



Collibra Data Intelligence Cloud

Edge Infrastructure

Collibra Data Intelligence Cloud - Edge Infrastructure

Release date: February 5, 2023

Revision date: Thu Feb 02, 2023

You can find the most up-to-date technical documentation on our Documentation Center at

https://productresources.collibra.com/docs/collibra/latest/Content/Edge/to_edge.htm

Contents

Contents	ii
Introducing Edge	viii
Edge Responsibilities	viii
Edge components	ix
Integration steps	ix
Edge security	x
Communication between Edge and Collibra	xi
Communication between Edge and other services	xiii
Authentication to data sources	xiv
Security scanning	xv
Storing secrets	xvi
Customer Credentials	xviii
Credentials storage	xviii
Secret encryption	xviii
Credential encryption	xviii
Credentials transfer	xix
Platform credentials	xix
Data samples in Edge	xx
Edge Cache	xxi
Edge service repository	xxii
Monitoring and logging	xxiii
Host hardening on K3S-based integration	xxiv
Prerequisites	xxiv

Enable host hardening	xxiv
Disable host hardening	xxv
Installing an Edge site	xxvi
About an Edge site installation	xxvii
Properties	xxvii
Statuses	xxviii
Installation directories on K3S	xxviii
System requirements of an Edge site	xxx
Software requirements	xxx
Hardware requirements	xxx
Network requirements	xxxii
EKS requirements	xxxiii
Software requirements	xxxiv
Hardware requirements	xxxv
Network requirements	xxxv
Create an Edge site	xxxvii
Prerequisites	xxxvii
Steps	xxxvii
What's next?	xxxviii
Install an Edge site	xxxix
Prerequisites	xxxix
Steps	xxxix
Configure a forward proxy	xliv
Steps	xliv
What's next?	xlix
Enable or disable classification on an Edge site	I

Enable classification	I
Disable classification	li
Reinstall an Edge site using backup and restore	liii
Upgrade the operating system of an Edge site	liv
Steps	liv
Troubleshooting	liv
Enable Manual Upgrade for Edge sites	lvi
New Edge sites	lvi
Existing Edge sites	lvii
Upgrade an Edge site	lviii
Software Bill of Materials	lx
Version compatibility	lx
What's next?	lxi
Private Docker Registry	lxii
Approaches	lxii
Configure private docker registry	lxvii
Edge connections	lxix
Edit a connection	lxx
Prerequisites	lxx
Steps	lxx
Delete a connection	lxxi
Prerequisites	lxxi
Steps	lxxi
Create a Shared storage connection	lxxii
Before you begin	lxxii
Prerequisites	lxxii

Steps	lxxii
What's next?	lxxiii
JDBC connections	lxxiv
Data sources supported by Edge	lxxv
Create a JDBC connection	lxxv
Available Catalog connectors	lxxv
Edit a JDBC connection	lxxv
Available Catalog connectors	lxxv
Delete a JDBC connection	lxxvi
Prerequisites	lxxvi
Steps	lxxvi
Use keys to access a database	lxxvii
Edge capabilities	lxxviii
About Edge capabilities	lxxix
Capability templates	lxxix
Capability template structure	lxxx
Page layout	lxxx
About Edge capabilities connecting to data sources	lxxxii
Connection types	lxxxii
Add an Edge capability to an Edge site	lxxxiii
Prerequisites	lxxxiii
Steps	lxxxiii
More information	lxxxiii
Edit an Edge capability of an Edge site	lxxxv
Prerequisites	lxxxv
Steps	lxxxv

Delete an Edge capability from an Edge site	lxxxvi
Prerequisites	lxxxvi
Steps	lxxxvi
Edge Jobs dashboard	lxxxvii
Cancel jobs	lxxxix
Prerequisites	lxxxix
Steps	lxxxix
Maintaining Edge sites	xc
Running Edge tools	xc
Prepare the Edge tools on K3S	xc
Overview Edge commands on K3S	xc
Prepare Edge tools on EKS	xcii
Overview Edge commands on EKS	xciii
Edit an Edge site	xcv
Prerequisites	xcv
Steps	xcv
Update Edge user password	xcvi
Steps	xcvi
Update the outbound proxy configuration	xcvii
Steps	xcvii
Help file of the script	xcvii
Back up an Edge site	xcviii
Restore an Edge site	xcix
Delete an Edge site	ci
Prerequisites	ci
Steps	ci

Troubleshooting Edge	ciii
General troubleshooting Edge	civ
Use an explicit resolv.conf file for Edge	cvi
Edge logging	cvii
Edge diagnostics file	cvii
Edge infrastructure log files	cvii
Metadata connector log files	cviii
Edge system monitoring	cviii
Create an Edge diagnostics file	cx
Prerequisites	cx
Steps	cx
What's next?	cx
Create Metadata connector log files	cxii
Prerequisites	cxii
Steps	cxii
Prerequisites	cxiii
Steps	cxiii
Enable debug logging for Edge infrastructure logs	cxiv
Prerequisites	cxiv
Steps	cxiv
Disable OpenTelemetry	cxvi
Disable OpenTelemetry at installation time	cxvi
Edge FAQ	cxvii

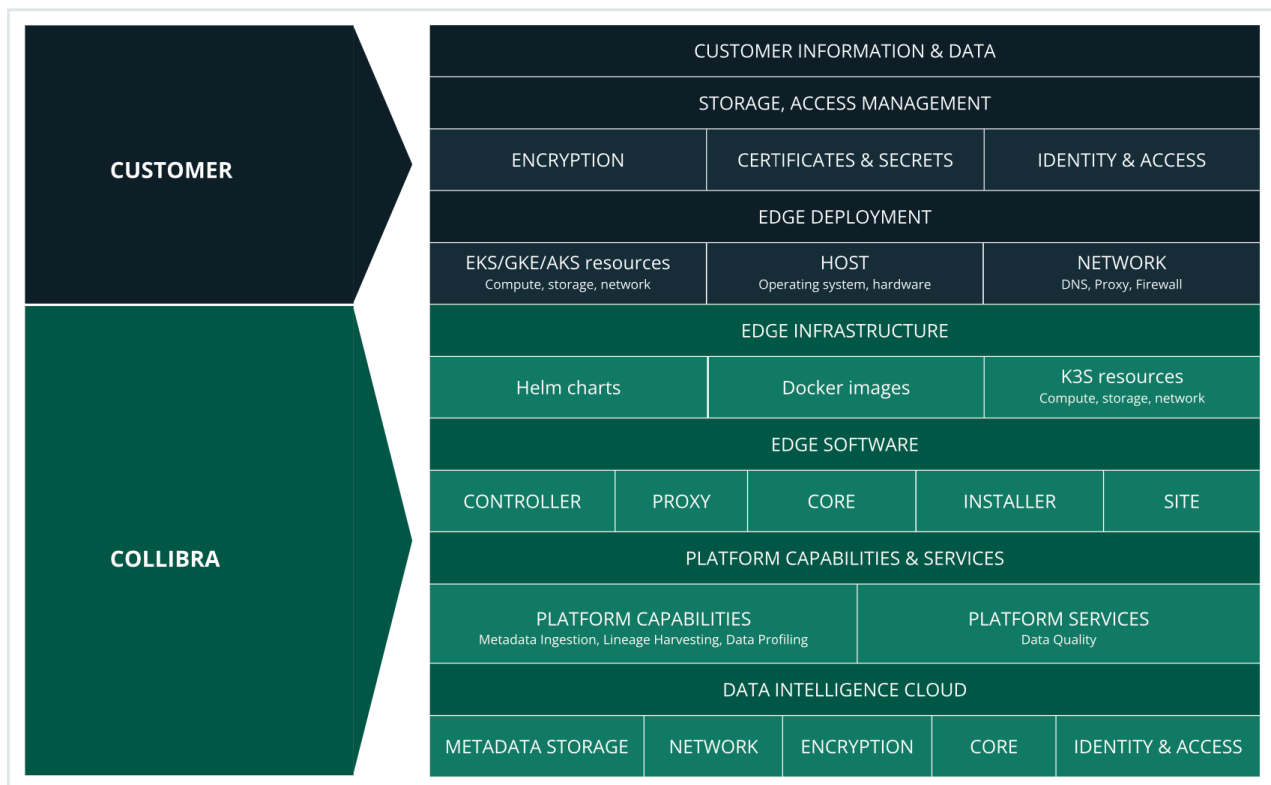
Introducing Edge

Edge is a cluster of Linux servers for accessing and processing data close to where it resides. It helps to connect to data sources and process information within your data landscape.

Edge enables Collibra Data Intelligence Cloud to [safely](#) connect to your data sources hosted in an on-premise or cloud environment. It processes the data source information on the Edge site and sends the process results to Collibra Data Intelligence Cloud.

Edge Responsibilities

The ownership of responsibility over the various Edge components is shared between you and Collibra. The diagram below illustrates which components you are responsible for and have control over, and those which belong to Collibra.



Edge components

Edge consists of three main components:

- An Edge configuration page in Collibra Data Intelligence Cloud to create and install Edge sites.
- An Edge integration capability repository that resides on the Collibra Platform and contains all capabilities that can run on an Edge site.
- An [Edge site](#) that is installed close to a data source in the customer's environment, whether it's in the cloud or on the customer's premises.

Integration steps

The following table shows which steps you have to take to set up Edge.

Step	Description	Required permissions
1	Create an Edge site via Collibra Data Intelligence Cloud Settings.	You have a global role with the Manage Edge sites global permission in Collibra Data Intelligence Cloud.
2	Install the Edge site close to the data source you want to access. You can only install an Edge site on a Linux system that meets the necessary system requirements .	You have a global role with the Install Edge sites global permission in Collibra Data Intelligence Cloud.
3	Update the credentials of the Edge site user.	You have a global role with the Connect Edge sites to Collibra global permission in Collibra Data Intelligence Cloud.

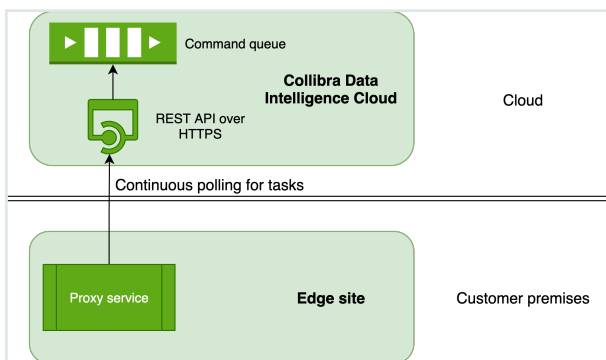
Edge security

Edge is built with security first approach. All communication channels are secured by TLS 1.2 and all endpoints outside Edge are accessible only via authentication. Edge does not send or store any customer data, its purpose is to host capabilities that process the data in its own environment and to send only processing results to Collibra Data Intelligence Cloud.



Communication between Edge and Collibra

Edge operates over an outbound-only model – it executes tasks as commands polled from your Collibra platform. Communication to Collibra uses basic authentication over TLS 1.2. A user account is generated for communicating to Collibra each time the Edge site installer is downloaded. This user account is unique to each Edge site. It is possible to change the password of this user account by following the steps outlined in our [Update Edge user password](#) article.

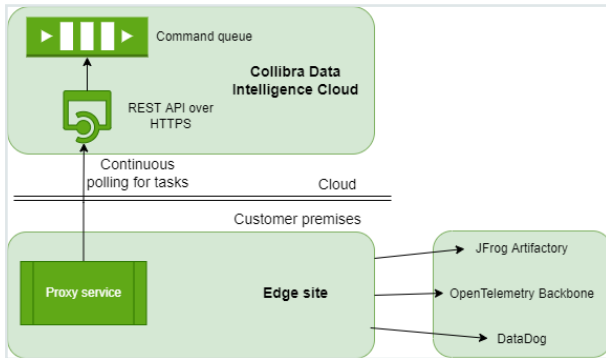


- Edge sites always use REST API endpoints to establish connections.
- Edge requires access to a Collibra server. It is needed for:
 - Reading a request queue, which is a queue with jobs that need to be run on Edge.
 - Returning the metadata results of Edge jobs.
- Edge manages Collibra Data Intelligence Cloud and data source credentials. This has the following consequences:
 - Credentials are not accessible outside of Edge.
 - Credentials used on an Edge site are encrypted with a key that is [secured](#) in Collibra.
 - Credentials of data sources and Collibra can be updated if necessary.
- All configuration parameters, files or strings marked as secret, are stored on the Edge site encrypted with a public key that resides in Collibra. The private part of that key is encrypted with a public key from the Edge site. As a result, secrets can only be decrypted with both key pairs, one residing on the Edge site and the other on Collibra.

- An Edge site communicates over a secure channel with your Collibra environment using certificates, issued by a Collibra-chosen Certificate Authority (CA). However, if there is a forward proxy server between the Edge site and Collibra, you have to use the [proxy server's CA](#).

Communication between Edge and other services

Edge communicates with other servers, such as JFrog, for maintenance purposes.



Edge requires access to the following servers:

Server	Communication	Authentication
JFrog	This is needed in order to download Helm Charts and Docker Images that are running on Edge.	API Key Pair over HTTPS.
OpenTelemetry Backbone	This is needed in order to upload various Edge related metrics.	Basic Authentication.
DataDog	This is needed in order to upload logs from all Edge components: <ul style="list-style-type: none"> • Core edge components • Edge capabilities , for example, ingestion, profiling, lineage, classification, quality. 	API Key Pair over HTTPS.

Authentication to data sources

Edge connections and capabilities use different ways to connect to data sources. The required level of privileges or security greatly depends on the data source type and supported Catalog Connectors.

Collibra regularly adds and certifies Catalog connectors. To understand the authentication methods and the level of security, consult the Catalog connector documentation.

Security scanning

Before Collibra composes an Edge installation package, [XRay scans](#) are performed on all images consumed by Edge to identify and mitigate vulnerabilities. [Contrast scanning](#) is performed post installation for runtime vulnerability detection. This strategy ensures that Edge remains secure.

You can also run your own security scans. We recommend that you run the following command in order to remove old containers and images from an Edge host before running your own scans:

```
sudo /usr/local/bin/k3s crictl rmi --prune.
```

This prune command is a native docker command to clean unused docker objects such as images, containers, volumes and networks. Running this command will avoid false positive vulnerabilities when performing scans as Kubernetes, which is responsible for the garbage control of old Edge images and containers, is not guaranteed to have cleaned up the files before the scan is run.

Storing secrets

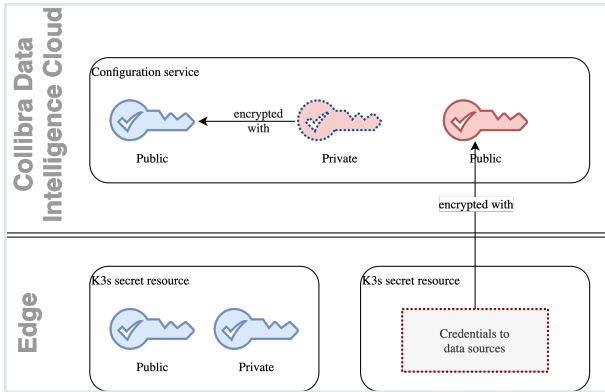
Secrets for connections and capabilities are stored solely on the Edge site. While at rest, secrets are using envelope encryption where the secret is encrypted by a key, which on its turn is encrypted by another key.

The Edge native encryption mechanism is based on two RSA key pairs. They are stored in the following places:

Keys	DIC server	Edge server	Purpose
Public red key	Yes	No	Used to encrypt credentials
Private red key	Yes (encrypted using public blue key)	No	Used to decrypt credentials
Public blue key	Yes	Yes	Used to encrypt red private keys
Private blue key	No	Yes	Used to decrypt red private key

The blue key pair is stored as a Kubernetes secret on the Edgeserver so it undergoes a native K3S encryption as described [here](#).

An Edge site owns the Blue key pair, with the Blue private key stored on Edge. Similar to that, Collibra Data Intelligence Cloud owns the Red key pair. Every secret on Edge is encrypted with the private Red key, which is sent to the Edge site for each capability execution, encrypted with the Blue public key. Once on the Edge site, Red private key is decrypted, and secrets needed to execute a connection or a capability are decrypted and injected into the capability container.



Note Inside the k8s cluster, all other secrets, for example data source credentials and datadog credentials, are stored encrypted at rest.

Customer Credentials

Credentials storage

All sensitive data is stored on Edge and encrypted by the native K3S mechanism. Additionally, all user entered credentials are encrypted using the native Edge encryption mechanism.

Secret encryption

In the case of Virtual Machine or Bare Metal installations (K3S based), all secrets are encrypted using the native Kubernetes mechanism. The whole state of the cluster, including secrets and ConfigMap, are subject to encryption. The encryption algorithm that is used is AES 256 in CBC mode and PKCS#7 padding, which can be checked by running the following command: `sudo /usr/local/bin/k3s secrets-encrypt status`

The entire database is stored in the `/var/lib/rancher/k3s/server/db/state.db` file which contains the SQLite data.

Credential encryption

Every value that is marked as **To be encrypted by Edge management** is additionally encrypted by the Edge site specific red public key.

The algorithm for encryption is summarized below:

1. User enters sensitive text either via Web UI or REST API.
2. Text is sent to aCollibra server (Edgemanagement module).
3. The Edge management module retrieves the red public key for the specific site.
4. A new AES 128 symmetric key (encryption key) is generated.
5. The encryption key is used to encrypt the sensitive text.
6. The encryption key itself is encrypted using the red public key.
7. The encrypted encryption key and encrypted text are concatenated and encoded using Base64 encoding to form the Edgesecret.

8. The Edgesecret is then sent directly to the Edge site, where it is stored as a Kubernetes secret.

In short the algorithms used are:

- RSA 2048 in EBC mode and PKCS#1 padding
- AES 128 in EBC mode and PKCS#7 padding

Note AES 128 was selected due to the restriction of the RSA algorithm which can only encrypt 245 bytes.

Credentials transfer

When the Collibra server (Edge management module) has encrypted the credentials, they are sent to the Edge site using the HTTP TLS 1.2 protocol.

Platform credentials

Apart from the credentials that users need to enter in order to connect to the data sources, there are also credentials which are needed to access the Collibraserver itself.

These credentials include:

- Collibraserver credentials (username and password, stored in dgc-secret Secret)
 - You can rotate these credentials by using the script: `edge update-dgc-cred`
- DataDog API key (stored in datadog-secret Secret)
 - Rotation is currently not possible. You have to reinstall Edge.
- JFrog credentials (stored in collibra-edge-repo-creds Secret)
 - Rotation is currently not possible. You have to reinstall Edge.

For K3S based installations, the JFrog credentials are also stored in file:

`/etc/rancher/k3s/registries.yaml`

Note This file is unencrypted, but it is only accessible by a root user.

Data samples in Edge

By default, Edge by design, doesn't store any samples. To view sample data for data sources registered via Edge, you can activate a sampling capability. For all details, see [Sample data](#).

Edge capabilities such as Profiling and Classification use data in memory, after which the data is discarded.

Edge Cache

Any metadata, logs or metrics stored in the Edge cache are encrypted by default to improve the security of your data and the platform. You are not required to make any changes to this security policy, and there is no impact on the functionality of your Edge sites.

Edge service repository

To keep Edge synchronized with your Collibra Data Intelligence Cloud version, we deploy core Collibra services and business capabilities in the Collibra repository of your environment. An Edge site uses token-based authentication with read privileges to download services for each release. The authentication and endpoint to access the Collibra repository are stored in the **registries.yaml** file as part of the Edge site installer.

For 2-day vulnerability, you can edit **registries.yaml** and access the registry independently, and download images for Edge to scan them. Currently there is no SLA for vulnerabilities that you may find. The standard support SLAs are applied.

Monitoring and logging

We monitor and log all interaction between an Edge site and Collibra Data Intelligence Cloud, as well as the Edge site infrastructure health. All logs are kept in the Collibra Datadog account.

Note We don't send Catalog connector logs to your environment. These Catalog connector logs are by default turned off. If they are enabled, they are kept on the Edge site itself. If you have Catalog connector issues, you have to extract these logs and send them to Collibra Support via a support ticket.

Host hardening on K3S-based integration

Each time you start K3S, a KUBECONFIG file is created. This file contains the credentials to access the K3S cluster as an administrator. The KUBECONFIG file is created by default under `/etc/rancher/k3s/k3s.yaml`. For security reasons, we recommend host hardening by making the KUBECONFIG file inaccessible for other users. As long as the host hardening is applied to Edge, you cannot connect to the K3S cluster using `kubectl` or the Edge tools.

In this article, you will learn how to enable and disable the host hardening.

Prerequisites

- Edge needs to be [installed](#).
- You need root privileges on the server that hosts the Edge site.

Enable host hardening

1. Sign into the server that hosts your Edge site with root privileges.
2. Open the file `/etc/systemd/system/k3s.service.env` for editing.
3. Add the line `K3S_KUBECONFIG_OUTPUT=/dev/null`.

Note If there are other lines, setting other environment variables do not remove them.

4. Restart the K3S service: `systemctl restart k3s`
5. Check if the KUBECONFIG file is empty: `cat /etc/rancher/k3s/k3s.yaml`

Note K3S is actually making `/etc/rancher/k3s/k3s.yaml` a symlink to `/dev/null`.

To further increase the security of your server, you can prevent connections to K3S from other sources than localhost.

Limit the access to the following ports other than localhost:

Protocol	Port	Description
TCP	6443	Kubernetes API Server
TCP	10250	Kubelet metrics
UDP	4500	strongSwan
UDP	500	strongSwan

The following commands prevent access to the ports mentioned in the table. Please check with your security team for compliance and for the tools used to filter the traffic before applying these commands.

```
iptables -I INPUT -j DROP -p tcp -m multiport --dports
6443,10250
iptables -I INPUT -j DROP -p udp -m multiport --dports
4500,500
iptables -I INPUT -j ACCEPT -i lo -p tcp -m multiport --dports
6443,10250
iptables -I INPUT -j ACCEPT -i lo -p udp -m multiport --dports
4500,500
```

Disable host hardening

1. Sign into the server that hosts your Edge site with root privileges.
2. Open the file `/etc/systemd/system/k3s.service.env` for editing.
3. Remove the line `K3S_KUBECONFIG_OUTPUT=/dev/null`.
4. Restart the K3S service: `systemctl restart k3s`
5. Check if the KUBECONFIG file is empty: `cat /etc/rancher/k3s/k3s.yaml`

Installing an Edge site

An Edge site is a component installed in a customer's environment. Each Edge site has a unique identifier and hosts an Edge capability that can access a data source.

This section contains the information that you need to know to install an Edge site.

About an Edge site installation

After [creating your Edge sites](#) in Collibra Data Intelligence Cloud, you have to install the Edge software on either K3S or EKS. You typically [install](#) Edge sites within the same secure environment as the relevant data source. A customer usually has several Edge sites depending on their requirements, for example the number of networks and secure environments, as well as the technical and legal spread of data sources.

An Edge site can have:

- Zero or more predefined connections to data sources via a JDBC driver.
- One or more integration capabilities to process data on site and send the results to Collibra.

An Edge site is a compute runtime on K3S or EKS, that executes capabilities close to your data but that is configurable from the Collibra Data Intelligence Cloud settings. It has a dedicated unique identifier and handles data sources that it can reach within its network. You can have more than one Edge site, depending on the number of networks, security domains, regions or VPCs that you have.

Properties

Property	Description
Name	The name of the Edge site. Use a meaningful name, for example NetherlandsDataCentre1. Do not use spaces or special characters. This field is mandatory and the name must be globally unique.
Status	The status of the Edge site. The status is automatically shown when you create an Edge site.
ID	The unique ID of the Edge site, which is generated automatically when you create the Edge site .

Property	Description
Description	The description of the Edge site. We recommend to put at least basic location information of the Edge site. This field is mandatory.
Installer and property files	A section where you can download the installer and property files to install an Edge site on a server. This section is only visible when the Edge site has the status To be installed .

Statuses

The status of an Edge site indicates if the Edge site can be used or not. The status is shown on the **Edge** settings page of the [Collibra settings](#). An Edge site can have one of the following statuses:

Status	Description
To be installed	The Edge site is created, but not installed yet.
Offline	Collibra cannot reach the Edge site. This can be caused by an unsuccessful installation or a lost connection. See the installation logs for more information.
Unhealthy	Collibra can connect to the Edge site, but some functions don't work correctly. This is typically caused by problems during the installation. See the installation logs for more information.
Healthy	The Edge site installation was successful.

Installation directories on K3S

The Edge site installer installs files in the following directories on your host server:

- `/var/lib/rancher/`
- `/var/log/`

- `/etc/`
- `/usr/local/bin/`

System requirements of an Edge site

To use [Edge](#), you must ensure that the following system requirements are met.

Software requirements

- You must be able to install the Edge software on the latest version of RedHat Enterprise Linux 8.
- The `sudo` package is installed on the Linux host.
- The user who installs Edge has full sudo access (`ALL=(ALL) ALL`)
- Optionally, if you want SE Linux enabled, you will need to install the following policy packages before installing Edge:
Packages¹

Tip If you are an early adopter or you use Edge for beta testing purposes, we highly recommend to [disable SELinux](#).

Hardware requirements

You need the following minimum hardware requirements:

- 64 GB memory
- 16 core CPU with x86_64 architecture
- You have at least 50 GB of free storage on the partition that must contain `/var/lib/rancher/k3s`. The partition mountpoint should not have the `noexec` option.

1

- `yum install -y container-selinux selinux-policy-base`
- `yum install -y https://rpm.rancher.io/k3s/stable/common/centos/7/noarch/k3s-selinux-0.2-1.el7_8.noarch.rpm`

These packages are not hosted by Collibra. If you have any questions, please reach out to your internal teams.

```
mkdir -p /var/lib/rancher/k3s
mkfs.xfs /dev/<block-device-name>
mount /dev/<block-device-name> /var/lib/rancher/k3s
echo '/dev/<block-device-name> /var/lib/rancher/k3s xfs
defaults 0 0' >> /etc/fstab
```

Note This is the default install path. If it is not created as a separate mount point following the steps above, the install will use 50 GB of disk space from either `/var`, or if not present, the root level of the drive.

Warning Any data in this location is fully managed by the Edge site, do not save any other data in here as it can be removed by Edge without notification.

- You have mounted at least 500 GB of dedicated storage for the Edge site data on a freely chosen mountpoint, for example `/var/edge/storage`.

```
mkdir -p /var/edge/storage
mkfs.xfs /dev/<block-device-name>
mount /dev/<block-device-name> /var/edge/storage
echo '/dev/<block-device-name> /var/edge/storage xfs
defaults 0 0' >> /etc/fstab
```

Note Change `<block-device-name>` to the name of the device that contains the storage.

Warning This dedicated storage should not be shared with other services because Edge can delete and overwrite files on this location without notice, so don't use `/home/<username>` or `/var`.

- If you run the Linux server on AWS or GCP, disable the services **nm-cloud-setup.service** and **nm-cloud-setup.timer**.

```
systemctl disable nm-cloud-setup.service nm-cloud-
setup.timer
reboot
```

Warning When new capabilities are added in the future, the hardware requirements may change.

Network requirements

- An Edge site needs outbound connections to all of the following:
 - The URL of your Collibra Data Intelligence Cloud environment.
 - `https://*.datadoghq.com`

Note If wildcard whitelisting is not accepted:

- `https://http-intake.logs.datadoghq.com`
- `https://7-40-1-app.agent.datadoghq.com`
- `https://agent-http-intake.logs.datadoghq.com`
- `https://api.datadoghq.com`

- `https://*.repository.collibra.io`

Note If wildcard whitelisting is not accepted:

- `https://repository.collibra.io`
- `https://edge-docker-delivery.repository.collibra.io`
- `https://mirror-docker.repository.collibra.io`

- `https://otlp-http.observability.collibra.dev/`
- Access to all data sources you need to connect to your Edge sites.
- Your Edge site has to be able to connect to port 443.

- The resolve configuration file of your Linux host has maximum three search domains and two name servers.

Note If a firewall is enabled, run the following commands to add the `cni0` and loopback interfaces to a trusted zone, so that Kubernetes can use it between its services:

```
firewall-cmd --zone=trusted --change-interface=cni0 --
permanent
firewall-cmd --zone=trusted --change-interface=lo --
permanent
firewall-cmd --reload
```

EKS requirements

You can install the Edge software on managed Kubernetes clusters.

Important A managed Kubernetes cluster must be fully dedicated for Edge, do not use the cluster for other purposes.

- AWS EKS 1.21, 1.22, 1.23 (all only with `--container-runtime containerd`) and 1.24 are supported.
 - We support EBS-CSI driver for 1.23.
- AWS EKS worker nodes use the EKS optimized Amazon Linux 2 AMI
- EKS cluster has [IRSA enabled](#)
- AWS EKS worker nodes need to be in the same (one) Availability Zone! This can be implemented by creating just one node group for the EKS cluster, which limits the subnets to just one subnet, that is one of the subnets of the EKS cluster.
- Set up security groups to ensure that worker nodes can communicate with each other on non-privileged ports.

```

module "eks" {
  source          = "terraform-aws-modules/eks/aws"
  version        = "17.24.0"
  cluster_name   = "${var.vpc_name}-${var.cluster_
name}-eks"
  cluster_version = "1.21"
  vpc_id         = var.vpc_id
  subnets       = data.aws_subnet_ids.public_subnet_
ids.ids # Subnets specified must be in at least two
different AZs
  worker_additional_security_group_ids = [aws_security_
group.worker_sg.id]
  enable_irsa    = true # enable iam role for service
account, for later use
  worker_groups = [
    {
      name              = "${var.vpc_name}-${var.cluster_
name}-eks-workers"
      instance_type     = var.worker_type
      asg_desired_capacity = var.instance_count_workers
      key_name          = aws_key_pair.cluster-ssh-
keypair.key_name
      bootstrap_extra_args = "--container-runtime
containerd" # mandatory to run with containerd if on
1.21
      subnets          = [subnet1]
      # restriction for now to use only 1 subnet due to
EBS tied to AZ
    },
  ]
  map_accounts = [
    data.aws_caller_identity.current.account_id
  ]

  tags = {
    Name              = "${var.vpc_name}-${var.cluster_
name}-eks"
  }
}

```

Software requirements

- A Linux server with bash available. This is the server from which you install the Edge software on EKS.

Tip This server will also contain the Edge tools.

- Plain `cluster_admin` `kubectl` access to the EKS cluster using its `kubeconfig`. With this `kubeconfig`, you must be able to use the `kubectl` command to communicate with the Kubernetes API server with full cluster access.
- Ensure your `Kubectl` client is compatible with the relevant EKS version.

Hardware requirements

You need an operational EKS cluster with at least 1 worker node. The cluster must meet the following requirements:

- The total cluster capacity has at least 16 core CPU and 64 GB memory, for example 4 worker nodes with each 4 core CPU and 16 GB.
- Each worker node needs at least 100 GB free disk space to store Docker images.
- Ability to create [EBS](#)-based persistent volumes as a default storage class, at least 500 GB in total.

Network requirements

- An Edge site needs outbound connections to all of the following:
 - The URL of your Collibra Data Intelligence Cloud environment.
 - `https://*.datadoghq.com`

Note If wildcard whitelisting is not accepted:

- `https://http-intake.logs.datadoghq.com`
- `https://7-40-1-app.agent.datadoghq.com`
- `https://agent-http-intake.logs.datadoghq.com`
- `https://api.datadoghq.com`

- https://*.repository.collibra.io

Note If wildcard whitelisting is not accepted:

- <https://repository.collibra.io>
- <https://edge-docker-delivery.repository.collibra.io>
- <https://mirror-docker.repository.collibra.io>

- <https://otlp-http.observability.collibra.dev/>
- Access to all data sources you need to connect to your Edge sites.
- Your Edge site has to be able to connect to port 443.
- The resolve configuration file of your Linux host has maximum three search domains and two name servers.

Note If a firewall is enabled, run the following commands to add the `cni0` and loopback interfaces to a trusted zone, so that Kubernetes can use it between its services:

```
firewall-cmd --zone=trusted --change-interface=cni0 --
permanent
firewall-cmd --zone=trusted --change-interface=lo --
permanent
firewall-cmd --reload
```

Create an Edge site



As jobs are run on an Edge site, rather than on the Collibra platform, creating an [Edge site](#) allows you to have a processing runtime at your own premises.

Prerequisites

- You have a global role that has the System administration global permission.
- You have a global role that has the Manage Edge sites global permission.
- You have [enabled](#) database registration via Edge in Collibra Console.

Note You must restart the Data Governance Center service when you have enabled this option.

Steps

1. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
2. Click **Edge**.
 - » The Edge sites overview appears.
3. Above the table, to the right, click **Create Edge site**.
 - » The **Create Edge site** wizard starts.
4. Enter the required information.

Field	Description
Name	The name of the Edge site. Use a meaningful name, for example NetherlandsDataCentre1. Do not use spaces or special characters. This field is mandatory and the name must be globally unique.
Description	The description of the Edge site. We recommend to put at least basic location information of the Edge site. This field is mandatory.

5. Click **Create**.

- » The Edge sites overview appears, including the new Edge site with the status **To be installed**.

What's next?

You can now [install the Edge site](#), or if necessary, first [configure a forward proxy](#).

Install an Edge site

After you have created the [Edge site](#) in Collibra Data Intelligence Cloud, you have to install the Edge software on a server.



Tip

Every time you download an Edge site installer, the previously downloaded Edge site installer becomes outdated. If you use this outdated installer, the Edge site cannot communicate with Collibra.

Prerequisites

- You have a global role with the Install Edge sites and the User Administration global permission, for example Edge site administrator.
- You have a global role that has the System administration global permission.
- You have [created](#) an Edge site.
- You have [configured the forward proxy](#), if a forward proxy is required for Edge to connect to Collibra, Datadog, OpenTelemetry and jFrog. Contact your network administrator if this is applicable.
- Your server meets all [system requirements](#).

Steps

1. Download the installer:
 - a. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.

- b. Click **Download** in the **Installer and properties files** section.

Tip When you download the installer, [an Edge user is automatically created](#) in Collibra.

- c. Depending on your operating system and browser, follow the regular steps for downloading files.

» The installer file is a TGZ archive that contains the files **proxy.properties**, **properties.yaml** and **registries.yaml**.

Warning If you download an installer, the previously downloaded Edge site installer becomes invalid.

2. Extract the TGZ archive on the server on which you are going to install the Edge site software.

```
tar -xf <edge-site-id>-installer.tgz
```

Tip Keep the installer or the content of the extracted installer somewhere safe on your server. It contains various tools that you may need later, for example to troubleshoot issues.

3. Execute the installation. Use the correct path to the mounted storage as described in the [prerequisites](#).

Important

- If the Edge site has to connect via a forward HTTP proxy, then first [configure the forward proxy](#) before executing the installation.

a. Clean installation:

```
sudo sh install-master.sh --storage-path
/path/mounted/storage properties.yaml -r
registries.yaml
```

for example:

```
sudo sh install-master.sh --storage-path
/var/edge/storage properties.yaml -r registries.yaml
```

b. Installation with classification enabled:

```
sudo sh install-master.sh --storage-path
/path/mounted/storage properties.yaml -r
registries.yaml --set collibra_
edge.collibra.classification.enabled=true
```

for example:

```
sudo sh install-master.sh --storage-path
/var/edge/storage properties.yaml -r registries.yaml --
set collibra_edge.collibra.classification.enabled=true
```

» In the Edge sites overview, you can see the [status](#) of the deployment.

4. Run the following commands to verify the status of the installation.

- To ensure that Kubernetes is running and that there is an existing node:

```
sudo /usr/local/bin/kubectl get nodes
```

- To ensure the state of all pods are installed and running:

```
sudo /usr/local/bin/kubectl get pods --all-namespaces
```

Tip If you have already installed the Edge site and you want to enable classification afterwards, see [this article](#).

1. Download the installer:

- a. Open an Edge site.
 - a. On the main menu, click ☰, and then click ⚙️ **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
- b. Click **Download the Installer and properties files** section.

Tip When you download the installer, an Edge user is automatically created in Collibra.

- c. Depending on your operating system and browser, follow the regular steps for downloading files.
 - » The installer file is a TGZ archive that contains the files **proxy.properties**, **properties.yaml** and **registries.yaml**.

Warning If you download an installer, the previously downloaded Edge site installer becomes invalid.

2. Extract the TGZ archive on the server on which you are going to install the Edge site.

```
tar -xf <edge-site-id>-installer.tgz
```

Tip Keep the installer or the content of the extracted installer somewhere safe on the workstation that has access to the EKS cluster. It contains various tools that you may need later, for example to troubleshoot issues.

3. Execute the installation.

a. Clean installation:

```
./run-installer-job.sh properties.yaml --set collibra_
edge.collibra.ozone.datanode.pvcSize=100Gi --set
collibra_edge.collibra.ozone.om.pvcSize=100Gi
```

b. Installation with classification enabled:

```
./run-installer-job.sh properties.yaml --set collibra_
edge.collibra.ozone.datanode.pvcSize=100Gi --set
collibra_edge.collibra.ozone.om.pvcSize=100Gi --set
collibra_edge.collibra.classification.enabled=true
```

» In the Edge sites overview, you can see the [status](#) of the installation.

4. Run the following commands to verify the status of the installation.

- To ensure that Kubernetes is running and that there is an existing node:

```
kubectl get nodes
```

- To ensure the state of the installation is either running or finished:

```
kubectl get pods --all-namespaces
```



Tip If you have already installed the Edge site and you want to enable classification afterwards, see [this article](#).

Configure a forward proxy

For security reasons, it is possible that an [Edge site](#) has to connect cloud services via a forward HTTP proxy. In that case, you have to update **proxy.properties** before installing the Edge site.

If the forward proxy server is responsible for decrypting TLS traffic, you also have to use the proxy server's CA certificate during the installation.

Steps

1. Download the Edge site installer:
 - a. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
 - b. In the **Installer and properties files** section, click **Download**.
 - c. Depending on your operating system and browser, follow the regular steps for downloading files.
 - » The installer file is a TGZ archive that contains the files **proxy.properties**, **properties.yaml** and **registries.yaml**.

Note If you download an installer, all previously downloaded installers become invalid.

2. Open the **proxy.properties** file.
3. Uncomment and update the outbound-proxy properties by removing "#" at the beginning of the following lines:

```
#noProxy=<host IP addresses>,<host DNS names>,<k8s-svc-ip-
addresses>,<k8s-pod-ip-addresses>,<others>
#proxyHost=<proxy domain name or IP address>
#proxyPort=<proxy-port>
#proxyUsername=<proxy username>
#proxyPassword=<proxy password>
```

Setting	Value
noProxy	<p>A comma-separated list of IP or DNS addresses that can bypass the proxy server.</p> <p>This list must include at least the Kubernetes cluster's internal IP addresses and the Kubernetes nodes' IP and DNS addresses.</p> <p>The list may not contain spaces.</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>Tip To get the values for this setting, you can use the edge-get-noproxy.sh script, which you can find in the extracted installer directory under /resources/tools. However, make sure that your network administrator reviews these values.</p> </div> <p>where:</p> <ul style="list-style-type: none"> ◦ <host-ip-addresses>: for example 172.20.0.0/16. ◦ <host-dns-names>: for example *.compute.internal. ◦ <k8s-svc-ip-addresses>: is by default 10.43.0.0/16, but this can differ for other k8s flavors or configurations. ◦ <k8s-pod-ip-addresses>: is by default 10.42.0.0/16, but this can differ for other k8s flavors or configurations. ◦ <others>: other IP addresses that don't need to be proxied. Add at least 169.254.169.254. for AWS. <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>Example noProxy=172.20.0.0/16,*.compute.internal,10.43.0.0/16,10.42.0.0/16,169.254.169.254</p> </div>

Setting	Value
proxyHost	<p>The IP or DNS address of the proxy server.</p> <pre>Example proxyHost=site4-proxy.shared.edge.collibra.dev</pre>
proxyPort	<p>The TCP port of the proxy server. This value must be a quoted string and not an integer value.</p> <pre>Example proxyPort="3128"</pre>
proxyUserName	<p>The username to authenticate at the proxy server.</p> <pre>Example proxyUsername=edge</pre> <p>Note Usernames with single quotations ', double quotations ", and backslashes \ need to be escaped using an additional backslash. For example, if the username is ge'smith\, it would need to be entered into proxy.properties as username: ge\'smith\\.</p>
proxyPassword	<p>The password to authenticate at the proxy server.</p> <pre>Example proxyPassword="la;fs90jpo4j3rR%"</pre> <p>Note Passwords with single quotations ', double quotations ", and backslashes \ need to be escaped using an additional backslash. For example, if the password is te"st\1234', it would need to be entered into proxy.properties as password: te\'st\\1234\\.</p>

```
#noProxy=<host IP addresses>,<host DNS names>,<k8s-svc-ip-addresses>,<k8s-pod-ip-addresses>,<others>
```

```
#proxyHost=<proxy domain name or IP address>
#proxyPort=<proxy-port>
#proxyUsername=<proxy username>
#proxyPassword=<proxy password>
```

Setting	Value
noProxy	<p>A comma-separated list of IP or DNS addresses that can bypass the proxy server.</p> <p>This list must include at least the Kubernetes cluster's internal IP addresses and the Kubernetes nodes' IP and DNS addresses.</p> <p>The list may not contain spaces.</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>Tip To get the values for this setting, you can use the <code>edge-get-noproxy.sh</code> script. However, make sure that your network administrator reviews these values.</p> </div> <p>where:</p> <ul style="list-style-type: none"> ◦ <code><host-ip-addresses></code>: for example <code>172.20.0.0/16</code>. ◦ <code><host-dns-names></code>: for example <code>*.compute.internal</code>. ◦ <code><k8s-svc-ip-addresses></code>: depends on your EKS installation. Typically this is <code>10.100.0.0/16</code> or <code>172.20.0.0/16</code>. ◦ <code><k8s-pod-ip-addresses></code>: depends on your EKS installation. Typically they are the same subnets as in the VPC, for example <code>172.20.0.0/16</code>. ◦ <code><others></code>: other IP addresses that don't need to be proxied, for EKS, always add <code>169.254.169.254..</code> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>Example <code>noProxy=172.20.0.0/16,*.compute.internal,10.100.0.0/16,169.254.169.254</code></p> </div>
proxyHost	<p>The IP or DNS address of the proxy server.</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>Example <code>proxyHost=site4-proxy.shared.edge.collibra.dev</code></p> </div>

Setting	Value
proxyPort	<p>The TCP port of the proxy server. This value must be a quoted string and not an integer value.</p> <p>Example <code>proxyPort=3128</code></p>
proxyUsername	<p>The username to authenticate at the proxy server.</p> <p>Example <code>proxyUsername=edge</code></p> <p>Note Usernames with single quotations ', double quotations ", and backslashes \ need to be escaped using an additional backslash. For example, if my username is ge'smith\, it would need to be entered into proxy.properties as username:ge\'smith\\.</p>
proxyPassword	<p>The password to authenticate at the proxy server.</p> <p>Example <code>proxyPassword=la;fs90jpo4j3rR%</code></p> <p>Note Passwords with single quotations ', double quotations ", and backslashes \ need to be escaped using an additional backslash. For example, if my password is te"st\1234', it would need to be entered into proxy.properties as password:te\'st\\1234\'.</p>

Important When you add a new node to a cluster, review and update, if necessary, the noProxy and implicitly forward proxy settings, unless the subnet used for nodes and their DNS suffix are added to noProxy.

- If you use a forward proxy that decrypts TLS traffic, a so-called man-in-the-middle proxy or MITM proxy, then on-the-fly TLS certificates that are generated by the MITM proxy must use the subjectAltName (SAN) extension. To enable Edge via a MITM proxy, perform also the following steps.
If the proxy server does not decrypt the TLS traffic, you can skip the following steps.

- a. Export your proxy server's CA certificate in PEM format.
- b. Save this certificate as **ca.pem** in the same directory as the Edge site installer.

Note If you save the certificate in another directory, use the `--ca` argument in the [Edge site installation command](#).

What's next?

[Install](#) the Edge site.

If you want to update the forward proxy afterwards, you can use the [update script](#).

Enable or disable classification on an Edge site

If you have an existing Edge site installation without [classification](#), you can enable it afterwards. Similarly, you can disable classification on installation where it is enabled.

Note Enabling or disabling classification can take a couple of minutes before the changes are in effect.

Note The following commands have been updated with the 2023.02 release. While these commands will work for all previous and new Edge site versions, the old commands will not work with any newly installed sites with version 2023.02 or later.

Enable classification

To enable classification on an existing Edge site, deployed on K3S, run this command:

```
POD_NAME=$(sudo /usr/local/bin/kubectl get pod -n collibra-edge -l app.kubernetes.io/component=application-controller -o name)

sudo /usr/local/bin/kubectl -n collibra-edge exec -it ${POD_NAME} \
-- bash -c 'argocd admin cluster kubeconfig https://kubernetes.default.svc \
/tmp/config --namespace collibra-edge ; env KUBECONFIG=/tmp/config argocd app set collibra-edge --core -p collibra.classification.enabled=true'
```

To enable classification on an existing Edge site, deployed on EKS, run this command:

```
POD_NAME=$(kubectl get pod -n collibra-edge -l app.kubernetes.io/component=application-controller -o name)
```

```
kubectl -n collibra-edge exec -it ${POD_NAME} \
  -- bash -c 'argocd admin cluster kubeconfig
https://kubernetes.default.svc \
  /tmp/config --namespace collibra-edge ; env
KUBECONFIG=/tmp/config argocd app set collibra-edge --
core -p collibra.classification.enabled=true'
```

Disable classification

To disable classification on an existing Edge site, deployed on K3S, run this command:

```
POD_NAME=$(sudo /usr/local/bin/kubectl get pod -n collibra-
edge -l app.kubernetes.io/component=application-controller -
o name)

sudo /usr/local/bin/kubectl -n collibra-edge exec -it
${POD_NAME} \
-- bash -c 'argocd admin cluster kubeconfig
https://kubernetes.default.svc \
  /tmp/config --namespace collibra-edge ; env
KUBECONFIG=/tmp/config argocd app set collibra-edge --
core -p collibra.classification.enabled=false'
```

Tip The only difference between disabling classification and enabling classification is that the last argument is false instead of true.

To disable classification on an existing Edge site, deployed on EKS, run this command:

```
POD_NAME=$(kubectl get pod -n collibra-edge -l
app.kubernetes.io/component=application-controller -o name)

kubectl -n collibra-edge exec -it ${POD_NAME} \
  -- bash -c 'argocd admin cluster kubeconfig
https://kubernetes.default.svc \
```

```
/tmp/config --namespace collibra-edge ; env  
KUBECONFIG=/tmp/config argocd app set collibra-edge --  
core -p collibra.classification.enabled=false'
```

Tip The only difference between disabling classification and enabling classification is that the last argument is false instead of true.

Successful execution of either command returns the following output:

```
INFO[0000] Starting configmap/secret informers  
INFO[0000] Configmap/secret informer synced
```

Note You do not need to restart Edge when you have enabled or disabled classification.

Reinstall an Edge site using backup and restore

You can reinstall an Edge site by keeping previous Edge site data.

1. **Back up** your current Edge site (optionally): `edge backup -o backup.yaml`
 2. If you are reusing the same server as your current Edge site, **uninstall** the current installation: `uninstall-edge.sh`
 3. **Download** the new installer and save it on your server.
 4. Extract the installer and ensure that your custom setup, for example **proxy.properties** and **ca.pem**, is available as in the previous setup.
 5. Reinstall with the new installer, optionally with backup and/or classification:


```
./install-master.sh --storage-path <storagepath> -b backup.yaml
--set collibra_edge.collibra.classification.enabled=true
```
-
1. **Back up** your current Edge site (optionally): `edge backup -o backup.yaml`
 2. If you are reusing the same server as your current Edge site, **uninstall** the current installation: `<extracted_installer>/resources/tools/installer-job/uninstall-edge-on-managed-k8s.sh`
 3. **Download** the new installer and save it on your Linux server that has `kubectl` access to the k8s cluster.
 4. Extract the installer and ensure that your custom setup, for example **proxy.properties** and **ca.pem**, is available as in the previous setup.
 5. Reinstall with the new installer: `./run-installer-job.sh properties.yaml -set collibra_edge.collibra.ozone.datanode.pvcSize=100Gi --set collibra_edge.collibra.ozone.om.pvcSize=100Gi`
 - Optionally, install with
 - **Backup:** `-b backup.yaml`
 - **Backup with classification:** `-b backup.yaml --set collibra_edge.collibra.classification.enabled=true`

Upgrade the operating system of an Edge site

When you have a running Edge site, you can safely upgrade the operating system by following the procedure in this article.

Steps



1. [Back up](#) the Edge site.

Note The backup is not mandatory, but highly recommended in case the upgrade of your OS would fail.

2. Upgrade your OS.
3. Restart the OS.
4. Wait until the Edge site becomes healthy in the Collibra Data Intelligence Cloud user interface.

Troubleshooting

If the Edge site does not become healthy after the OS upgrade, then [reinstall](#) the Edge site with a new Edge installer and the backup that you created before the OS upgrade.

1. In Collibra, go to the Edge site you want to reinstall.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. In the Edge site overview, click the name of an Edge site.
 - » The Edge site page appears.
2. In the top right corner, click **Actions** → **Redownload installer**.
 - » A new Edge installer is downloaded.

3. Install the Edge site with the backup that you created earlier.

```
install-master.sh properties.yaml --storage-path  
/var/edge/storage properties.yaml -r registries.yaml -b  
/<path to backup file>/edge-backup.yaml
```

4. Wait until the Edge site becomes healthy in the Collibra Data Intelligence Cloud user interface.

Enable Manual Upgrade for Edge sites

Important This is a [beta feature](#).

The Manual Upgrade feature allows you to choose when and to which version you want to upgrade your Edge sites. Whenever a new version of Edge becomes available, a banner is displayed at the top of the page with an **Upgrade Now** button. After you select the version to which you want to upgrade your site, you can download the Software Bill of Materials to review and scan before beginning the upgrade.

Note Upgrades must be performed within 3 months of Collibra's quarterly releases. For more information, go to [Upgrade an Edge site](#).

New Edge sites

Automatic upgrades are enabled by default for all new Edge sites. If you do not change this, all Edge sites will automatically upgrade when a new version has been detected. To control when your Edge site upgrades, select **Manual** under the **Upgrade Mode** section when creating your new Edge site.

Sites ▶ Create site

Create Site

Name *

Please provide a unique name.

Description *

Please provide a meaningful description of the site including i.e. data center name, location, necessary connection and capabilities.

Upgrade Mode

Automatic

Manual

Edge Version



2023.02.0-main.75

[Download Bill Of Materials for 2023.02.0-main.75](#)

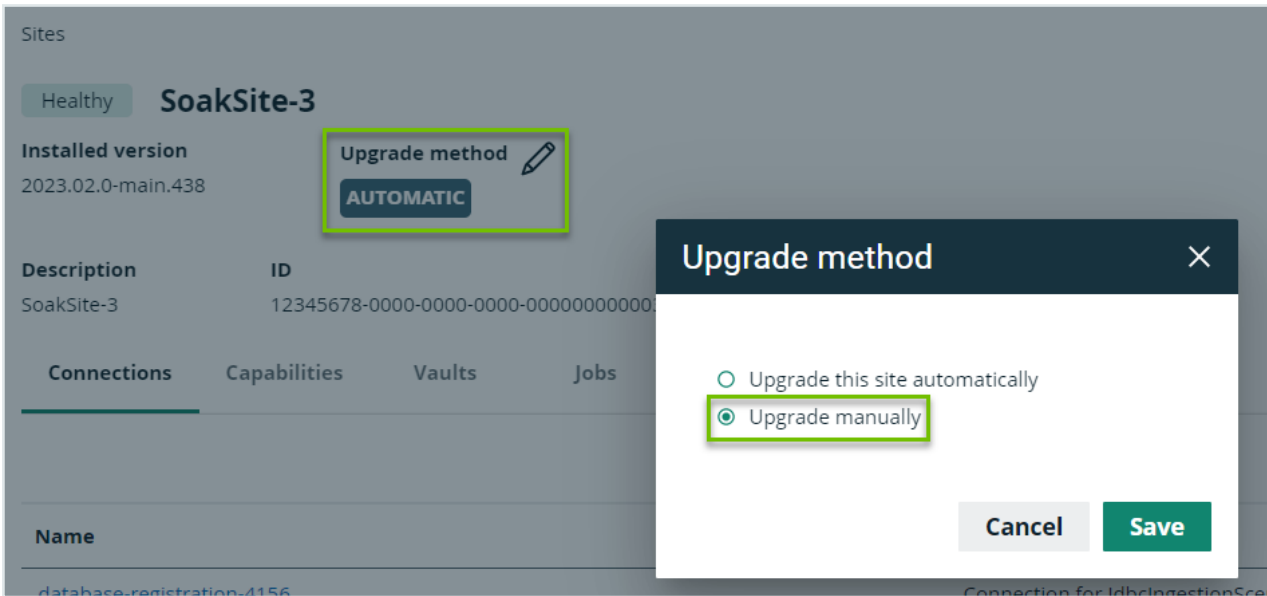
Existing Edge sites

You can change the upgrade method to manual for existing Edge sites by following the steps below:

Steps

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. Click the pencil icon next to **Upgrade method**.
 - » The Upgrade method wizard opens.

3. Select **Upgrade manually**.
4. Click **Save**.



Your Edge site will no longer automatically upgrade to the newest available version.

Upgrade an Edge site



You will receive a notification whenever a new version of Edge is available. There are two types of upgrades: optional and mandatory. Optional upgrades are the minor updates which occur between quarterly releases. Mandatory upgrades are the major releases, which include new features and enhancements. A mandatory upgrade is required within 3 months of the Collibra Data Intelligence Cloud quarterly release. You will be notified as to whether an upgrade is optional or mandatory.

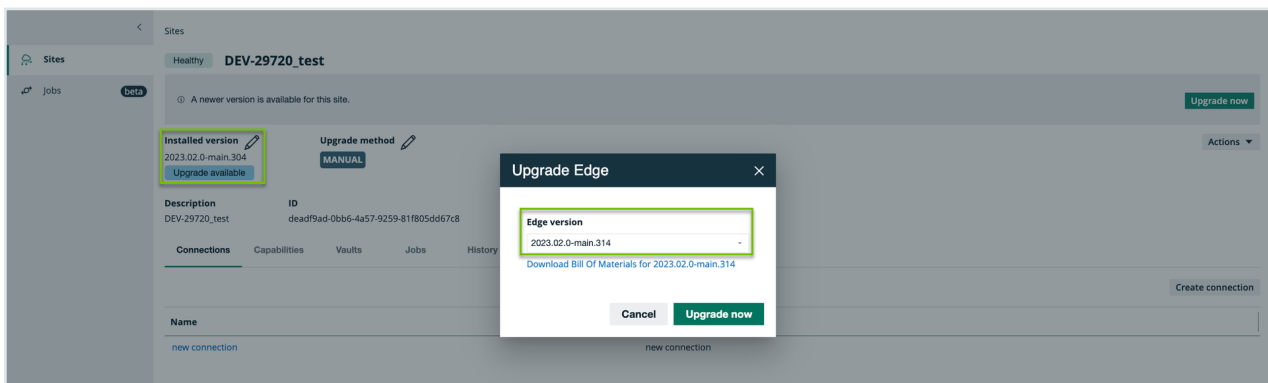
When a mandatory upgrade becomes available and you have manual upgrades enabled for an Edge site, your site will be in read-only mode until you upgrade the site to the mandatory version. For more information, go to the [Version compatibility](#) section below. This is to ensure that all Edge features are appropriately updated and compatible with Collibra.

Note You cannot start any capabilities or configure any connections, capabilities or vaults if your Edge site is in read-only mode. You must perform the mandatory upgrade or wait until an upgrade has been completed to resume full access to Edge.

You can either upgrade to the newest version by clicking **Upgrade now** on the Edge site page or manually select an available version by following the steps below:

Steps

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. Click the pencil icon next to **Installed version**.
 - » The Upgrade Edge wizard opens.
3. Click the dropdown to review available Edge versions.
4. Select the version from the dropdown you want to review or upgrade to.
5. Optional: Click the hyperlink to download the Software Bill of Materials.
6. Click **Upgrade now**.



Software Bill of Materials

You can download a Software Bill of Materials (SBOM) for a version of Edge to review its contents. A SBOM is a list of images included in a version of Edge on which your security team may want to perform security scans and evaluations before your Edge site is upgraded to a new version.

For more information about Edge security and scanning, go to [Security scanning](#).

You can retrieve the SBOM through one of the following methods:

- A REST API.
 - **Location:** `<hostname>/edge/api/rest/v2/releaseinfo/<edge version>`
- Selecting an upgrade version in the Edge platform.
 - When you select a version to upgrade your Edge site to, you are provided with a link to download the SBOM, as shown in the above [Upgrade an Edge site](#) section.

The SBOM is downloaded as a zip file containing JSON files. These are in SPDX and CYCLONEDX formats which you can use as input files for your security scanning tools.

Version compatibility

The following table is an example of the Collibra Data Intelligence Cloud and Edge versions. The mandatory releases are indicated in the bold formatting.

Collibra version	Edge version	Minimum Edge version supported for Collibra	Maximum Edge version supported for Collibra	Actively supported Edge versions
2023.11 (Q)	2023.11.x	2023.08.x	2023.11.*	<ul style="list-style-type: none"> • 2023.08 • 2023.11
2023.12	2023.11.x+1	2023.08.x	2023.11.*	<ul style="list-style-type: none"> • 2023.08 • 2023.11
2024.01		2023.08.x	2023.11.*	<ul style="list-style-type: none"> • 2023.08 • 2023.11

Collibra version	Edge version	Minimum Edge version supported for Collibra	Maximum Edge version supported for Collibra	Actively supported Edge versions
2024.02 (Q)	2024.02.x	2203.11.x	2024.02.*	<ul style="list-style-type: none"> • 2023.11 • 2024.02
2024.03	2024.02.x+1	2203.11.x	2024.02.*	<ul style="list-style-type: none"> • 2023.11 • 2024.02
2024.04	2024.02.x+2	2203.11.x	2024.02.*	<ul style="list-style-type: none"> • 2023.11 • 2024.02

If you are using a version of Edge that is no longer compatible with Collibra, you will receive a notification that there is a required Edge upgrade. Your site will be in read-only mode and your capabilities will not start until the upgrade has been completed.

What's next?

You can perform your own [security scans](#) before upgrading to a new version of Edge if you set up a [private docker registry](#).

Private Docker Registry

You can use your own infrastructure to serve docker images consumed by Edge by setting up a private docker registry. This allows you to perform your own [security scans](#) and auditing when you use the [Manual Upgrade](#) feature.

Note Switching to a private docker registry is only possible during installation.

Approaches

There are two approaches you can choose when implementing a private docker registry. You can either use registry mirroring or manually pull images from Collibra's docker registry and push images to your private docker registry. Both options are outlined below.

Approach 1: Use registry mirroring

Collibra internally uses JFrog's Artifactory, which is a repository manager that allows for dynamic mirroring of docker registries, to manage our repository. If your company has their own JFrog Artifactory, you can configure it to automatically mirror images from Collibra's Artifactory.

This method is easy to set up and supports automatic updates. However, it has limited options for security scanning.

Requirements and permissions

- You must have admin access to JFrog Artifactory.
- You have the Edge site administrator global role.

Steps

Note When configuring a private Artifactory docker registry, the registry must follow the "subdomain first" method for pulling images. For example, <repository-key>.artifactory.my.org.

1. From the **Administration** pane, in the **Repositories** menu, click **Repositories**.
2. Click the **Remote** tab.
3. Click **New Remote Repository**.
4. In the **Select Package Type** section, select **Docker**.
5. Name the registry using the repository key of your choice.
6. Set the URL to: *edge-docker-delivery.repository.collibra.io*
7. Enable remote authentication.

Note The username and password should be the same as those found in the installer file **registries.yaml**.

8. Click **Save and finish**.
9. Install the new or reinstall the existing Edge site using the full private registry URL. This will be the private Artifactory hostname and repository key from step 4.

Note You can set the upgrade method to either automatic or manual, however, the manual upgrade method will take longer to mirror.

- a. For new Edge sites:
 - i. Create a new Edge site.
 - ii. Select the latest version of Edge.
 - iii. Download the installer.
- b. For existing Edge sites:
 - i. Upgrade to the latest version of Edge.
 - ii. Create a **backup and reinstall** the Edge site.
- c. The installer supports the following installation parameters:
 - `--registry-url`, **example**: `https://edge-docker-delivery.repository.collibra.io`
 - `--registry-host`, **this is optional and is automatically derived from** `--registry-url`, i.e., `edge-docker-delivery.my-registry-docker.io`
 - `--registry-pass`, **not required if registry is public.**
 - `--registry-user`, **not required if registry is public.**
- d. In order to install Edge with support for a private docker registry, the following set of additional parameters is required:
 - **For Bare metal installations:** `./install-master.sh (...usual parameters...) --registry-url https://private-`


```
docker.registry.com --registry-user user --registry-
pass pass
```

- **For EKS installations:** `./run-installer-job.sh (...usual parameters...) --registry-url https://private-docker.registry.com --registry-user user --registry-pass pass`

Note The Registry user should have read/pull permissions for the docker registry. This should be validated manually prior to installing Edge.

Approach 2: Pull images from Collibra's docker registry and push them to your registry

This approach works with any docker registry and is more flexible, as it allows you to use any security and scanning tool. However, it only supports manual updates, and you must be familiar with manually pushing and pulling images using a docker client.

Requirements and permissions



- You have the Edge site administrator global role.
- You have access to a private docker registry with Write permissions.
 - We recommend that you have an isolated docker registry for storing Edge related images.
- You have a computer with the docker client installed.

Steps

1. Authenticate with the Collibra Docker Registry.
 - a. In your Edge site installer, find the `registries.yaml` file, which contains credentials to download an installer.
 - b. Run the following command with the `username` and `password` from the `registries.yaml` file: `docker login edge-docker-deliv-`

```
ery.repository.collibra.io -u <username> -p <password>
```

Note Docker credentials are read-only.

2. Determine the version of Edge you want to upgrade to.
 - We recommend using the latest version of Edge.
 - You can find this in Edge by following the steps below:
 - i. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - ii. Click **Edge**.
 - » The Edge sites overview appears.
 - iii. Above the table, to the right, click **Create Edge site**.
 - » The **Create Edge site** wizard starts.
 - iv. Select **Manual** under the **Upgrade Mode**
 - » The latest version is selected by default.
 - You will need the version, for example 2023.11 or 2023.01, to obtain the images list for the next step.
3. Obtain the images list of the Edge version you want to mirror in your private docker registry.

Note You must install the `jq` command.

- Use CURL to get an image:

```
curl -u <dic_user> -p <dgc_pass> <dic_url>/edge/rest/v2/releaseinfo/<edge_version> | \
jq '.images[].image' -r
```

- `dic_user`: The username you use to log into Collibra Data Intelligence Cloud.
- `dic_pass`: The password you use to log into Collibra.
- `dic_url`: The Collibra URL.
- `edge_version`: The version of Edge you want to upgrade to found in step 3. For example, `<2023.11>`.

a. Pull and push each image to your private docker registry.

i. Pull the image from Collibra's registry:

```
docker pull edge-docker-
delivery.repository.collibra.io/capabilities/edgeharves
ter:1.5.0
```

ii. Tag the image to for your private docker registry.

```
docker tag edge-docker-
delivery.repository.collibra.io/capabilities/edgeharves
ter:1.5.0 my-
registry.com/capabilities/edgeharvester:1.5.0
```

iii. Push the image to your private docker registry.

```
docker push my-
registry.com/capabilities/edgeharvester:1.5.0
```

Warning You can not change the path to the image. `edge-docker-delivery.repository.collibra.io/capabilities/edgeharvester:1.5.0` must be saved to `my-registry.com/capabilities/edgeharvester:1.5.0`

b. Install or reinstall the Edge site configured to use your private docker registry.

Note The Registry user should have read permissions for the docker registry.

a. For new Edge sites:

- i. Create a new Edge site with manual upgrade selected as the **Upgrade method**.
- ii. Download the installer.

b. For existing Edge sites:

- i. Set the **Upgrade method** to manual on the site details page.
- ii. Upgrade to the Edge version you want to use.
- iii. Download a new installer.
- iv. Create a [backup and reinstall](#) the Edge site.

c. The installer supports the following installation parameters:

- `--registry-url`, **example**: `https://edge-docker-delivery.repository.collibra.io`

- `--registry-host`, this is optional and is automatically derived from `--registry-url`, i.e., `edge-docker-delivery.my-registry.docker.io`
 - `--registry-pass`, not required if registry is public.
 - `--registry-user`, not required if registry is public.
- d. In order to install Edge with support for a private docker registry, the following set of additional parameters is required:
- **For Bare metal installations:** `./install-master.sh (...usual parameters...) --registry-url https://private-docker.registry.com --registry-user user --registry-pass pass`
 - **For EKS installations:** `./run-installer-job.sh (...usual parameters...) --registry-url https://private-docker.registry.com --registry-user user --registry-pass pass`

Note Before upgrading your Edge site, you will need to repeat the steps above using the new version of Edge.

Configure private docker registry

You must extract the private repository credentials to the current Edge docker repository before you can begin configuring your private docker registry.

To do that, open the **registries.yaml** file. The content in the file should be similar to the following:

```
version: 1
configs:
- url: "edge-docker-delivery.repository.collibra.io"
  username: "user"
  password: "pass"
```

Currently, this file contains a single registry, (`<edge-docker-delivery.repository.collibra.io>`), though in the future, it may contain more. It is important that any additional future registries are mirrored in the private docker registry as well.

Note If the docker registry does not have a user name/password authentication, then simply omit the docker username and password from registries.yaml or the Edge installer CLI arguments.

Edge connections

Edge connections define how an [Edge capability](#) accesses a data source.

Edit a connection

You can update the details of a data source by editing the connection. This topic will discuss how you can generally edit a connection. For more specific information, review the requirements for your data source, such as Technical lineage and [Sample data](#).



Note Refer to the [JDBC connections](#) documentation for how to edit JDBC connections.

Prerequisites

- You have a global role that has the System administration global permission.
- You have a global role that has the Manage Edge sites global permission.
- You have [created](#) and [installed](#) an Edge site.

Note It is possible there are extra requirements for your specific data source. Review the requirements and permissions of your data source before making any changes.

Steps

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. Locate and click the connection you want to edit.
3. At the bottom of the page, click **Edit**.
4. Edit the connection information.
5. Click **Save**.

Delete a connection



You can delete a connection from an [Edge site](#) to a data source if you no longer need it. This topic will discuss how you can generally delete a connection. For more specific information, review the requirements for your data source, such as [Technical lineage](#) and [Sample data](#).

Note Refer to the [JDBC connections](#) documentation for how to edit JDBC connections.

Prerequisites

- You have a global role that has the System administration global permission.
- You have a global role that has the Manage Edge sites global permission.
- You have [created](#) and [installed](#) an Edge site.

Steps

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. Locate and select the connection you want to delete.
3. At the bottom of the page, click **Delete**.
 - » The **Delete confirmation** dialog box appears.
4. Click **Delete Connection**.

Create a Shared storage connection

A Shared storage connection allows you to grant your capabilities read-only access to files from a shared folder on any cloud storage provider. This connection is especially useful for capabilities with large files, such as Technical lineage, as you do not need to manually upload these files directly to Edge, but rather define the file path when creating the new connection.

Note The Shared storage connection is currently only available for the Technical lineage capability.

Before you begin



- You have [created](#) an Edge site on K3S.

Note This connection type is not available for Edge sites created on EKS.

Prerequisites

- You have a global role that has the System administration global permission.
- You have a global role that has the Manage Edge sites global permission.

Steps

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. Click **Create Connection**.
 - » The Connection settings page appears.

3. Enter the connection information.

Field	Description	Required
Name	The name of the connection. Edge	✓ Yes
Description	The description of the connection.	✗ No
Connection provider	Select Shared storage connection <div style="border: 1px solid #ccc; background-color: #f9f9f9; padding: 5px; margin-top: 5px;"> <p>Note This folder path must be relative to <code>/var/lib/edge/storage/</code>.</p> </div>	✓ Yes

4. Click **Create**.

What's next?

You can now [create an Edge capability](#) using your Shared storage connection.

JDBC connections

JDBC connections define how an [Edge capability](#) accesses a data source.

To [create a connection to your data source](#), you need to select a connection provider, which determines the available properties of the connection, such as the authentication method and connection string and driver.

Example If you want to ingest data from an Amazon Redshift data source, you need a specific JDBC driver for Amazon Redshift. You use that driver to create a connection between your Edge site and your Amazon Redshift data source.

Tip Collibra provides a selection of certified JDBC drivers on [Collibra Marketplace](#). We highly recommend to only use [JDBC drivers that are certified for Edge](#).



Copy the URL of this page.

ources supported by Edge

You can [register](#), [profile](#) and [classify](#) several data sources via Edge. Depending on your data source, you can use a Collibra-provided Catalog connector, or your own JDBC driver when you create a [JDBC connection](#).

The following data sources have been tested for registering, profiling and classifying via Edge.

Create a JDBC connection

You can create a [JDBC connection](#) from an [Edge site](#) to a data source. You can then [register the data source via Edge](#).

Available Catalog connectors

Edit a JDBC connection

You can edit a [JDBC connection](#), for example if you want to change one of its connection properties. You can then [register the data source via Edge](#).

Available Catalog connectors



Delete a JDBC connection

You can delete a [JDBC connection](#) from an [Edge site](#) to a data source if you no longer need it.

Prerequisites

- You have a global role that has the System administration global permission.
- You have a [global role](#) that has the Manage connections and capabilities [global permission](#).
- You have [created](#) and [installed](#) an Edge site.
- You have created a [JDBC connection](#).

Steps

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. In the **JDBC Connections** section, click the name of a JDBC connection.
 - » The **Connection settings** page appears.
3. At the bottom of the page, click **Delete**.
 - » The **Delete confirmation** dialog box appears.
4. Click **Delete Connection**.

Use keys to access a database

It is possible that, to access a database, the JDBC driver requires a private key. In this case, you have to manually add extra connection properties when you [create a JDBC connection](#).

For example, the Snowflake driver exposes **private_key_file** and **private_key_file-pwd** properties. You can use these connection properties for the connection with Snowflake as shown in the following image.

Create connection

Connection settings

Name *

Description

Connection provider *
 Generic connection provider, established JDBC connection with the provided JDBC driver. Uses any authentication scheme supported by the driver

Connection parameters

Driver class name *
 The fully qualified name of the jdbc driver

Driver jar *
 The jar file containing the jdbc driver class

Connection string *
 The jdbc connection string

Connection properties

Name *	Type *	Encryption *	File upload *
<input type="text" value="private_key_file"/>	<input type="text" value="File"/>	<input type="text" value="To be encrypted by Edge management server"/>	<input type="text" value="snowflake.pk"/> <input type="button" value="Upload"/>
Name *	Type *	Encryption *	Value *
<input type="text" value="private_key_file_pwd"/>	<input type="text" value="Text"/>	<input type="text" value="To be encrypted by Edge management server"/>	<input type="text" value="....."/>

Edge capabilities

An Edge capability is an application that runs on an [Edge site](#) to extract and process data. It delivers the results to Collibra Data Intelligence Cloud.



About Edge capabilities

An Edge capability, like Sampling or S3 synchronization, is an application that can run on an Edge site. It can access a data source to extract and process data as needed. This data can be stored in an encrypted cache to improve the security of your data and platform. An Edge capability for a specific data source runs as a job and delivers the output to Collibra Data Intelligence Cloud in a secure and reliable way.

An Edge capability has a capability template that defines a specific use case, for example data source ingestion.

Capability templates

A capability template is developed for a specific task on a specific data source type. The capability template also determines which properties are available to configure the Edge capability.

Currently, the following capability templates are available:

- [Catalog JDBC ingestion](#): A capability template you use to [register a data source](#) and [synchronize schemas](#) from a data source via a JDBC connection.
- [JDBC Profiling](#): A capability template you use to [profile and classify](#) data from a registered data source.
- [Catalog JDBC Sampling](#): A capability template you use to [collect and cache sample data](#) from a data source in the Edge site via a JDBC connection.
- [S3 synchronization](#): A capability template you use to [connect to Amazon S3](#).
- [GCS synchronization](#): A capability template you use to [connect to Google Cloud Storage](#).
- [DQ Connector](#): A capability template you use to ingest Collibra Data Quality & Observability user-defined rules, metrics, and dimensions into Collibra Data Catalog.
- [Technical lineage capabilities \(beta\)](#): Capability templates you use to create technical lineage for different data sources. For details, go to:
 - [Add a technical lineage capability to an Edge site for JDBC data sources and ETL tools](#).

- Add the Technical Lineage for Power BI capability to the Edge site.
- Add the Technical Lineage for Tableau capability to the Edge site.

Important While these capability templates are available for all customers, the features that you use them for might still be in beta.

Capability template structure

Each Edge capability template contains the following:

File	Description
A manifest file (YAML)	This file contains the capability metadata and input parameter requirements.
A workflow file (YAML)	This file defines the workflow and binds the parameters to capability containers.
Docker images	One or more Docker images that implement the business logic.

Page layout

The following image shows the page for adding an edge capability.

Edge sites ▶ site3

Add capability

Capability

Name *

Please provide a meaningful name.

Description

Please provide a meaningful description.

Capability template *

Select ▼

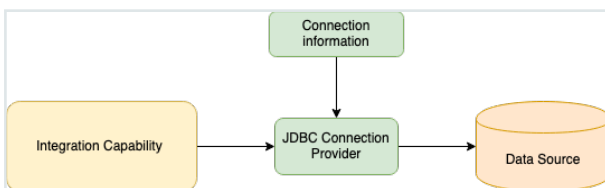
Cancel

Create

About Edge capabilities connecting to data sources

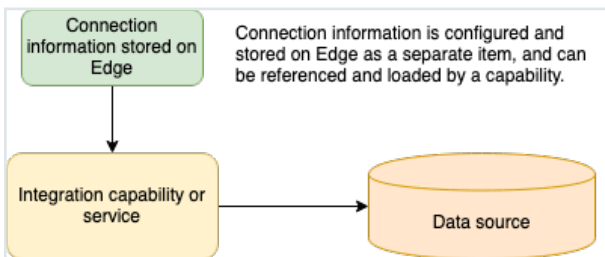
A connection on an Edge site identifies a unique system, whether it's a database, a file share or a REST service.

For JDBC (Java Database Connectivity), you can connect directly from the Edge user interface. When you create a JDBC connection, you will enter your login credentials, which will then be stored for authentication. This means that you will not need to enter these credentials again for any capability that uses this JDBC connection.



If an integration capability does not connect to a JDBC data source, it has to connect on its own by using the information provided by Edge. The connection information is defined and stored as a Connection instance. The connection properties are shown on the Connections configuration page within Edge user interface.

Below is an example of a capability that does not use JDBC to connect:



Connection types

All supported connection types are bundled in Edge. You cannot add new connection types, for example Tableau or S3.

Add an Edge capability to an Edge site

After you have created and installed an [Edge site](#), you can add an [Edge capability](#) to perform specific tasks on a data source. For example, you can [register a data source](#) using a [JDBC connection](#) that belongs to an Edge capability.

Prerequisites

- You have a global role that has the System administration global permission.
- You have a global role with the Manage connections and capabilities global permission, for example Edge integration engineer.
- You have a global role with the Register profiling information global permission. (optional)
- You have [created](#) and [installed](#) an Edge site.
- You have created a [JDBC connection](#).

Steps

Tip For more information about all fields in the capability, go to the [online version of the documentation](#).

More information

Go to the [Catalog JDBC ingestion](#) documentation for more information.

Go to the [JDBC Profiling](#) documentation for more information.

Go to the [Catalog JDBC Sampling](#) documentation for more information.

Go to the [S3 synchronization](#) documentation for more information.

Go to the [GCS synchronization](#) documentation for more information.

Go to the [Technical lineage for JDBC data sources and ETL tools \(public beta\)](#) documentation for more information.

Go to the [Technical lineage for Tableau \(public beta\)](#) documentation for more information.

Go to the [Technical lineage for Power BI \(public beta\)](#) documentation for more information.

Edit an Edge capability of an Edge site

You can edit an [Edge capability](#) of an [Edge site](#), for example to change the custom properties.

Prerequisites

- You have a global role that has the System administration global permission.
- You have a global role with the Manage connections and capabilities global permission, for example Edge integration engineer.
- You have a global role with the Register profiling information global permission. (optional)
- You have [created](#) and [installed](#) an Edge site.
- You have added an [Edge capability](#) to the Edge site.

Steps

Tip For information about the various capabilities, go to the [online version of the documentation](#).

Delete an Edge capability from an Edge site



You can remove an [Edge capability](#) from an [Edge site](#) if you no longer need it.

Warning If you delete a JDBC Profiling capability and synchronize previously profiled and classified schemas again, the profiling and classification results are removed.

Prerequisites

- You have a global role that has the System administration global permission.
- You have a global role with the Manage connections and capabilities global permission, for example Edge integration engineer.
- You have [created](#) and [installed](#) an Edge site.
- You have added an [Edge capability](#) to the Edge site.

Steps

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. In the **Capabilities** section, click the name of a capability.
 - » The **Capability** page appears and shows a read-only overview of the capability.
3. Click **Delete**.
4. Click **Delete Capability**.
 - » The capability is deleted from the Edge site.

Edge Jobs dashboard

The Edge Jobs dashboard gives you an overview of all jobs that are executed by an Edge site.

When you [enable](#) the Edge Jobs feature (beta) in Collibra Console, the Edge Jobs dashboard becomes available in the Collibra Data Intelligence Cloud settings.


Note Only users with the Admin role can enable this feature.

Important This is a [beta feature](#).

 **Edge**
Connect to local data sources to extract metadata (structure, profiling stats, quality, classes, lineage...) and show it in Collibra.

Sites Jobs (beta)

On the Edge Jobs dashboard, you find an overview of all jobs that have either been scheduled or completed in your Edge sites. Each job is a row in the table and contains basic information such as start and completion date, status, Edge site, capability and so on. You can also open the [log files](#) of a job and [cancel a scheduled job](#) from this dashboard.

 Sites

Jobs beta

0/528 [View Output Files](#) [Cancel](#)

ⓘ This feature is being actively developed and may not meet all necessary criteria for enterprise-grade functionality. In the future, the beta label will be removed and this feature will be considered Generally Available.

Job ID	Capability	Started ↓	Completed	Duration	Status	Site	Started by
0e5f445b-2bec-4ea0-a...	DEMO SqlDir cap	14/09/2022, 15:03:54	14/09/2022, 15:04:23	0m 29s	SUCCEEDED	techlin	Admin Istrator
1aec633a-8694-45db-a...	Tableau generic c	14/09/2022, 14:46:52	14/09/2022, 14:48:24	1m 31s	FAILED	techlin	Admin Istrator
dbca10f3-2916-4e49-b...	Tableau generic c	14/09/2022, 14:19:49	14/09/2022, 14:22:14	2m 24s	FAILED	techlin	Admin Istrator
ad9fb8f9-7486-4265-9...	Tableau generic c	14/09/2022, 14:14:48	14/09/2022, 14:14:52	0m 3s	FAILED	techlin	Admin Istrator
33e1d78e-93c2-4bb2-...	Tableau generic c	14/09/2022, 13:54:46	14/09/2022, 13:55:18	0m 31s	FAILED	techlin	Admin Istrator
641cb411-f942-4aba-b...	Tableau generic c	14/09/2022, 12:39:38	14/09/2022, 12:39:49	0m 10s	FAILED	techlin	Admin Istrator
d694eb31-36d6-4087-...	Tableau generic c	14/09/2022, 12:29:37	14/09/2022, 12:29:59	0m 21s	FAILED	techlin	Admin Istrator
cab0326-117c-4a85-a...	Tableau Legacy	14/09/2022, 11:55:33	14/09/2022, 11:56:37	1m 3s	FAILED	techlin	Admin Istrator
53d90615-9796-4282-...	DEMO SqlDir cap	14/09/2022, 10:59:27	14/09/2022, 10:59:23	-1m -5s	FAILED	techlin	Admin Istrator
3aa95aaf-8baf-4baf-89...	DEMO SqlDir cap		14/09/2022, 10:35:01		FAILED	techlin	Admin Istrator
c2043a51-40c2-4d61-a...	DEMO SqlDir cap	14/09/2022, 10:25:24	14/09/2022, 10:25:53	0m 29s	FAILED	techlin	Admin Istrator
74706644-70f8-4a42-8...	DEMO SqlDir cap	14/09/2022, 10:16:10	14/09/2022, 10:16:26	0m 16s	SUCCEEDED	techlin	Admin Istrator
6176b3c3-b791-4b75-...	DEMO SqlDir cap	14/09/2022, 10:01:16	14/09/2022, 10:01:33	0m 16s	SUCCEEDED	techlin	Admin Istrator
d0db09fb-c140-4d7b-b...	DEMO SqlDir cap	14/09/2022, 09:54:07	14/09/2022, 09:54:43	0m 35s	SUCCEEDED	techlin	Admin Istrator
e8df0139-8486-4234-8...	jdbc-sampler	13/09/2022, 14:29:31	13/09/2022, 14:29:52	0m 21s	FAILED	5fc572-4322-4eac-ae9f-fAdmin Istrator	
a7492a50-4514-41b0-a...	jdbc-sampler	13/09/2022, 14:26:30	13/09/2022, 14:28:30	2m 0s	SUCCEEDED	5fc572-4322-4eac-ae9f-fAdmin Istrator	
cd415543-a5b8-4a63-9...	PG Cat lng	12/09/2022, 15:22:51	12/09/2022, 15:25:12	2m 20s	SUCCEEDED	5fc572-4322-4eac-ae9f-fSystem User	

◀ 1-20 21-40 41-60 61-80 ... 521-528 ▶
Updated 20/09/2022 17:03:29

You can also view the jobs associated to a specific Edge site by going to the **Jobs** tab of that site.

1. Click **Sites**.
2. Select your site from the list.
3. Click **Jobs** in the tab menu.

The screenshot displays the 'Jobs' tab for a site named 'techlin'. The interface includes a navigation sidebar with 'Sites' and 'Jobs' (marked as beta). The main content area shows the site's health as 'Healthy' and provides details for the 'Techlin team site'. Below this, there are tabs for 'Connections', 'Capabilities', 'Vaults', and 'Jobs'. A warning message indicates that the feature is in beta. A table lists 10 jobs with their respective IDs, capabilities, start and completion times, durations, and statuses. The jobs are performed by 'Admin Istrator'.

Job ID	Capability	Started	Completed	Duration	Status	Started by
fab0e821-a196-4be3-b...	DEMO SqlDir cap	15/09/2022, 12:51:28	15/09/2022, 12:51:45	0m 17s	SUCCEEDED	Admin Istrator
68ffaddb-2907-4182-b...	DEMO SqlDir cap	15/09/2022, 11:45:13	15/09/2022, 11:45:29	0m 16s	SUCCEEDED	Admin Istrator
5bb32bab-5dcd-4c05-...	Tableau generic c	15/09/2022, 08:41:13	15/09/2022, 08:42:29	1m 16s	FAILED	Admin Istrator
073cfbb4-e2bf-4f64-9b...	DEMO SqlDir cap	15/09/2022, 08:41:13	15/09/2022, 08:41:35	0m 21s	SUCCEEDED	Admin Istrator
9d8c1206-efe2-45a9-b...	DEMO SqlDir cap	14/09/2022, 15:23:54	14/09/2022, 15:24:14	0m 20s	SUCCEEDED	Admin Istrator
0e5f445b-2bec-4ea0-a...	DEMO SqlDir cap	14/09/2022, 15:03:54	14/09/2022, 15:04:23	0m 29s	SUCCEEDED	Admin Istrator
1aec633a-8694-45db-a...	Tableau generic c	14/09/2022, 14:46:52	14/09/2022, 14:48:24	1m 31s	FAILED	Admin Istrator
dbc10f3-2916-4e49-b...	Tableau generic c	14/09/2022, 14:19:49	14/09/2022, 14:22:14	2m 24s	FAILED	Admin Istrator
ad9fb8f9-7486-4265-9...	Tableau generic c	14/09/2022, 14:14:48	14/09/2022, 14:14:52	0m 3s	FAILED	Admin Istrator
33e1d78e-93c2-4bb2-...	Tableau generic c	14/09/2022, 13:54:46	14/09/2022, 13:55:18	0m 31s	FAILED	Admin Istrator
641cb411-f942-4aba-b...	Tableau generic c	14/09/2022, 12:39:38	14/09/2022, 12:39:49	0m 10s	FAILED	Admin Istrator

Navigation: 1-20 21-40 41-60 61-80 81-82 Updated 20/09/2022 17:02:48



Cancel jobs

You can cancel an Edge site job which is either running or queued to run.

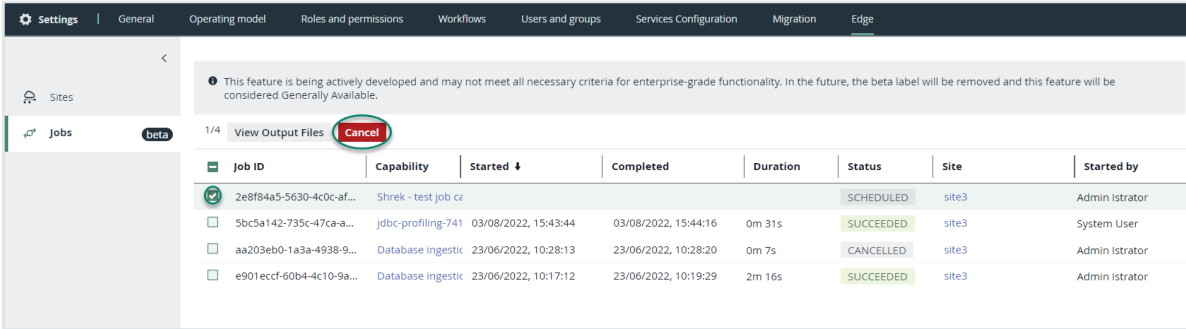
Prerequisites

- You have a [global role](#) that has the Manage connections and capabilities [global permission](#).
- You have jobs currently running or queued .

Steps

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. In the tab pane, click **Edge**.
 - » The Edge sites overview appears.
 - c. In the Edge site overview, click the name of an Edge site with the status **Healthy**.
 - » The Edge site page appears.
2. Click **Jobs**.
3. Select the checkbox next to the job you would like to cancel.

Tip You can select more than one job at a time.



Job ID	Capability	Started ↓	Completed	Duration	Status	Site	Started by
<input checked="" type="checkbox"/> 2e8f84a5-5630-4c0c-af...	Shrek - test job ce	03/08/2022, 15:43:44	03/08/2022, 15:44:16	0m 31s	SCHEDULED	site3	Admin Istrator
<input type="checkbox"/> 5bc5a142-735c-47ca-a...	jdbc-profiling-741	23/06/2022, 10:28:13	23/06/2022, 10:28:20	0m 7s	SUCCEEDED	site3	System User
<input type="checkbox"/> aa203eb0-1a3a-4938-9...	Database ingestio	23/06/2022, 10:17:12	23/06/2022, 10:19:29	2m 16s	CANCELLED	site3	Admin Istrator
<input type="checkbox"/> e901ecf-60b4-4c10-9a...	Database ingestio				SUCCEEDED	site3	Admin Istrator

4. In the action toolbar, click **Cancel**.
 - » The job is canceled, and the status of this job is CANCELED.

Maintaining Edge sites

In this section, you will learn how you can maintain your Edge site installations, such as performing backups or updating credentials.



Running Edge tools

This section contains an overview on how to use the Edge tools, for example to create a backup of your Edge site.

Prepare the Edge tools on K3S

On K3S, the Edge tool is downloaded at the end of a successful installation.

Alternatively, you can download it from the cluster:

```
TOOLS_POD=$(sudo /usr/local/bin/kubectl -n collibra-edge get
pod -l edge.collibra.com/contains=tools -o jsonpath='{.items
[0].metadata.name}')

sudo /usr/local/bin/kubectl cp collibra-edge/$TOOLS_POD:edge
/usr/local/bin/edge

sudo chmod +x /usr/local/bin/edge
```

The Edge command is in `/usr/local/bin` on the host. This is your first worker node, so you run the Edge command on the actual host where K3S runs.

Overview Edge commands on K3S

Edge tool	Command for K3S
Uninstall Edge	<code>/usr/local/bin/uninstall-edge.sh</code>
Create Edge diagnostics file	<ul style="list-style-type: none"> Edge site is not yet installed: <code><extracted installer directory>/resources/tools/edge-diagnostics.sh -d <file name>.tgz</code> Edge site is up and running: <code>edge diagnostics -d <file name>.tgz</code>

Edge tool	Command for K3S
Create an Edge site backup	<code>edge backup -o /<path to folder>/<backup-name>.yaml</code>
Set Edge storage cache ttl	<code>edge cachettl --ttl <value in days></code>
Retrieve logs from a catalog connector	<code>edge catalog-connector --jobid <Edge job ID> \ --dst <path to destination></code>
Update Collibra credentials	<ul style="list-style-type: none"> • Interactive way: <code>edge update-dgc-creds -i</code> • Explicit update: <code>edge update-dgc-creds <username> <password> <url collibra environment></code>
Update forward proxy settings	<code>edge update-outbound-proxy --update-outbound-proxy /path/to/proxy.properties</code>
Get help to set up no_proxy configuration	<ul style="list-style-type: none"> • Edge site is not yet installed: <code><extracted installer directory>/resources/tools/edge-get-noproxy.sh k3s</code> • Edge site is up and running: <code>edge get-noproxy k3s</code>

Prepare Edge tools on EKS

Edge is installed from a Linux machine that has access to the actual K8S cluster.

There is no automatic download of the Edge tool after installation because we don't want to enforce it in some location. Therefore, you have to download the Edge tool to your Linux machine, for example in your current folder:

```

TOOLS_POD=$(kubectl -n collibra-edge get pod -l
edge.collibra.com/contains=tools -o jsonpath='{.items
[0].metadata.name}')

kubectl cp collibra-edge/$TOOLS_POD:edge edge

chmod +x edge

```

You can now run Edge commands from your current folder.

Note As you are not on the worker node itself, you cannot collect worker node diagnostics. If you need these diagnostics, [create a support ticket](#).

Overview Edge commands on EKS

Edge tool	Command for EKS
Uninstall Edge	<code><extracted installer directory>/resources/tools/installer-job/uninstall-edge-on-managed-k8s.sh</code>
Create Edge diagnostics file	<ul style="list-style-type: none"> Edge site is not yet installed: <code><extracted installer directory>/resources/tools/edge-diagnostics.sh -d <file name>.tgz</code> Edge site is up and running: <code>edge diagnostics -d <file name>.tgz</code>
Create an Edge site backup	<code>edge backup -o /<path to folder>/<backup-name>.yaml</code>
Set Edge storage cache ttl	<code>edge cachettl <value in seconds></code>

Edge tool	Command for EKS
Retrieve logs from a catalog connector	<pre>edge catalog-connector --jobid <Edge job ID> \ --dst <path to destination>/<file name>.txt</pre>
Update Collibra credentials	<ul style="list-style-type: none"> • Interactive way: <pre>edge update-dgc-creds -i</pre> • Explicit update: <pre>edge update-dgc-creds <username> <password> <url collibra environment></pre>
Update forward proxy settings	<pre>edge update-outbound-proxy --update-outbound-proxy /path/to/proxy.properties</pre>
Get help to set up no_proxy configuration	<ul style="list-style-type: none"> • Edge site is not yet installed: <pre><extracted installer directory>/resources/tools/edge-get-noproxy.sh eks <clustername></pre> • Edge site is up and running: <pre>edge get-noproxy eks <clustername></pre>



Edit an Edge site

You can edit a [Edge site](#) to give it another name or description.

Prerequisites

- You have [created](#) an Edge site.
- You have a global role that has the System administration global permission.
- You have a global role that has the Manage Edge sites global permission.

Steps

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. In the top right corner, click **Actions** → **Edit**.
 - » The Edit Edge site wizard starts.
3. Enter the required information.

Field	Description
Name	<p>The name of the Edge site. Use a meaningful name, for example NetherlandsDataCentre1. Do not use spaces or special characters.</p> <p>This field is mandatory and the name must be globally unique.</p>
Description	<p>The description of the Edge site. We recommend to put at least basic location information of the Edge site.</p> <p>This field is mandatory.</p>

4. Click **Save**.
 - » The Edge sites overview appears with the new name and description.

Update Edge user password

When you [download the Edge site installer](#), a dedicated user account is created in Collibra Data Intelligence Cloud. This user always has "Edge" as first name and the "Edge's site name" as the last name.

A user will be created for each Edge site. This user is deleted when you delete the Edge site.

Note The Edge user account must have the Connect Edge to Collibra global permission.

Steps

1. Reset the password of the Edge user in Collibra by following the steps in our [Set or reset a user password](#) article.

Note Review the default password requirements [here](#).

2. Connect to the Edge master node via SSH.
3. Run the following script: `/usr/local/bin/edge update-dgc-creds -i`
4. Enter the username and new password of the Edge user.

Update the outbound proxy configuration

If you have to change the outbound proxy configuration of a running Edge site, you can use Collibra's outbound proxy update script.

Steps

1. Find the **proxy.properties** file on the server that you used during the [configuration of the outbound proxy](#).
2. Update the file with the new [property](#) values and save the file.
3. Go to **/usr/local/bin** and run the following command:

```
./edge update-outbound-proxy -u /path/to/proxy.properties
```

Help file of the script

```
$ /usr/local/bin/edge update-outbound-proxy --help
Collibra Edge Utility for updating Outbound Proxy settings.
Usage:
    edge update-outbound-proxy.sh -h|--help
    edge update-outbound-proxy.sh -g|--generate-template
<filename>
    edge update-outbound-proxy.sh -u|--update-outbound-
proxy <filename>

    -h|--help                - Show help
    -g|--generate-template   - generate template file for
proxy properties in <filename>
    -u|--update-outbound-proxy - update outbound-proxy secret
based on proxy properties <filename>
```

Back up an Edge site

To avoid losing your Edge site configurations, you can [back up](#) an [Edge site](#). You can use this backup to [restore](#) it later, for example when you want to reinstall an Edge site with a new installer.

The following will be included in the backup:

- The public/private key of the site used for sending and encrypting secrets.
- The [secrets](#) used in connections, capabilities and vaults.

Note For privacy reasons, Edge site backups remain in your personal environment and are not sent to the cloud.

On the server that runs your Edge site, execute the following command:

```
~$ edge backup -o /<path to folder where you want to save the backup file>/<backup-name>.yaml
```

» Edge creates a backup of your Edge site in the selected folder of the command.

Tip If the Edge command is not available, you will need to [download](#) the Edge tool.

On the server from which you manage your EKS cluster, execute the following command:

```
~$ edge backup -o /<path to folder where you want to save the backup file>/<backup-name>.yaml
```



» Edge creates a backup of your Edge site in the defined folder of the last command.

Tip If the Edge command is not available, you will need to [download](#) the Edge tool.



Restore an Edge site

This article walks through how to restore your Edge site. You may want to restore it if you've previously created a [backup](#) or want to reinstall an Edge site with a new installer.

Restoring an Edge site is the same command as installing an Edge site but with an extra argument to use a backup.

1. Optionally, [download a new Edge installer](#).
1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. In the top right corner, click **Actions** → **Redownload installer**.
 - » A new Edge installer is downloaded.
2. Run the Edge installer and add the backup file as a parameter:

```
install-master.sh properties.yaml --storage-path
/var/edge/storage -r registries.yaml -b /<path to backup
file>/edge-backup.yaml
```

1. Optionally, [download a new Edge installer](#).
1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. In the top right corner, click **Actions** → **Redownload installer**.
 - » A new Edge installer is downloaded.
2. Run the Edge installer and add the backup file as a parameter:

```
./run-installer-job.sh properties.yaml --set collibra_
edge.collibra.ozone.datanode.pvcSize=100Gi --set collibra_
edge.collibra.ozone.om.pvcSize=100Gi
```



Delete an Edge site

You can delete an [Edge site](#) if you no longer need it.

Prerequisites

- You have [created](#) an Edge site.
- You have a global role that has the System administration global permission.
- You have a global role that has the Manage Edge sites global permission.



Steps

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. In the top right corner, click **Actions** → **Delete Edge site**.
 - » The Delete Edge site wizard starts.
3. Click **Delete Edge site**.
 - » The Edge sites overview appears, without the deleted Edge site.
4. On the server that hosts the Edge site, go to `/usr/local/bin` where you can find the uninstall script `uninstall-edge.sh`, then run one of the following commands:

Command	
Delete Edge site but keep its data. The data consists of drivers, required files for capabilities, and data that was saved by Edge capabilities	<pre>/usr/local/bin/uninstall-edge.sh</pre>

	Command
Delete Edge site and its data.	<pre>/usr/local/bin/uninstall-edge.sh --remove-local-data</pre>
Delete Edge site without confirmation request, for example if you want to delete the site via a script. You can use this in combination with removing the site data.	<pre>/usr/local/bin/uninstall-edge.sh --remove-local-data --force</pre>

Warning When you delete an Edge site, the Elastic Block Store (EBS) volumes containing the data are also removed. If you like to keep your data, first back up these EBS volumes.

1. Open an Edge site.
 - a. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
 - b. Click **Edge**.
 - » The Edge sites overview appears.
 - c. Click the name of an Edge site in the Edge site overview.
 - » The Edge site page appears.
2. In the top right corner, click **Actions** → **Delete Edge site**.
 - » The Delete Edge site wizard starts.
3. Click **Delete Edge site**.
 - » The Edge sites overview appears, without the deleted Edge site.
4. On the server from which you manage your EKS cluster, run this command:

```
<extracted installer>/resources/tools/installer-job/uninstall-edge-on-managed-k8s.sh
```

Troubleshooting Edge

In this section, you find some articles that help you to troubleshoot Edge issues.

General troubleshooting Edge

The following table shows how to solve issues you may encounter while working with Edge. Select the tab of your installation type, K3S or EKS.

Issue	Proposed solution
<p>You get the following error message:</p> <pre>Out of disk space</pre>	<p>You have to restart the Kubernetes pod in Edge.</p> <p>Run the following command:</p> <pre>sudo /usr/local/bin/kubectl delete pod <pod_name> -- namespace <pod_namespace></pre> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>Tip For more information about Pods and namespaces, see the Kubernetes documentation.</p> </div>
<p>You get one of the following error message:</p> <ul style="list-style-type: none"> • Cannot allocate memory • Error syncing pod 	<p>You have to restart Edge.</p> <p>Do the following:</p> <ol style="list-style-type: none"> 1. Run the following commands to remove all workflows: <pre>sudo /usr/local/bin/kubectl delete --all workflows -- namespace=collibra- capabilities</pre> <pre>sudo /usr/local/bin/kubectl delete --all workflows -- namespace=collibra-fast</pre> 2. Run the following command to reboot Edge: <pre>sudo reboot</pre>

Issue	Proposed solution
<p>You get the following error message:</p> <pre>Out of disk space</pre>	<p>You have to restart the Kubernetes pod in Edge.</p> <p>Run the following command:</p> <pre>kubect1 delete pod <pod_name> -- namespace <pod_namespace></pre> <p>Tip For more information about Pods and namespaces, see the Kubernetes documentation.</p>
<p>You get one of the following error message:</p> <ul style="list-style-type: none"> • Cannot allocate memory • Error syncing pod 	<p>You have to restart Edge.</p> <p>Do the following:</p> <ol style="list-style-type: none"> 1. Run the following commands to remove all workflows: <pre>kubect1 delete --all workflows --namespace=collibra- capabilities</pre> <pre>kubect1 delete --all workflows --namespace=collibra-fast</pre> 2. Run the following command to reboot Edge: <pre>sudo reboot</pre>

Use an explicit `resolv.conf` file for Edge

Important This is only applicable for K3S installations.

The default resolver configuration file `/etc/resolv.conf` is in most cases picked-up by K3S and used successfully, but on Google Cloud Platform, where the default nameserver is `169.254.169.254`, K3S generates another file with a nameserver pointing to `8.8.8.8`.

Your firewall or network configuration may be filtering connections to `8.8.8.8`, in which case the resolver file has to be explicitly configured with a reachable nameserver. If the default file `/etc/resolv.conf` is explicitly configured even on GCP nodes having nameserver `169.254.169.254`, then K3S will successfully use it.

You can also explicitly indicate in `install-master.sh` to use `/etc/resolv.conf` by adding the argument `--resolv-conf </path/to/resolv.conf>`.

Example

```
sudo sh install-master.sh --storage-path /var/edge/storage
properties.yaml -r registries.yaml --resolv.conf
/etc/resolv.conf
```

Edge logging

When you encounter an issue in Edge, you can use diagnostic and log files which provide data about the issue. If you want to report a problem to Collibra support, you can include these files in the support ticket. As a result, Collibra support will be able to determine what went wrong and find a solution to your issue.

You can create a diagnostics ZIP file with logs and information about the server or EKS environment on which you installed the Edge site. Edge also generates two types of log files that are not included in the diagnostics file:

- [Edge infrastructure log files](#), which are sent to Collibra Data Intelligence Cloud immediately upon creation.
- [Metadata connector log files](#), which can only be stored locally.

Edge diagnostics file

The Edge diagnostics file is a ZIP file that is created by running the diagnostics script in the Edge site installer folder. The diagnostics script checks amongst others:

- Your operating system setup
- Your firewall settings
- Connectivity information
- Edge cluster logs.

You can send the diagnostics file to Collibra support when you have an issue with the [Edge site installation](#).

Edge infrastructure log files

Edge infrastructure logs contain Edge infrastructure information, for example Edge status updates and capability information. The logs can be used by Collibra Support to help solve



general Edge issues. The log files do not contain any database content or private information.

By default, the Edge infrastructure log files are always enabled on an information level. You can [enable debug level logging](#) per specific capability when you add or edit an Edge capability. As a result, Edge sends infrastructure logs with more information about that capability to Collibra Data Intelligence Cloud. Edge infrastructure log files can contain the following information:

- Job execution phases
- The Edge status
- Service updates
- System upgrades

These log files can only be accessed by Collibra Support.

Note By default, **Debug** logging for an Edge capability is set to `False`. We highly recommend only enabling the **Debug** logging for an Edge capability if an issue arises.

Metadata connector log files

Metadata connector log files contain the logs of the JDBC connections between the Edge capability and your data source. These log files can be used by Collibra Support to help solve issues with processing or accessing data. The log files may contain information about your data source.

For security reasons, these log files are not automatically sent to your Collibra Data Intelligence Cloud environment. You can, however, [create the log files](#), save and review them locally and then attach them to a Collibra support ticket.

Edge system monitoring

The system monitoring, executed via OpenTelemetry, sends the following information to your Collibra environment:

- CPU usage
- Memory usage
- Network statistics

Collibra Support can then analyze this information to troubleshoot potential anomalies. This data is only available to Collibra personnel.

Verbosity log levels

The verbosity log levels indicate how much information you want to see in the Catalog Connector log files. You can change the verbosity log levels in the Edge capability for which you want to create logs. The following verbosity log levels are available:

Verbosity log level	Description
No logging	The Catalog Connector logs are not created. This is the default.
Low	The Catalog Connector logs contain the following: <ul style="list-style-type: none">• All connection query logs• Any errors
Middle	The Catalog Connector logs include the Low logs and: <ul style="list-style-type: none">• All cache queries• Additional information about the request
High	The Catalog Connector logs include the Middle logs and: <ul style="list-style-type: none">• The body of the request• The response

Create an Edge diagnostics file

You can create an [Edge diagnostics file](#) to check issues with the [Edge site installation](#) in your environment.

Prerequisites

- You have [created](#) an Edge site.
- You have [downloaded](#) the Edge installer.

Steps

Edge site is not yet installed

You can run the diagnostics script without an Edge site installed to check if your system meets all requirements to install the Edge site.

1. Extract the Edge installer.

```
tar -xf <edge-site-id>-installer.tgz
```

2. On the command line, go to the folder with the extracted files.
3. In this folder, go to **resources/tools**.
4. Run the following command to create the diagnostics file:

```
edge-diagnostics.sh --diag-file <file name>.tgz
```

» A TGZ file with the given file name is created and contains all Edge diagnostics file.

Edge site is already installed

On the command line, run the following command to create the diagnostics file:

```
edge diagnostics --diag-file <file name>.tgz
```

» A TGZ file with the given file name is created and contains all Edge diagnostics file.

Tip If the Edge command is not available, you will need to [download](#) the Edge tool.

What's next?

You can send the diagnostics file to Collibra support to help you resolve your installation issues.

Create Metadata connector log files

If you have an issue with a JDBC connection, for example while registering a data source via Edge, you can create the [Metadata connector log files](#) and then save and review them locally. If you create a support ticket, attach the reviewed Metadata connector log files to your ticket so Collibra Support can help you with your issue.

Prerequisites

- You have a global role that has the System administration global permission.
- You have a global role with the Manage connections and capabilities global permission, for example Edge integration engineer.
- You have [created](#) and [installed](#) an Edge site.
- You have created a JDBC connection and an Edge capability.

Steps

1. Edit the Edge capability that contains the JDBC connection for which you want to create a log file.
 - a. Click the name of the Edge capability to open it.
 - b. Click **Edit**.
 - c. In the **General** section, click the **Log level** drop-down menu.
 - d. Select the log verbosity level.

Tip The level must be at least *low*.

- e. Click **Save**.
 - » The fields become read-only.
2. Click **Run** to rerun the Edge capability.
 3. Contact Collibra support to request the Edge job ID of the Edge capability.
 4. Run the following command:

```
./edge catalog-connector --jobid <Edge job ID> --dst <path to destination>
```

- » The log file is created and stored in the predefined destination.

Tip If the Edge command is not available, you will need to [download](#) the Edge tool.

Prerequisites

- You have a Linux host with kubectl access to your EKS installation.
- You have mc (minio client) installed in /usr/local/bin:

```
sudo curl -L "https://dl.min.io/client/mc/release/linux-amd64/mc" -o /usr/local/bin/mc
sudo chmod +x /usr/local/bin/mc
```

Steps

Execute the following commands:

```
kubectl -n collibra-edge port-forward service/minio 9000:9000 &

MC_ACCESSKEY="$(kubectl get secrets edge-minio -n collibra-edge
-o jsonpath="{.data.accesskey}" | base64 --decode) "

MC_SECRETKEY="$(kubectl get secrets edge-minio -n collibra-edge
-o jsonpath="{.data.secretkey}" | base64 --decode) "

export MC_HOST_edge="http://${MC_ACCESSKEY}:${MC_
SECRETKEY}@localhost:9000"

mc cp --quiet --recursive edge/cdata/<jobId> <destination_
directory>

pkill -f "port-forward"
```



Enable debug logging for Edge infrastructure logs

By default, the Edge infrastructure logs are always enabled on an information level. If you have an issue with [Edge](#) in general, you can enable Edge to create [Edge infrastructure debug log files](#) and send them to Collibra Data Intelligence Cloud. Collibra support uses these log files to solve Edge issues.

Prerequisites

- You have a global role that has the System administration global permission.
- You have a global role with the Manage connections and capabilities global permission, for example Edge integration engineer.
- You have [created](#) and [installed](#) an Edge site.
- You have created a JDBC connection and an Edge capability.

Steps

1. On the main menu, click , and then click  **Settings**.
 - » The [Collibra settings page](#) opens.
2. On the **Settings** page, click **Edge**.
 - » The Edge sites overview appears.
3. Click the site that runs the capability with issues.
 - » The site details page appears.
4. On the **Capabilities** tab, click the name of the Edge capability.
5. Click **Edit**.
6. In the **General** section, click the **Debug** drop-down menu and select *true*.

Note This field is by default set to *false*. If you set it to *true*, it will automatically revert to *false* after 24 hours.

7. Click **Save**.
 - » The fields become read-only.

8. Click **Run** to rerun the Edge capability.
 - » The log files are automatically sent to Collibra Data Intelligence Cloud.

Disable OpenTelemetry

If you reinstalled an Edge site with a new version it is possible that the new setup is not working due to a missing network connectivity. You would have to request for OpenTelemetry to be added again. If this request takes days to be completed, you may want to disable OpenTelemetry to still have a running Edge site.

Disable OpenTelemetry at installation time

Add the flag `--disable-otel` when you [run the installation script](#).

```
sudo sh install-master.sh --storage-path /var/edge/storage
properties.yaml \
  --disable-otel \
  -r registries.yaml
```

```
./run-installer-job.sh properties.yaml --repositories
repositories.json \
  --set collibra_edge.collibra.minio.persistence.size=120Gi \
  --disable-otel
```

Edge FAQ

The following table contains the most frequently asked questions about Edge that were not answered anywhere else in the Edge documentation.

Question	Answer
Who benefits from using Edge?	<p>All customers who want to ingest data into Collibra Data Intelligence Cloud benefit from Edge.</p> <p>Some of the benefits for using Edge are:</p> <ul style="list-style-type: none"> • Data is processed in the customer's secure environment and only the process results are sent to Collibra Data Intelligence Cloud. • Edge can automatically anonymize sensitive profiling data before sending it to Collibra Data Intelligence Cloud. • Edge can automatically classify the metadata and send the classification results together with the profiling results to Collibra Data Intelligence Cloud. • Edge enables better profiling performance, because data no longer has to be copied or moved. • Edge can execute capabilities in parallel, considering this is dependent of available resources. Jobserver only executes capability jobs sequentially.
Where can I find Edge API documentation?	<p>You can find the Edge API reference documentation in your Collibra environment on this URL: <a href="https://<your_collibra_platform_url>/edge/docs/index.html">https://<your_collibra_platform_url>/edge/docs/index.html</p>

Question	Answer
Does Edge replace the Jobserver?	<p>Customers can choose between Edge and Jobserver.</p> <p>The main differences between Edge and Jobserver are the following:</p> <ul style="list-style-type: none"> • Edge is based on Kubernetes, a distributed runtime, which means: <ul style="list-style-type: none"> ◦ It offers built in resource management. ◦ It has reliable delivery of results to Collibra Data Intelligence Cloud. • Edge is a Collibra service compatible with on-premises as well as cloud environments. • Edge offers continuous delivery of capability types and updates will be installed on a regular basis. • Edge updates are included in Collibra Data Intelligence Cloud releases. <p>Jobserver features correspond to Edge capabilities, each one is developed and deployed independently of one another. New capabilities will not be developed for Jobserver and it will be gradually phased out until early 2024. In the future, we will provide a script for migrating features from Jobserver to Edge where applicable.</p>
Can Edge run alongside Jobserver?	<p>Yes. Both can even be installed on the same server as long as the server has enough resources to support both, though we recommend not to run both services on a single server.</p>
What does the Edge architecture look like?	<p>You can see how Edge interacts with other components in this architecture and components overview.</p>

Question	Answer
<p>Can Edge use Kubernetes provided by a Cloud vendor, for example Google Kubernetes Engine (GKE), Azure Kubernetes Services (AKS) or Amazon Elastic Kubernetes Service (EKS)?</p>	<p>When the Edge site is installed in a Cloud environment, it does not use a managed Kubernetes provided by the Cloud vendor, because Kubernetes is already included in the Edge site installation process.</p> <p>You can install Edge on Amazon EKS. In the first releases, we cannot benefit from seamless integration of various Cloud services offered by those platforms, for example, embedded authentication, auto-scaling and databases. Edge on AKS and GKE are not part of the short term road map at this time. Please contact your Customer Success Manager if you have any questions.</p>
<p>Can Edge be installed on Windows servers?</p>	<p>No, you cannot install an Edge site on Windows servers. Support for K8S, K3S in particular, and container technology is underserved on Windows without the equivalent of a Linux sub-system. We will continue to prioritize your experience on Linux-based operating systems, and as such, will not support Edge installation on Windows servers until the support is seamless.</p>
<p>What are the supported data sources on Edge?</p>	<p>You can find the list of supported data sources in the Data sources supported by Edge section.</p>
<p>How does authentication from Edge to the customer's data sources work?</p>	<p>Authentication to data sources depends on the source type that the capability is connecting to. JDBC sources are covered via Edge connection providers. Other sources are accessed in different ways by capabilities themselves.</p>
<p>Can you connect using a cloud provider key manager such as AWS Secrets Manager, GCP Secret Manager or Azure Key Vault?</p>	<p>Not at this time.</p>
<p>Is CentOS Linux 8 supported for Edge installations?</p>	<p>Not for any versions of the Edge installation later than and including 2022.11. These later versions will require RedHat 8 in order to receive support. If you have an existing site, everything will work as before unless you need to reinstall a new site with a later version.</p>

Question	Answer
Why are you removing support for CentOS Linux 8?	CentOS Linux 8 has been made end-of-life. We are committed to using the latest technologies to ensure the best performance of our software, and as such RedHat 8 is required in order to receive support for Edge installations after the 2022.11 release.
How does Edge connect to Collibra Data Intelligence Cloud?	An Edge site is installed in the customer's environment, close to the data source. The Edge site communicates to Collibra Data Intelligence Cloud using an outbound HTTPS connection via port 443.
Is Edge on premises or in the Cloud?	Edge is always close to your data, and therefore can be on your premises or in a private or public Cloud setup.
Who controls Edge?	Edge is controlled by the customer through local access via the Collibra Data Intelligence Cloud user interface. You can also use local access via the Linux shell for advanced troubleshooting when Edge is unable to connect.
How is Edge updated?	Edge is updated automatically based on your Collibra Data Governance Center platform. The ability to disable automatic updates is currently on our road map, but is not currently supported with the available Edge installer.
Can an Edge site connect to more than one Collibra environment?	No. Every Edge site belongs and authenticates to only one Collibra Data Intelligence Cloud environment.

Question	Answer
<p>Do you need multiple instances of Edge for Data Quality to run?</p>	<p>It depends on your current setup. While you can technically run Collibra Data Quality & Observability and capabilities in the same Edge instance, you will need to ensure resources and space are available if you have a large Edge site.</p> <ul style="list-style-type: none"> • If have an existing Edge site that runs capabilities without Data Quality, you can update Edge Config to enable/disable any service or configuration during any run time, in order to provide space to run Data Quality. • If you have an existing Edge site and are open to reinstalling, then you can enable the Data Quality flags during the reinstallation process in order to keep one instance of Edge. <div data-bbox="683 857 1418 1055" style="background-color: #f0f0f0; padding: 10px; margin-top: 10px;"> <p>Note It is not recommended to run Classification and Data Quality in the same Edge instance, as they will compete for resources. Best practice is to have separate Edge sites for Classification and Data Quality.</p> </div>
<p>Can Edge use customer-provided certificates to connect to Collibra Data Intelligence Cloud?</p>	<p>Currently, we do not support this.</p> <p>Edge is a Collibra product that can run on the customer's on-premises or cloud environment. The authentication between the Edge site and Collibra Data Intelligence Cloud is controlled and secured by Collibra. The keys and credentials are generated when you install the Edge site.</p>
<p>When do internal K3S certificates expire?</p>	<p>The internal K3S certificates expire 12 months after the initial installation. You should restart the K3S-based Edge site in the last 3 months to ensure the internal certificates are rotated. If not, restart K3S or reinstall the Edge site.</p>
<p>Does Edge implement Cross-Site Request Forgery (CSRF) tokens?</p>	<p>Yes, the Edge management user interface can now implement CSRF tokens.</p> <div data-bbox="683 1686 1418 1816" style="background-color: #f0f0f0; padding: 10px; margin-top: 10px;"> <p>Note The CSRF token needs to be unique per user session and should be a large, random value.</p> </div>

Question	Answer
Does Edge support mTLS when connecting to Collibra Data Intelligence Cloud?	Currently, we do not support this.
Is Edge horizontally scalable?	Currently, Edge is not horizontally scalable. You cannot add more nodes.
Does Edge support High Availability and disaster recovery?	<p>Edge does not support High Availability, but core Edge services can be replicated if Edge is installed on a multi-node cluster, and Edge capabilities can be restarted in the event of a failure.</p> <p>Disaster recovery is supported through regular backups. More information about our disaster recovery process can be found in this overview.</p>
What troubleshooting information is collected and where is it stored?	<p>When Edge is operational and has deployed running capabilities, jobs or services, it can collect information on multiple levels:</p> <ul style="list-style-type: none"> • Infrastructure logs - default level info is collected, sent to the Cloud and accessible by Collibra. • Edge system monitoring - sent to the Cloud and accessible by Collibra. • Metadata connector logs - off by default and accessible by the customer . • Edge diagnostics - information is collected on demand by the customer on site and sent to Collibra as part of the support ticket.
<p>Edge Sample Data capability:</p> <ol style="list-style-type: none"> 1. Can everybody see sample data? 2. How is sample data queried from the database? 3. Which user account pulls the sample data from the database? 	<p>The Sample Data capability for Edge is a beta feature and needs to be activated.</p> <ol style="list-style-type: none"> 1. Only users with the permission will be able to view the sample data. 2. Samples are queried from the data source upon request. 3. The samples will be pulled from the database using the ID of the account specified in the Edge connection.

Question	Answer
Can metrics data from an Edge site be sent to Collibra through a private link instead of over the Internet?	No, this data can only be sent over the Internet.
What are Edge security considerations?	<p>Edge is designed around security first principles. Several highlights:</p> <ol style="list-style-type: none"> 1. No inbound connectivity - Edge site is always polling the platform via a REST endpoint. 2. Data is not stored on Edge after a job has finished. 3. Credentials are managed by Edge and not accessible outside of it. 4. Credentials on Edge site are encrypted with the key secured in the Collibra Data Governance Center. 5. Credentials can be updated both for data sources and Collibra Data Governance Center.
How are secrets stored on an Edge site?	You can find the details of how Edge stores secrets in this Storing secrets overview .