Managing and Monitoring Cloudbreak

Date of Publish: 2019-02-06

https://docs.hortonworks.com/

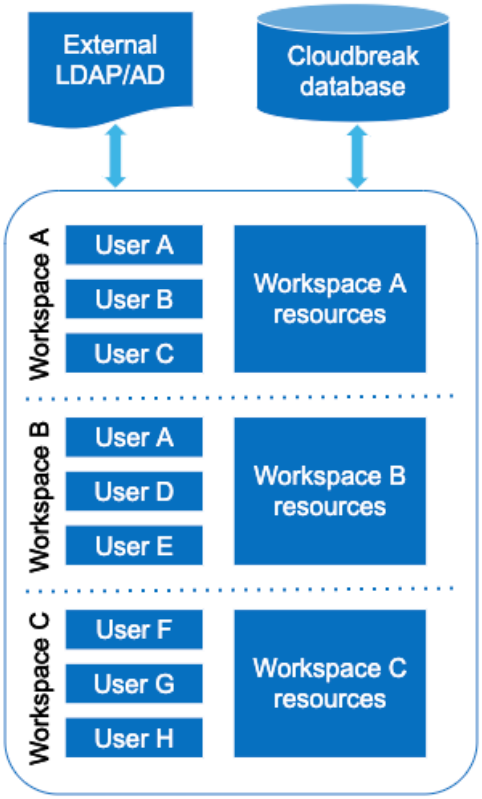
Contents

Workspaces

The user authorization model used by Cloudbreak allows resource sharing between users.

When a Cloudbreak instance is configured with an LDAP or Active Directory, the LDAP/AD users can share Cloudbreak resources with other users via workspaces. Any user can create a workspace and invite other users to that workspace, granting them specific access rights. Once a workspace is created, users who are part of it share all resources, such as clusters, blueprints, recipes, and so on, created as part of that workspace.

This user authorization model is summarized in the following diagram:



As the diagram illustrates:

- Users can be members of multiple workspaces and access any resources created within these workspaces (shared).
- Resource sharing allows users to access resources created by other users within the same workspace (credentials, clusters, blueprints, recipes, images, and external sources) but resources cannot be shared or moved between workspaces.

Example scenario

Consider the following example: Mark, Sarah, and Jeff work on the same project and instead of accessing Cloudbreak independently, they would like to share the same HDP clusters to run their workloads. Therefore, Mark creates a workspace called “Marketing-analytics” and invites Sarah and Jeff, granting them the read and write access. Now the three of them, in addition to accessing their private Cloudbreak accounts, can access the “Marketing-analytics” account and have read and write all the resources. Mark is still the only one who can manage the workspace. Once he receives the invite, Jeff accesses the “Marketing-analytics” workspace, registers a blueprint and a few recipes, and then creates a cluster. Now all three of them can access these resources as part of the “Marketing-analytics” workspace.

At the same time, Sarah is involved in another project, which requires her to work with a separate group of people. Therefore, she decides to create a separate workspace called “Finance-reports” and share it with Richard and Monica, granting them read and write access to all the resources. Now in addition to having her own Cloudbreak account, Sarah is part of two workspaces “Marketing-analytics” and “Finance-reports”.

Workspace permissions

Currently, Cloudbreak only supports three general access types:

User type	Permission	Access
Workspace owner	All: Read	Create/edit/delete resources
Workspace user	All: Write	View resources
Workspace user	Workspace: Manage	• Delete the workspace • Add/remove users, edit user's access • Create/edit/delete resources

Access permissions can be assigned per workspace, without the ability to restrict them per specific resource type (clusters, blueprints, and so on). This means that all members of a given workspace have access to all resources.

The user who creates a workspace is automatically assigned the "All: Read", "All: Write", and "Workspace: Manage" access permission.

Prerequisites for using workspaces

Sharing resources via workspaces is only possible when Cloudbreak is configured with an LDAP/AD.

Note:

Only users who have previously logged in to Cloudbreak can be added to a shared workspace.

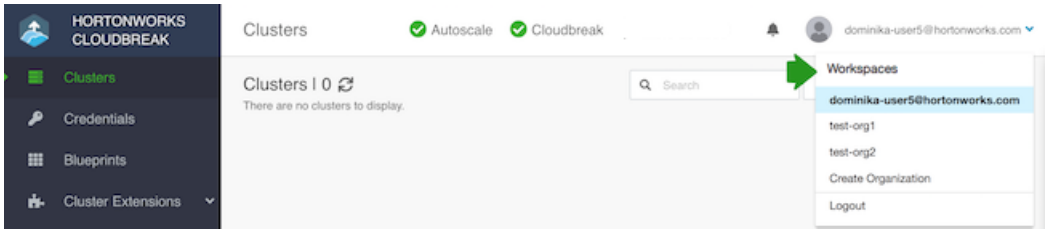
Related Information

Configuring Cloudbreak for LDAP/AD

Manage workspaces

You can create new workspaces and manage existing workspaces from the Cloudbreak web UI and CLI.

To manage workspaces, navigate to the Workspaces page accessible from the dropdown in the top-right corner in the Cloudbreak web UI:



Create a workspace

Any user can create a workspace by using the following steps:

Steps

1. Click on Create Workspace.
2. Provide a name and description.
3. Click Create.

After creating a workspace, your workspace will be available and you will be added as a member of this workspace with the full access rights. Your workspace will be accessible from the dropdown in the top-right corner. The next step is to add users to your workspace.

Add a user to a workspace

A workspace owner can add users to a workspace by using the following steps:

Steps

1. Click on the workspace name.
2. Under Users, select one or more the users that you would like to add and click on the  button.
3. This will add the user(s) to the workspace and the selected users will appear under Members.

   Automatically, the "All: Read" access permission is assigned to a newly added user. You can edit the access permissions once your user appears under Members.

   When a user is invited to a workspace, there is no notification in the UI and a page refresh is needed in order for the new workspace to be visible in the web UI.

Edit user's access

A workspace owner can edit user's access to a workspace by using the following steps:

Steps

1. Click on the workspace name.
2. Under Members, select one or more users and click on Edit permissions.
3. Select or deselect permissions.
4. Click Save.

Remove users from a workspace

A workspace owner can remove users from a workspace by using the following steps:

Steps

1. Click on the workspace name.
2. Under Members, select one or more users.
3. Click on  .
4. Confirm delete.

When a user is removed from a workspace, there is no notification in the UI and a page refresh is needed in order for the workspace to be removed from the web UI.

Delete a workspace

A workspace owner can delete a workspace by using the following steps:

Steps

1. Select the checkbox next to the workspace name.
2. Click on the  icon.
3. Confirm delete.

A page refresh is needed in order for the workspace to be removed from the web UI.

Managing workspaces from the CLI

You can create new workspaces and manage existing workspaces from the CLI.

To create an manage workspaces from the CLI, use the following commands:

- cb workspace (describe, list, create, delete, add-user, remove-user)
- cb user (list)

For more information, refer to the CLI documentation

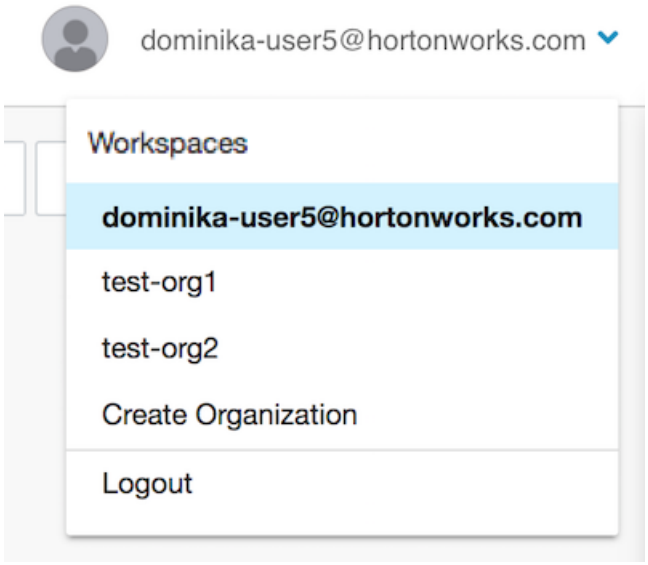
Access a workspace

As a member of multiple workspaces, you can switch between the workspaces in the UI and CLI in order to access workspace-specific resources.

Access a workspace from the web UI

To navigate to either your private account or a workspace that you are part of:

1. Click on the arrow next to your account name in the top-right corner.
2. Select either your private account or a workspace:



3. This will load Cloudbreak resources that are part of the selected account, allowing you to access them from the UI.

Access a workspace from the CLI

There are two options to access a workspace via the CLI:

Option	When to use
Configure CLI to use a workspace permanently	This method is useful if you would like to use one workspace permanently.
Provide the –-workspace flag with each command	This method is useful if you would like to switch between multiple workspaces.

Configure CLI to use a workspace permanently

To configure the CLI to use a specific workspace permanently, use:

```
cb configure --username <value> --password <value> --server <values> --
workspace <value>
```

For example:

```
cb configure --username  test-user@hortonworks.com --password MyPassword123!
 --server http://191.118.89.112 --workspace marketing
```

This will permanently save the configuration in ~/.cb/config so you don't need to provide the –workspace flag when running CLI commands. Here is an example config file:

```
default:
  username: test-user@hortonworks.com
  password: MyPassword123!
  server: http://191.118.89.112
  workspace: marketing
```

Provide the –-workspace flag with each CLI command

If you do not want to permanently configure a specific workspace in the Profile, you can simply provide the –-workspace <value> flag with each CLI command. For example, the following command returns all clusters that are part of the workspace called "marketing":

```
cb cluster list --workspace marketing
```

Operations audit logging

Cloudbreak records an audit trail of the actions performed by Cloudbreak users as well as those performed by the Cloudbreak application.

This includes actions related to creating, modifying, and deleting Cloudbreak-managed resources such as clusters, credentials, blueprints, cluster extensions, and external sources. For each action, information about the event type, resource name, resource type, timestamp, user, and status output is captured.

The output can be accessed as follows:

- A root-level administrator can enable writing the audit logging output to a log file or Kafka queue for ingestion into log management and event correlation systems.
- Any user can access their own logs from the UI or CLI. Similarly, any member of an organization can access the organization's logs.

Anonymization is applied to all sensitive information such as passwords, cloud credentials, access keys, and secret keys. Email addresses are included in the audit log.

Enable audit logging output

By default, Cloudbreak does not have audit logging enabled.

As a root-level administrator, you can do one or both of the following:

- Enable audit log file, which allows you to see aggregated logs for all users.
- Enable audit log sending to Kafka.

Enable audit log file

You can enable audit log file by performing the following steps.

Steps

1. Navigate to the deployment directory and open the Profile file for editing. For example:

```
cd /var/lib/cloudbreak-deployment/
vi Profile
```

2. Append the following to the Profile:

```
export CB_AUDIT_FILE_ENABLED=true
```

3. Restart Cloudbreak.
4. Cloudbreak will create the audit log file under <CLOUDBREAK_DEPLOYMENT_DIRECTORY>/logs/cloudbreak/cb-audit.log. Where <CLOUDBREAK_DEPLOYMENT_DIRECTORY> is the location where cbd is deployed (usually /var/lib/cloudbreak-deployment/).

Example output to log file:

```
{"errorConsumer":null,"id":"060bd4c0-a0ba-11e8-96af-a543d6cf1e8e","headers":
{"origin":null},"replyTo":null,"key":"SAVE_STRUCTURED_EVENT_TO_FILE","data":
{"type":"StructuredNotificationEvent","operation":
{"eventType":"NOTIFICATION","resourceId":1,"resourceName":"hdpcluster1","resourceType":"
a83f-fcafefb55ab3","userId":"c59bd2ae-53e3-498f-a83f-
fcafefb55ab3","userName":"test@hortonworks.com","cloudbreakId":"53e939c8-1135-4973-907b-
afe6bd3711c6","cloudbreakVersion":"2.8.0-dev.362","zonedDateTime":{"offset":
{"totalSeconds":0,"id":"Z","rules":{"fixedOffset":true,"transitions":
[],"transitionRules":[]}},"zone":{"totalSeconds":0,"id":"Z","rules":
{"fixedOffset":true,"transitions":[],"transitionRules":
[]}},"month":"AUGUST","dayOfYear":227,"dayOfWeek":"WEDNESDAY","year":2018,"monthValue":8
{"calendarType":"iso8601","id":"ISO"}}},"notificationDetails":
{"notificationType":"BILLING_STARTED","notification":"Billing started,
 Infrastructure successfully provisioned","cloud":"GCP","region":"us-
east1","availabiltyZone":"us-east1-
b","stackId":1,"stackName":"hdpcluster1","stackStatus":"CREATE_IN_PROGRESS","nodeCount"
data-science-spark2","blueprintId":7},"duration":0,"status":"SENT"}}
```

Enable audit log sending to Kafka

You can enable audit log sending to Kafka by performing the following steps.

Steps

1. Navigate to the deployment directory and open the Profile file for editing. For example:

```
cd /var/lib/cloudbreak-deployment/
vi Profile
```

2. Append the following to the Profile:

```
export CB_KAFKA_BOOTSTRAP_SERVERS=<server1>:<port1>,<server2>:<port2>
```

   Where each <server>:<port> is a Kafka server and port. For example:

```
export CB_KAFKA_BOOTSTRAP_SERVERS=kafka-broker1.mycompany.com:9092,kafka-
broker2.mycompany.com:9092
```

3. Restart Cloudbreak.

Example JSON output to Kafka:

```
{
    "type": "StructuredRestCallEvent",
    "operation": {
```

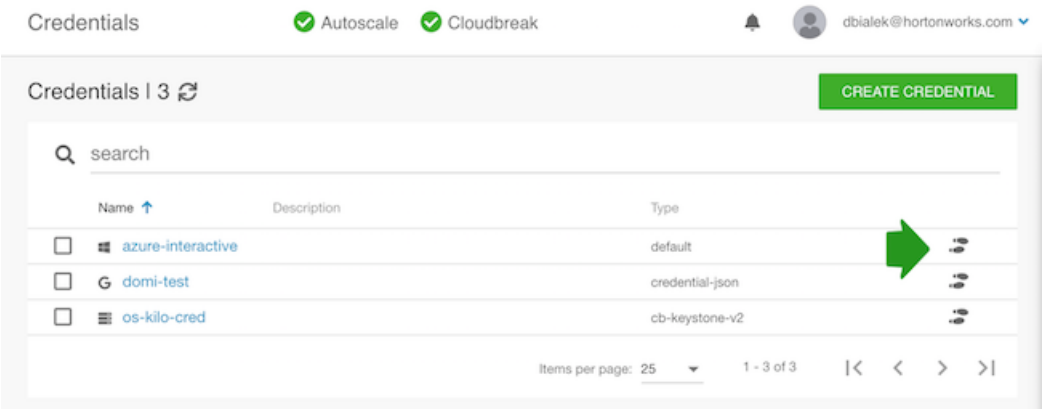
```
        "eventType": "REST",
        "resourceId": null,
        "resourceName": null,
        "resourceType": "util",
        "timestamp": 1533721820279,
        "account": "a10141ba-a8dc-4d2d-af46-3d77c53867d0",
        "userId": "a10141ba-a8dc-4d2d-af46-3d77c53867d0",
        "userName": "test@hortonworks.com",
        "cloudbreakId": "c71f28e3-567e-4379-8a0e-669e31561bbf",
        "cloudbreakVersion": "2.8.0-dev.321",
        "zonedDateTime": {
            "offset": {
                "totalSeconds": 0,
                "id": "Z",
                "rules": {
                    "transitions": [],
                    "transitionRules": [],
                    "fixedOffset": true
                }
            },
            "zone": {
                "totalSeconds": 0,
```


Audit resource events

As a user, you can check events related to resources that are part of your account or organization.

Audit resource events from the web UI

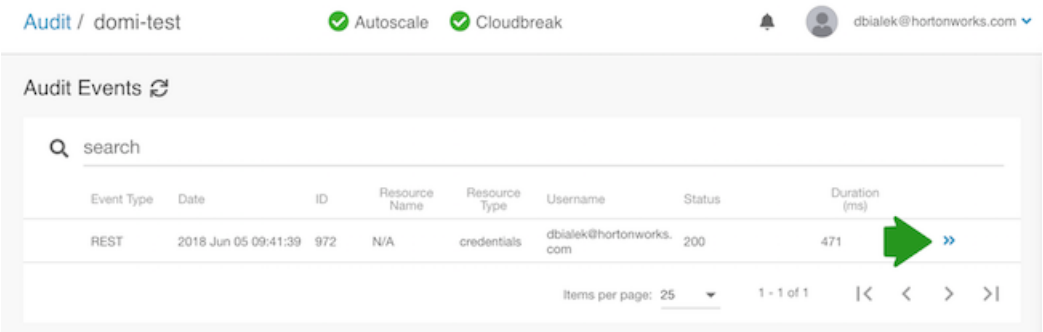
1. From the navigation menu, click on the name of a resource (for example, clusters) in order to navigate to its page.
2. From the list of resources (for example clusters), find the specific resource (for example “test-cluster”) and click on the icon next to it:



3. The Audit Events page including events related to the resource is displayed. The most recent event is listed at the top of the page. The following information in included for each event:
   - Event Type
   - Date
   - Resource ID
   - Resource Name
   - Resource Type
   - Username
   - Status

- Duration

4. Click on the » next to a specific event to get more information about the event:



Audit resource events from the CLI

As a user, you can use the cb audit command to check events related to resources that are part of your account or organization.

List all audit events

1. Obtain Resource ID either from the output of the cb <resource> describe command (for example cb cluster describe) or from the web UI.

2. List all audit events for a specific resource (such as cluster):

cb audit list <resource-name> --resource-id <value>

Display audit event details

1. Obtain audit IDs from the output of cb audit list or from the web UI.

2. Display details of a specific audit event identified by an audit-id:

cb audit describe --audit-id <value>