranger Usersync	 1 Good Health
Hosts	 1 Good Health

Health History

-   Ranger Admin Health Good
-   1 Became Bad
- 2 Became Good

It can also be refreshed using the Actions drop-down.

**CLOUDERA**
Manager

Clusters

Hosts

Diagnostics

Audits

Charts

Backup

Administration

Cluster 1

RANGER-1

Actions ▾

Status

Instances

Config

Health Tests

Show 3 Good

Status Summary

Ranger Admin

1 G

Ranger Tagsync

1 G

Ranger Usersync

1 G

Hosts

1 G

Health History

Ranger Admin Health Good

1 Became Bad

2 Became Good

Ranger Usersync Configuration

Usersync has a lot of moving parts and can have very different outcomes. Two main sets of properties govern the way users and groups are synchronized.

Without Enable Group Search First, the primary access pattern is user-based, and groups are only searched/added based on the users it finds first. In contrast, with Enable Group Search First enabled, the primary access pattern is group-based (in turn based on the group search filter) and users are only searched/added based on the group memberships it finds first.



CLUDERA
Manager

Search

Clusters

Hosts

Diagnostics

Audits

Charts

Backup

Administration

Cluster 1

✓

RANGER-1

Actions

Status

Instances

Configuration

Enable Group Search First

Filters

Clear All

SCOPE

Clear

RANGER-1 (Service-Wide)0

Ranger Admin0

Ranger Tagsync0

Ranger Usersync2

CATEGORY

Advanced0

Logs0

Main2

Monitoring0

Performance0

Ports and Addresses0

Resource Management0

Security0

Stacks Collection0

STATUS

Error0

Warning0

40

```
OU=CorpUsers,DC=field,DC=hortonworks,DC=com
```

```
Value of 'User Search Filter':
```

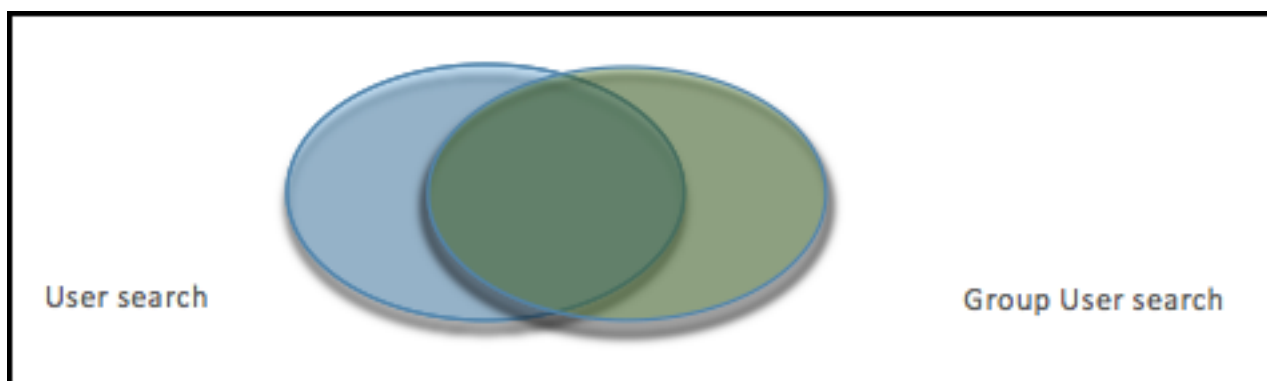
```
(|(memberOf=CN=Hdp_admins,OU=Company,OU=User Accounts,OU=CorpUsers,DC=field,DC=hortonworks,DC=com)(memberOf=CN=Hdp_users,OU=Company,OU=User Accounts,OU=CorpUsers,DC=field,DC=hortonworks,DC=com))
```

```
Value of 'User Group Name Attribute':
```

```
sAMAccountName
```

```
Value of 'Group Search Base':
```

```
(|(CN=Hdp_users)(CN=Hdp_admins))
```



Be aware that the filters on the group level limit the returns on the user search, and vice versa. In the graph above if the left oval represents the results of all users queried by the user configuration settings, and the right oval represents all users queried by the group configuration settings, the eventual set of users that make it to Ranger usersync is the overlap between the two.

Therefore it is recommended that you set the filters on both ends exactly the same to potentially have a 100% overlap in the ovals.

In the example configuration above, the scope of the usersync would be all members of the "Hdp_admins" and "Hdp_users" groups.

The best of both worlds is to have both Enable Group Search First and Enable User Search enabled.

The logging of a run of the usersync daemon can be retrieved from `/var/log/ranger/usersync/usersync.log` on the server hosting Ranger Admin. A successful run might output logging like below:

```

rscEnabled: true, userSearchEnabled: true, ldapReferral: ignore
08 Dec 2016 19:40:05 INFO UserGroupSync [UnixUserSyncThread] - Begin: init
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - LDAP
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Perf
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Addi
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Addi
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - No.
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Addi
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Addi
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Addi
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - No.
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - LDAP
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - User
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Upda
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Upda
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Upda
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Upda
08 Dec 2016 19:40:05 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Upda
08 Dec 2016 19:40:06 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Upda
08 Dec 2016 19:40:06 INFO LdapUserGroupBuilder [UnixUserSyncThread] - LDAP
08 Dec 2016 19:40:06 INFO UserGroupSync [UnixUserSyncThread] - End: initia
08 Dec 2016 19:40:06 INFO UserGroupSync [UnixUserSyncThread] - Done initia

```

From that log it clearly shows that the groups are synced first and that all users belonging to those groups are then retrieved according to its own settings, after which the user parts are enriched/overwritten by the returns from the user queries.

Beware:

If you don't enable Enable User Search, that enrichment does NOT happen. Logging for such a run looks like this:

```

08 Dec 2016 18:24:28 INFO LdapUserGroupBuilder [UnixUserSyncThread] - LdapUserGroupBuilder info: password: *****, ldapAuthenticationMechanism: simple, searchBase: dc=hadoop,dc=apache,dc=org, user, userSearchFilter: (memberOf=OU=Hdp_admins,OU=,OU=User Accounts,OU=CorpUsers,DC=CorpUsers,DC=field,DC=hortonworks,DC=com)), userNameAttribute: sAMAccountName, userSearchAttribute, groupSearchBase: [OU=,OU=User Accounts,OU=CorpUsers,DC=field,DC=hortonworks,DC=com], (objectclass=group)(!(OU=Hdp_users)(OU=Hdp_admins)), extendedAllGroupsSearchFilter: (&(objectclass=group), member], groupUserMapSyncEnabled: true, groupSearchFirstEnabled: true, userSearchEnabled: true
08 Dec 2016 18:24:28 INFO UserGroupSync [UnixUserSyncThread] - Begin: initial load of user/group
08 Dec 2016 18:24:28 INFO LdapUserGroupBuilder [UnixUserSyncThread] - LDAPUserGroupBuilder update
08 Dec 2016 18:24:28 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Performing Group search
08 Dec 2016 18:24:28 INFO PolicyMgrUserGroupBuilder [UnixUserSyncThread] - Using principal = j.kirk
08 Dec 2016 18:24:28 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Adding Hdp_users to user
08 Dec 2016 18:24:28 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Adding Hdp_users to user
08 Dec 2016 18:24:28 INFO LdapUserGroupBuilder [UnixUserSyncThread] - No. of members in the group
08 Dec 2016 18:24:28 INFO PolicyMgrUserGroupBuilder [UnixUserSyncThread] - Using principal = j.kirk
08 Dec 2016 18:24:29 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Adding Hdp_admins to user
08 Dec 2016 18:24:29 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Adding Hdp_admins to user
08 Dec 2016 18:24:29 INFO LdapUserGroupBuilder [UnixUserSyncThread] - Adding Hdp_admins to user
08 Dec 2016 18:24:29 INFO LdapUserGroupBuilder [UnixUserSyncThread] - No. of members in the group
08 Dec 2016 18:24:29 INFO LdapUserGroupBuilder [UnixUserSyncThread] - LDAPUserGroupBuilder.get
08 Dec 2016 18:24:29 INFO LdapUserGroupBuilder [UnixUserSyncThread] - User search is disabled
08 Dec 2016 18:24:29 INFO LdapUserGroupBuilder [UnixUserSyncThread] - longUserName:
08 Dec 2016 18:24:29 INFO LdapUserGroupBuilder [UnixUserSyncThread] - longUserName:
08 Dec 2016 18:24:29 INFO LdapUserGroupBuilder [UnixUserSyncThread] - longUserName:
08 Dec 2016 18:24:30 INFO LdapUserGroupBuilder [UnixUserSyncThread] - longUserName:
08 Dec 2016 18:24:30 INFO LdapUserGroupBuilder [UnixUserSyncThread] - longUserName:
08 Dec 2016 18:24:30 INFO UserGroupSync [UnixUserSyncThread] - End: initial load of user/group
08 Dec 2016 18:24:30 INFO UserGroupSync [UnixUserSyncThread] - Done initializing user/group sync

```

The result in the Ranger UI are other user names (LongUserName) derived from "member" group attributes full DN. You get the long name "James Kirk" in the Ranger userlist in stead of "j.kirk". Ranger does not treat those as one and the same user. Policies that are defined for user "k.reshi" will not map to the user "Kvothe Reshi", and vice versa. To prevent any confusion it is probably best to delete the long username versions from the Rangers user list.

**Important:**

On the first page of Rangers user list there are many system users. Most of them were put there by the Ranger installer and during the plugins installs:

Users/Groups/Roles

Users

Groups

Roles

User List

Search for your users...

☐	User Name	Email Address
☐	admin	
☐	rangerusersync	
☐	rangertagsync	
☐	hive	
☐	cloudera-scm	
☐	httpfs	
☐	superset	
☐	atlas	
☐	ranger	
☐	kudu	
☐	kms	
☐	accumulo	
☐	polkitd	
☐	nfsnobody	
☐	spark	

Do NOT remove these system users!

There are basic access policies based on those system users designed to keep a Ranger-governed component working after Ranger is given all control over that component's authorizations. Without those policies/users many components may not function as expected.

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.

Ranger

Access Manager

Users/Groups/Roles

Users

Groups

User List

Search for your users...

☐	User Name
☐	hdfs
☐	rangerlookup
☐	livy

Known issue: Ranger group mapping

For Ranger AD integration, there is an issue with Ranger not being able to map a user on a group 'Hdp_admins' to a policy that allows/denies access to the group 'Hdp_admins'. The issue is the upper case characters that might be in a AD group name definition.

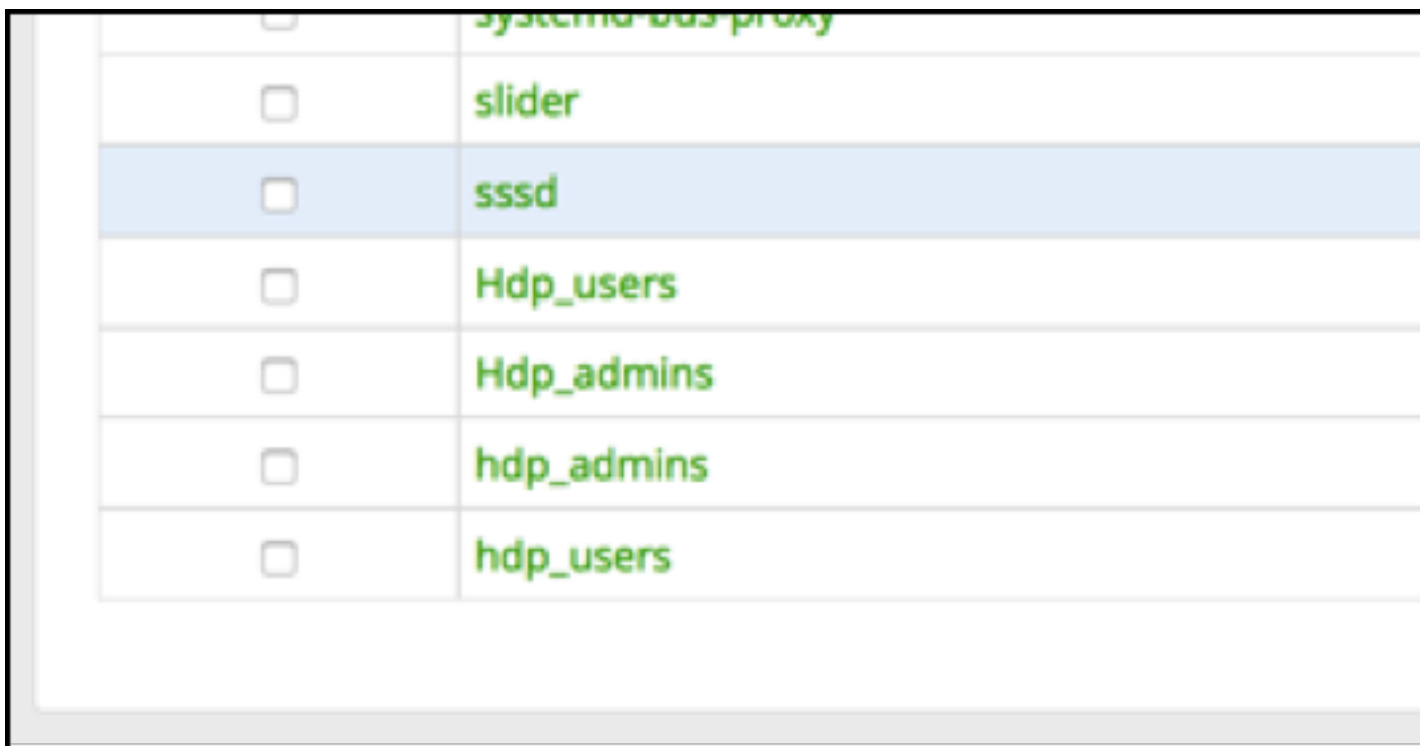
Most HDP components get the group information for a user via the SSSD daemon. When asked for the groups the user 'd.threpe' belongs to we get:

```
[centos@rjk-hdp25-m-01 ~]$ groups d.threpe  
d.threpe : domain_users hdp_admins hadoop
```

So 'hdp_admins' all in lower case. Ranger does not treat this as the same value as 'Hdp_admins' which came via the group sync and was applied to some policies.

There is no way to make the group sync write or retrieve the group names all in lower case since there is no AD attribute that rewrites it in lowercase.

This issue can be worked around fortunately (till it gets solved). The solution is to define a local group in Ranger as a shadow group of a real group from AD, but then all in lower case:



☐	systemd-bus-proxy
☐	slider
☐	sssd
☐	Hdp_users
☐	Hdp_admins
☐	hdp_admins
☐	hdp_users

If we now create policies and use that lower case 'shadow' group literal the result is that policies are correctly mapped to the AD groups again:

List of Policies : HDP_ atlas

Search for your policy...

Policy ID	Policy Name	Status	
9	all - taxonomy	Enabled	Enab
10	all - operation	Enabled	Enab
11	all - type	Enabled	Enab
12	all - entity	Enabled	Enab
13	all - term	Enabled	Enab

*The 'Hdp_admins' entry does not have to be there, it is shown for clarification only. 'hdp_admins' is necessary to make it work.