CDP Private Cloud Data Services 1.5.3

Installing CDP Private Cloud Data Services on the Embedded Container Service

Date published: 2023-12-16
Date modified: 2024-03-23

CLOUDERA

Legal Notice

Contents

Requirements

Software Support Matrix for ECS

This support matrix lists the supported software for the CDP Private Cloud Base cluster and the CDP Private Cloud Data Services containerized cluster when installing using the Embedded Container Service (ECS).

Base Cluster	Version	• Cloudera Manager 7.11.3 CHF 4 • 7.1.9 CHF 3 • 7.1.7 SP 2 • 7.1.8 CHF 19
	Base OS	• See Private Cloud Base OS requirements
	TLS	• AutoTLS (Custom CMCA) • AutoTLS (Self-signed)
	Kerberos	• AD • FreeIPA
	JDK	• openjdk version "11.0.21" 2023-10-17 LTS • OpenJDK Runtime Environment (Red_Hat-11.0.21.0.9-1) (build 11.0.21+9-LTS) • OpenJDK 64-Bit Server VM (Red_Hat-11.0.21.0.9-1) (build 11.0.21+9-LTS, mixed mode)
	Custom service principals	• Not supported
	Data Lake Storage	• HDFS • Ozone • Iceberg v2 (with HDFS and Ozone)
	Base DB (HMS access from CDW Data Services)	• MySQL 5.7, 8.0 • Maria DB 10.2, 10.3, 10.4, 10.5, 10.6 • Oracle 19.19 • Postgres 12, 14
Containerized Cluster	ECS OS	• RHEL 8.8, 8.9, 9.1 • CentOS 7.9
	Control Plane Metadata DB	• Embedded
	Vault	• Embedded
	Docker registry type	• Secure registry with self signed CA certs (pwd protected + self signed certs) • Embedded (ECS only. Not recommended)
	NFS	• Embedded • External
	IdP	• FreeIPA • ActiveDirectory (LDAP) • OpenLDAPs

	Network Access	• Airgap • Internet • HTTP proxy (CML)
	TLS	• Manual - CA signed • ESC server signed (ECS only)
GPU Nodes	OS	• CentOS 7.9 • RHEL 8.8

CDP Private Cloud Base Software Requirements

The software requirements for the nodes on which CDP Private Cloud Data Services are deployed are identical to CDP Private Cloud Base.

Your Private Cloud Base cluster must have the operating system, JDK, database, CDP components, and CDP Runtime version compatible with CDP Private Cloud Data Services. You must first set up the Private Cloud Base cluster, then you can install the Private Cloud Containerized cluster.

For more information about the requirements for the Private Cloud Base cluster, see CDP Private Cloud Base Requirements and Supported Versions and the Base Cluster section of the Software Support Matrix for ECS on page 4.

The following CDP Private Cloud Base cluster services are required to fully access the Data Services:

• Zookeeper
• HBase
• Hive Metastore (HMS)
• Hive on Tez (needed for using compaction)
• Ranger
• Atlas
• HDFS
• Ozone
• YARN
• Kafka
• Solr

In addition to this, the hive user should be able to create and list an Ozone bucket. For information about creating and listing ozone bucket, see *Managing buckets*.

Related Information
Managing buckets

CDP Private Cloud Data Services Hardware Requirements

Minimum and recommended hardware to successfully install and run Private Cloud Data Services.

In addition to the resources required for the Control Plane, additional resources will be required depending on the Data Service(s) you intend to run. Minimum and recommended additional resource requirements for each of the Data Services can be found in the pages below. To calculate the total minimum or recommended resource requirements for your CDP Private Cloud Data Services cluster, add the resources required for the Control Plane to the total minimum or recommended additional resources for your chosen Data Service(s).

You can also use the CDP Private Cloud Data Services Spreadsheet to model the number and specification of hosts required for a deployment. See How to use the CDP Private Cloud Data Services sizing spreadsheet on page 8.

Requirements for HA and Non-HA Control Plane

Standard resource mode requirements for standalone HA and Non-HA Control Plane.

Component	Minimum	Recommended
Node Count	1 (Non-HA)	3 (HA)
CPU	16 cores	32 cores (per node)
Memory	32 GB	64 GB (per node)
Storage	300 GB	1 TB (per node)
Network Bandwidth	1GB/s to all nodes and base cluster	1GB/s to all nodes and base cluster

Additional resource requirements for Cloudera Data Warehouse

Standard resource mode requirements for Cloudera Data Warehouse.

The following table lists the minimum and recommended compute (processor), memory, storage, and network bandwidth required for each OpenShift or ECS worker node using the Standard Resource Mode for production use case. Note that the actual node still needs some extra resources to run the operating system, Kubernetes engine, and Cloudera Manager agent on ECS.

Component	Minimum	Recommended
Node Count	4	10
CPU per worker	16 cores [or 8 cores or 16 threads that have Simultaneous Multithreading (SMT) enabled]	32+ cores (can also be achieved by enabling SMT)
Memory per worker	128 GB per node	384 GB* per node
FAST (Fully Automated Storage Tiering) Cache - Locally attached SCSI device(s) on every worker. Preferred: NVMe and SSD. OCP uses Local Storage Operator. ECS uses Local Path Provisioner.	1.2 TB* SATA, SSD per host	1.2 TB* NVMe/SSD per host
Network Bandwidth	1 GB/s guaranteed bandwidth to every CDP Private Cloud Base node	10 GB/s guaranteed bandwidth to every CDP Private Cloud Base node

⚠️ **Important:** When you add memory and storage, it is very important that you add it in the increments as follows:

- Increments of 128 GB of memory
- Increments of 600 GB of locally attached SSD/NVMe storage

If you add memory or storage that is not in the above increments, the memory and storage that exceeds these increments is not used for executor pods. Instead, the extra memory and storage can be used by other pods that require fewer resources.

For example, if you add 200 GB of memory, only 128 GB is used by the executor pods. If you add 2 TB of locally attached storage, only 1.8 TB is used by the executor pods.

Additional resource requirements for Cloudera Data Engineering

For standalone Cloudera Data Engineering, Cloudera recommends three nodes (one master and two workers) with the following minimum memory, storage, and hardware requirements for each node:

---

[*] Depending on the number of executors you want to run on each physical node, the per-node requirements change proportionally. For example, if you are running 3 executor pods per physical node, you require 384 GB of memory and approximately 1.8TB (600GB per executor) of locally attached SSD/NVMe storage for FAST Cache.

Component	Minimum	Recommended
Node Count	2	4
CPU	16 cores for CDE workspace (base and virtual cluster) and 8 cores for workload	16 cores for CDE workspace (base and virtual cluster) and 32 cores (you can extend this depending upon the workload size)
Memory	64 GB for CDE workspace (base and virtual cluster) and 32 GB (you can extend this depending upon the workload size)	64 GB for CDE workspace (base and virtual cluster) and 64 GB (you can extend this depending upon the workload size)
Storage	200 GB blob storage and 500 GB NFS storage	200 GB blob storage and 500 GB NFS storage
Network Bandwidth	1 GB/s to all nodes and base cluster	10 GB/s to all nodes and base cluster

> ⚠️ **Important:** Optionally, if you want to use GPU in Spark, the Spark RAPIDS library is validated and certified by Nvidia for *NVIDIA P100, V100, T4 and A2/A10/A30/A100* GPU architecture.

CDE Service and Virtual Cluster requirements

- CDE Service requirements: Overall for a CDE service, it requires 110 GB Block PV or NFS PV, 7 CPU cores, and 15 GB memory.

Table 1: CDE Service requirements:

Component	vCPU	Memory	Block PV or NFS PV	Number of replicas
Embedded DB	4	8 GB	100 GB	1
Config Manager	500 m	1 GB	--	2
Dex Downloads	250 m	512 MB	--	1
Knox	250 m	1 GB	--	1
Management API	1	2 GB	--	1
NGINX Ingress Controller	100 m	90 MB	--	1
FluentD Forwarder	250 m	512 MB	--	1
Grafana	250 m	512 MB	10 GB	1
Data Connector	250 m	512 MB	--	1
Total	7	15 GB	110 GB	

- CDE Virtual Cluster requirements:

  - For Spark 3: Overall storage of 400 GB Block PV or Shared Storage PV, 5.35 CPU cores, and 15.6 GB per virtual cluster.
  - For Spark 2: If you are using Spark 2, you need additional 500 m CPU, 4.5 GB memory and 100 GB storage, that is, the overall storage of 500 GB Block PV or Shared Storage PV, 5.85 CPU cores, and 20.1 GB per virtual cluster.

> ⚠️ **Important:** The CDE service and virtual cluster requirements does not include workloads. See the below workload information on the additional resources based on workload.

Table 2: CDE Virtual Cluster requirements for Spark 3:

Component	vCPU	Memory	Block PV or NFS PV	Number of replicas
Airflow API	350 m	612 MB	100 GB	1
Airflow Scheduler	1	1 GB	100 GB	1
Airflow Web	250 m	512 MB	--	1

Component	vCPU	Memory	Block PV or NFS PV	Number of replicas
Runtime API	250 m	512 MB	100 GB	1
Livy	3	12 GB	100 GB	1
SHS	250 m	1 GB		1
Pipelines	250 m	512 MB	--	1
Total	5350 m	15.6 GB	400 GB	

- Workloads: Depending upon the workload, you must configure resources.

  - The Spark Driver container uses resources based on the configured driver cores and driver memory and additional 40% memory overhead.
  - In addition to this, Spark Driver uses 110 m CPU and 232 MB for the sidecar container.
  - The Spark Executor container uses resources based on the configured executor cores and executor memory and additional 40 % memory overhead.
  - In addition to this, Spark Executor uses 10 m CPU and 32 MB for the sidecar container.
  - Minimal Airflow jobs need 100 m CPU and 200 MB memory per Airflow worker.

Additional resource requirements for Cloudera Machine Learning

Standard resource mode requirements for standalone Cloudera Machine Learning. Node count should not be a limiting factor assuming the other memory and CPU minimums are reached.

Component	Minimum	Recommended
Node Count	1	1 per workspace + additional nodes depending on expected user workloads
CPU	32 Cores Per Workspace+ additional Cores depending on expected user workloads	32 Cores Per workspace + additional Cores depending on expected user workloads
Memory	128 GB + additional memory depending on the expected workloads	256 GB Per Workspace + additional memory depending on the expected workloads
Storage	600 GB Block storage + 1000 GB NFS storage (Block if internal and NFS if external)	4500 GB Block storage + External NFS with 1000 GB NFS storage minimum + additional storage based on sizing of project files
Network Bandwidth	1GB/s to all nodes and base cluster	1GB/s to all nodes and base cluster

Additional Resources for User Workloads:

Component	Minimum	Recommended
CPU	1 Core per concurrent workload	2–16 cores per concurrent workload (dependent on use cases)
Memory	2 GB per concurrent workload	4–64 GB per concurrent workload (dependent on use cases)

How to use the CDP Private Cloud Data Services sizing spreadsheet

You can use the sizing spreadsheet to model the hardware requirements for a CDP Private Cloud Data Services deployment.

Overview

The CDP Private Cloud Data Services Sizing spreadsheet is a spreadsheet that you can use to model the quantity and specifications for worker hosts required in a CDP Private Cloud Data Services deployment.

This spreadsheet is intended to use information about workloads you are planning to run and hardware specifications for worker nodes to arrive at an approximate number of worker nodes required for your deployment. Due to the complexity of estimating workloads, Cloudera recommends you review any sizing or purchasing decisions with Cloudera Professional Services before committing to those decisions.

How to access the spreadsheet

You can access the spreadsheet here: CDP Private Cloud Data Services Sizing. The file is in Microsoft Excel format. You can open the file in Excel, or upload it to Google Sheets.

There are three tabs in the spreadsheet. You will make your inputs only on the Worker Node Totals tab. Do not modify the following tabs (these tabs contain data used to calculate values in the spreadsheet and should not be modified):

- Component Lookup
- K8s Resources

⚠️ **Important:** Do not modify any cells except for the ones indicated below. Modifying the formulas in other cells will result in inaccurate calculations.

Workload inputs

The spreadsheet calculates the total amount vcores, RAM, and storage required based on information you enter about the combined workloads you intend to deploy. Then based on the hardware specifications entered, calculates the number of worker nodes required, which is displayed in cell E24.

The following sections describe values you must enter into the spreadsheet. Values are required for each Data Service you intend to deploy, and values to enter for the hardware specifications for your worker nodes.

Control plane monitoring

Label	Cell	Description
CP Monitoring	B3	Increment this number by one for each environment.

Cloudera Data Warehouse (CDW)

If you will deploy CDW, on the Worker Node Totals tab, enter the following information:

Label	Cell	Description
CDW Data Catalog (min 1 per env)	B5	Enter the number of Data Catalogs you will need in your deployment. You must have at least one Data Catalog.
CDW LLAP warehouses	B6	Enter the number of LLAP warehouses you will need for each Virtual Warehouse in your deployment.
-- LLAP Executors	B7	Enter the total number of LLAP Executors you will need in your deployment.
CDW Impala warehouses	B8	Enter the number of CDW Impala warehouses for each Virtual Warehouse you will need in your deployment.
-- Impala Coordinators (2 x for HA)	B9	Enter the number of Impala Warehouses you will need in your deployment. If you have enabled high availability, enter twice the number of Warehouses.
-- Impala Executors	B10	Enter the number of Impala Executors you will need in your deployment.
CDW Cache	B11	Enter the amount of CDW Cache space for each coordinator and executor (Default 600)
Data Viz - small instances	B12	Enter the size selected when creating a Data Visualization instance.
Data Viz - medium instances	B13	
Data Viz - large instances	B14	

For more information about sizing Cloudera Data Warehouse deployments, see:

- (OCP) CDE hardware requirements.
- (ECS) Additional resource requirements for Cloudera Data Engineering

Cloudera Machine Learning (CML)

Sizing for a CML deployment depends on the number of concurrent jobs you expect to run and the number of Workspaces you provision.

Label	Cell	Description
CML Workspace (min of 1)	B16	Enter the number of workspaces you need in your deployment.
-- CML Small concurrent sessions	B17	Enter the number of concurrent small-sized sessions you intend to run.
-- CML Average concurrent sessions	B18	Enter the number of concurrent average-sized sessions you intend to run.

For more information about sizing the Cloudera Data Engineering service, see the following topics:

- Additional resource requirements for Cloudera Machine Learning.
- (OCP) Cloudera Machine Learning requirements
- (ECS) Cloudera Machine Learning requirements

Cloudera Data Engineering (CDE)

Label	Cell	Description
CDE Service (min/max 1 per cluster)	B20	Enter the number of CDE clusters you will need in your deployment.
CDE Virtual Cluster	B21	Enter the number of CDE Virtual Clusters you will need in your deployment.
-- CDE Small concurrent jobs	B22	Enter the number of concurrent small-sized jobs you intend to run.
-- CDE Average concurrent jobs	B23	Enter the number of concurrent average-sized jobs you intend to run.

For more information about sizing the Cloudera Data Engineering service, see Additional resource requirements for Cloudera Data Engineering.

Worker node hardware specifications

Based on the inputs you supplied for your workloads, the spreadsheet totals the number of vcores, RAM, and storage required for the cluster in cells C20-C26. Then, based on the worker node hardware specifications you enter in cells B26-B29, divides the totals for vcores, RAM and storage by each of the worker node specifications to arrive at the required number of nodes for vcores, RAM and storage shown in cells D5-D29. The final number, in cell E27 chooses the higher value of these cells.

You may notice that the calculated values in cells D26 and D27 are different. This indicates that some nodes are oversubscribed for RAM or vcores. Adjust the hardware specifications for CPU and RAM until the two cells are closer together in value. Changing these values may also change the calculated number of worker nodes.

Label	Cell	Description
CPU recommend 40+ cores (80 vcores)	B27	Enter the number of vcores for each worker node.
RAM (GB) recommend 415 GB RAM	B28	Enter the amount of RAM, in gigabytes, for each worker node.

Label	Cell	Description
Disk (GB) Block (OCP CSI block, ECS Longhorn)	B29	Enter the number of gigabytes Block required for: - OpenShift Container Platform: CSI block - Embedded Container Service: ECS Longhorn
Disk (GB) Fast Cache for CDW (nvme,ssd)	B30	Enter the number of gigabytes of Fast Cache used in Cloudera Data Warehouse.
CP Block Overhead per host (300 to 1024)	B31	Enter the Control Plane block overhead
NFS (GB) (choose 1 from below)	B33	Enter required storage in either cell B34 or cell B35
-- Embedded nfs - (subtract from Block provider) non-prod	B34	Enter the number of gigabytes storage for an embedded NFS.
-- External nfs	B35	Enter the number of gigabytes of storage for an External NFS.
ECS Master Node requires 1 for non HA - 3 for HA If you are using the Embedded Container Service, you will also need to provision a host for the ECS Master Node (a node running the ECS Server component). The values described here contain Cloudera's recommendations for specifications for the ECS Master node.	B38	Minimum: 16 vcores Recommended: 32 vcores
	B39	Minimum: 32 GB RAM Recommended: 64 GB RAM
	B40	Minimum: 300 GB HDD (This amount is adequate for a proof-of-concept cluster.) Recommended: 1 TB HDD

Docker repository access

You must ensure that the cluster has access to the Docker Container Repository in order to retrieve the container images for deployment.

There are several types of Docker Repositories you can use:

Embedded Repository

> During installation, a Docker daemon is provisioned to act as the Repository. Passwords and certificates are auto generated. No additional set up is needed. Images are copied to the repository during installation. During upgrades, only the new and changed images are copied. Copying images generally takes one to two hours.

> It is important to note that the Embedded Repository can be a single point of failure. If the node that runs the Docker Repository fails or becomes unavailable, some cluster functionalities might become unavailable. Moving the Docker Repository to another node is a complex process and will require engaging Cloudera Professional Services.

Cloudera Repository

> Using the Cloudera Repository requires that the cluster have internet connectivity to the Cloudera public repository. Using the Cloudera Repository is the fastest option.

> The Cloudera-hosted Docker Repository option may increase the time required to deploy or start the services in the cluster. Cloudera generates Docker Repository credentials that are identical to your paywall credentials. Refer to your welcome letter for the credentials or use the credential generator on cloudera.com to generate credentials from your license key.

This option is best suited for proof-of-concept, non-production deployments or deployments that do not have security requirements that disallow internet access.

Custom Repository

A Custom Repository is a repository that you manage in your environment and can be Enterprise grade and highly available.

During installation and upgrade, a custom script is generated that you use to copy the images. Copying images can take 4 - 5 hours.

Only TLS-enabled custom Docker Registry is supported. Ensure that you use a TLS certificate to secure the custom Docker Registry. The TLS certificate can be self-signed, or signed by a private or public trusted Certificate Authority (CA).

> ⚠️ **Important:** When using an Embedded Container Service cluster, passwords must not contain the $ character.

CDP Private Cloud Data Services Software Requirements

This release ships with Cloudera Manager 7.11.3 CHF 4. If you have an existing CDP Private Cloud Base cluster set up using an earlier version of Cloudera Manager, you must first upgrade Cloudera Manager to version 7.11.3 CHF 4.

For more information about specific software requirments, see the Software Support Matrix for ECS on page 4.

Additionally, you must perform the following:

- For CML, you must install nfs-utils in order to mount longhorn-nfs provisioned mounts. The nfs-utils package is required on every node of the ECS cluster. Run this command `yum install nfs-utils` to install nfs-utils.
- If you have nodes with GPU, ensure that the GPU hosts have nVidia Drivers and nvidia-container-runtime installed. You must confirm that drivers are properly loaded on the host by executing the command `nvidia-smi`. You must also install the nvidia-container-toolkit package.
- You must have a minimum of one agent node for ECS.
- Set up Kerberos on these clusters using an Active Directory.
- Enable TLS on the Cloudera Manager cluster for communication with components and services.
- If you do not have entitlements, contact your Cloudera account team to get the necessary entitlements.
- The default docker service uses /docker folder. Whether you wish to retain /docker or override /docker with any other folder, you must have a minimum of 300 GiB free space.
- Ensure that all of the hosts in the ECS cluster have more than 300 GiB of free space in the /var/lib directory at the time of installation.
- The cluster generates multiple hosts and host based routing is used in the cluster in order to route it to the right service. You must decide on a domain for the services which Cloudera Manager by default points to one of the host names on the cluster. However, during the installation, you should check the default domain and override the default domain (only if necessary) with what you plan to use as the domain. The default domain must have a wildcard DNS entry. For example, "*.apps.myhostname.com".
- It is recommended that you leave IPv6 enabled at the OS level on all ECS nodes.
- You must install `nvidia-container-toolkit`. (`nvidia-container-runtime` migrated to `nvidia-container-toolkit`, see Migration Notice.) The steps for this are shown in the NVIDIA Installation Guide. If using Red Hat Enterprise Linux (RHEL), use `dnf` to install the package. For an example with RHEL 8.7, see Installing the NVIDIA Container Toolkit.
- Python 3.8 is required for Cloudera Manager version 7.11.3.0 and higher versions. Cloudera Manager agents will not start unless Python 3.8 is installed on the cluster nodes.

Related Information

Software Support Matrix for ECS

Installation using the Embedded Container Service (ECS)

Preparing CDP Private Cloud Base

Use Cloudera Manager to configure your Private Cloud Base cluster in preparation for the Private Cloud Data Services installation.

1. Configure the Private Cloud Base cluster to use TLS.Configuring TLS Encryption for Cloudera Manager Using Auto-TLS.

2. Configure Cloudera Manager with a JKS-format (not PKCS12) TLS truststore.Database requirements.

3. Configure Cloudera Manager to include a root certificate that trusts the certificate for all Cloudera Manager server hosts expected to be used with the Private Cloud, LDAP server (if you are using LDAP), and the Postgres DB of all Hive Metastores that you use with Private Cloud. If a single CA is used to sign all of them, then just that single CA must be imported.

   a. Import the necessary certificates into the truststore configured in Configure Administration > Settings > Security > Cloudera Manager TLS/SSL Client Trust Store File.

4. Enable Kerberos for all the services in the cluster.Enabling Kerberos for authentication.

5. Configure Ranger and LDAP for user authentication. Ensure that you have configured Ranger user synchronization.Configure Ranger authentication for LDAP and Ranger usersync.

6. Configure LDAP using Cloudera Manager. Only Microsoft Active Directory (AD) and OpenLDAP are currently supported.Configure authentication using an LDAP-compliant identity service.

7. Check if all the running services in the cluster are healthy. To check this using Cloudera Manager, go to Cloudera Manager > Clusters > [***CLUSTER NAME***] > Health Issues. If there are no health issues, the No Health Issues message is displayed.

8. If you want to reuse data from your legacy CDH or HDP deployment in your Private Cloud, copy the data from your CDH or HDP deployments into the CDP Private Cloud Base cluster that will be accessed by CDP Private Cloud Data Services. For more information about data migration, see the Data Migration Guide.

9. For installing CDP Private Cloud Base, see Install CDP Private Cloud Base

CDP Private Cloud Base checklist

Use this checklist to ensure that your CDP Private Cloud Base is configured and ready for installing CDP Private Cloud Data Services.

Note: The Cloudera Manager mentioned in this checklist is the CDP Private Cloud Base Cloudera Manager using which you want to install CDP Private Cloud Data Services.

Table 3: CDP Private Cloud Base checklist to install CDP Private Cloud Data Services

Item	Summary	Documentation	Notes
Runtime components	Ensure that you have Ranger, Atlas, Hive, HDFS, and Ozone installed in your CDP Private Cloud Base cluster.	• Software Support Matrix for ECS on page 4 • CDP Private Cloud Base requirements	If you do not install these components, you see an error when creating an environment in CDP Private Cloud Data Services.
Network requirement	Ensure that all the network routing hops in production. Cloudera recommends not to use more than 4:1 oversubscription between the spine-leaf switches.		

Item	Summary	Documentation	Notes
Cloudera Manager database requirement	Refer to the the CDP Private Cloud Base database requirements.	• Database Requirements • Cloudera Support Matrix	N/A
Cloudera Manager TLS configuration	Ensure that Cloudera Manager in the CDP Private Cloud Base cluster is configured to use TLS.	Configuring TLS Encryption for Cloudera Manager Using Auto-TLS	You can also manually configure TLS to complete this task. See Manually Configuring TLS Encryption for Cloudera Manager
Cloudera Manager JKS-format TLS truststore	Ensure that the Cloudera Manager is configured with a JKS-format (not PKCS12) TLS truststore.	Obtain and Deploy Keys and Certificates for TLS/SSL	N/A
Cloudera Manager truststore and root certificate	Ensure that the Cloudera Manager truststore contains a root certificate that trusts the certificate for all Cloudera Manager server hosts used with CDP Private Cloud Data Services.	How to Add Root and Intermediate CAs to Truststore for TLS/SSL	Import the necessary certificates into the truststore configured in Configure Administration > Settings > Security > Cloudera Manager TLS/SSL Client Trust Store File .
LDAP configuration	Ensure that you configure LDAP using Cloudera Manager.	N/A	Only Microsoft Active Directory (AD) and OpenLDAP are currently supported.
Apache Ranger configuration for LDAP	Ensure that the CDP Private Cloud Base cluster is configured with Apache Ranger and LDAP for user authentication.	Configure Ranger authentication for LDAP	N/A
Apache Ranger usersync configuration	Ensure that you have configured Apache Ranger and Apache Ranger usersync.	Ranger usersync	Apache Ranger user synchronization is used to get users and groups from the corporate ActiveDirectory to use in policy definitions.
Kerberos configuration	Ensure that Kerberos is enabled for all services in the cluster.	Enabling Kerberos for authentication	Custom Kerberos principals are not currently supported.
Internet access or air gap installation	Ensure that CDP Private Cloud Base and the ECS hosts have access to the Internet. If you do not have access to the Internet, you must do an air gap installation.	Install CDP Private Cloud Data Services in air gap environment	You need access to the Docker registries and the Cloudera repositories during the installation process.
Services health check	Ensure that all services running in the cluster are healthy.	Cloudera Manager Health Tests	N/A
CDP Private Cloud entitlement	Ensure that you have the necessary CDP entitlement from Cloudera to access the Private Cloud installation.	N/A	
Reuse data from CDH or HDP (Optional)	To reuse data from your legacy CDH or HDP deployment in your Private Cloud, ensure that you have migrated that data into your CDP Private Cloud Base. You must be using Cloudera Runtime 7.1.7 for migrating data from your CDH or HDP cluster.	Data Migration Guide	N/A

Item	Summary	Documentation	Notes
(Recommended) Configure HDFS properties to optimize logging	CDP uses "out_webhdfs" Fluentd output plugin to write records into HDFS, in the form of log files, which are then used by different data services to generate diagnostic bundles. To optimize the size of logs that are captured and stored on HDFS, you must update a few HDFS configurations in the hdfs-site.xml file using Cloudera Manager.	Configuring HDFS properties to optimize logging	N/A

Embedded Container Service (ECS) checklist

Use this checklist to ensure that your Embedded Container Service (ECS) is configured and ready for installing CDP Private Cloud Data Services.

Table 4: Embedded Container Service (ECS) checklist to install CDP Private Cloud Data Services

Item	Summary	Documentation	Notes
DNS configuration	Ensure that you have set up the DNS and Reverse DNS between Embedded Container Service (ECS) hosts and CDP Private Cloud Base. This is required for obtaining Kerberos ticket-granting tickets.	N/A	A wildcard DNS entry is required for resolving the ingress route for applications. The ingress route is usually behind a load balancer.
Check that ECS Ingress can be resolved in DNS.	Ensure that Embedded Container Service (ECS) application hostnames can be accessed from outside the cluster. You can test this by creating an ingress point on the target cluster.	The cluster generates multiple hosts and host-based routing is used in the cluster in order to route it to the right service. You must decide on a domain for the services which Cloudera Manager, by default points to one of the hostnames on the cluster. However, during the installation, you should check the default domain and override the default domain (only if necessary) with what you plan to use as the domain. The default domain must have a wildcard DNS entry. For example, *.apps.myhostname.com.	Perform a DNS query on the ingress point, to check if you can access the hostnames outside the cluster.
Clock time from NTP source	Ensure that the NTP clock in CDP Private Cloud Base is in sync with the time configured in the Embedded Container Service (ECS) cluster. This is an important step if your setup does not have access to the Internet.	Enable an NTP Service	Installing CDP Private Cloud Data Services (ECS)

Adding a CDP Private Cloud Data Services cluster

Using Cloudera Manager 7.11.3 CHF 4, you can either install Private Cloud Data Services by downloading the repository from the Internet, or you can do an air gap installation if Cloudera Manager does not have access to the Internet.

Before you begin:

- Ensure that you have Cloudera Manager 7.11.3 CHF 4 installed and you have the entitlements to the CDP Private Cloud Data Services product.
- Python 3.8 is required for Cloudera Manager version 7.11.3.0 and higher versions. Cloudera Manager agents will not start unless Python 3.8 is installed on the cluster nodes.
- Only TLS 1.2 is supported for authentication with Active Directory/LDAP. You require TLS 1.2 to authenticate the CDP control plane with your LDAP directory service like Active Directory.
- The Kubeconfig file is available in /etc/rancher/rke2/rke2.yaml
- If the installer fails, do not cancel the installation. For more information, see Manually uninstalling ECS from a cluster.
- Do not use any antivirus or other security tools on the ECS nodes. These third-party tools may cause issues with ECS functionality.

Installing CDP Private Cloud Data Services using ECS

Follow the steps in this topic to install CDP Private Cloud Data Services with the Embedded Container Service (ECS).

Procedure

1. If your ECS hosts are running the CentOS 8.4, OEL 8.4, RHEL 7.9, or RHEL 8 operating systems, you must install iptables on all the ECS hosts.

   For CentOS 8.4, OEL 8.4, or RHEL 8, run the following command on each ECS host:

   ```
   yum --setopt=tsflags=noscripts install -y iptables
   ```

   For RHEL 7.9, run the following command on each ECS host:

   ```
   yum install -y iptables
   ```

2. If you are installing ECS on RHEL 8:
   a) Add the hosts you intend to use for ECS to Cloudera Manager, without specifying a cluster. See Add New Hosts To Cloudera Manager.
   b) If you are using RHEL 8, and if the nm-cloud-setup.service and nm-cloud-setup.timer services are enabled, disable them by running the following command on each host you added:

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

   For more information, see Known issues and limitations.
   c) If you disabled the nm-cloud-setup.service and nm-cloud-setup.timer services, reboot the added hosts.

3. In Cloudera Manager, click Data Services in the left menu.

The Add Private Cloud Containerized Cluster page appears. Click Continue.

> **Note:** You can also click Add Add Cluster at the top right in Cloudera Manager, then select Private Cloud Containerized Cluster as the cluster type.

4. On the Getting Started page of the installation wizard, select Internet or Air Gapped as the Install Method.

Internet install method (To use a custom repository link provided to you by Cloudera, click Custom Repository) :

If you select the Air Gapped install option, extra steps are displayed. Follow these steps to download and mirror the Cloudera archive URL using a local HTTP server.

a. Download everything under https://archive.cloudera.com/p/cdp-pvc-ds/latest

```
wget -l 0 --recursive --no-parent -e robots=off -nH --cut-dirs=2 --rejec
t="index.html*" -t 10 https://<username>:<password>@archive.cloudera.com
/p/cdp-pvc-ds/latest
```

b. Edit the manifest.json file in the downloaded directory. Change "http_url": "..." to

"http_url":   "http://your_local_repo/cdp-pvc-ds/latest"

c. Mirror the downloaded directory to your local http server, e.g. http://your_local_repo/cdp-pvc-ds/latest

d. Click Custom Repository and add http://your_local_repo/cdp-pvc-ds/latest as a custom repository.

e. Click the Select Repository drop-down and select http://your_local_repo/cdp-pvc-ds/latest

Click Continue.

5. On the Cluster Basics page, type a name for the Private Cloud cluster that you want to create in the Cluster Name field. From the Base Cluster drop-down list, select the cluster that has the storage and SDX services that you want this new Private Cloud Data Services instance to connect with. Click Continue.

Add Private Cloud Containerized Cluster
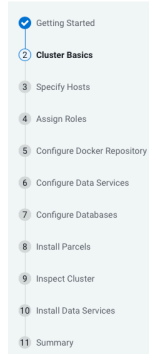
✔ Getting Started	Cluster Basics
② **Cluster Basics**	Cluster Name Containerized Cluster 1
③ Specify Hosts	
④ Assign Roles	
⑤ Configure Docker Repository	**Private Cloud Containerized Cluster**
⑥ Configure Data Services	A Private Cloud Containerized Cluster helps you to install and run CDP Private Cloud Data Services such as Machine Learning and Data Warehouse with data from an existing Base Cluster. Learn more at CDP Private Cloud Containerized Cluster.
⑦ Configure Databases	Base Cluster Cluster 1 (7.1.7)
⑧ Install Parcels	☐ Use Default Configuration
⑨ Inspect Cluster	Use embedded Docker Repository, Vault and Database with default settings, and use default configurations for Role Assignments. Not recommended for production.
⑩ Install Data Services	
⑪ Summary	

6. On the Specify Hosts page, hosts that have already been added to Cloudera Manager are listed on the Currently Managed Hosts tab. You can select one or more of these hosts to add to the ECS cluster.

You can also click the New Hosts tab to specify one or more hosts that have not been added to Cloudera Manager. Enter a Fully Qualified Domain Name in the Hostname box, then click Search.

> **Note:** Click the pattern link under the Hostname box to display more information about allowed FQDN patterns.

CLOUDERA
Manager

Add Private Cloud Containerized Cluster

CDEP Deployment from 2020-Oct-23 11:35

- ✓ Getting Started
- ✓ Cluster Basics
- ③ **Specify Hosts**
- ④ Select JDK
- ⑤ Enter Login Credentials
- ⑥ Install Agents
- ⑦ Assign Roles
- ⑧ Configure Docker Repository
- ⑨ Configure Data Services
- ⑩ Configure Databases
- ⑪ Install Parcels
- ⑫ Inspect Cluster
- ⑬ Install Data Services
- ⑭ Summary

Specify Hosts

Currently Managed Hosts (3/3 Selected)   **New Hosts (3 Selected)**

Hosts should be specified using the same hostname (FQDN) that they will identify themselves with.

| Hostname | dh-centos79um-[1-3].vpc.cloudera.com |

Hint: Search for hostnames or IP addresses using pattern ☐

SSH Port   22   **Search**

3 hosts scanned, 3 running SSH.

	Expanded Query	Hostname (FQDN) ↑	IP Address	Currently Managed	Result
✓	dh-centos79um-1.vpc.cloudera.com	dh-centos79um-1.vpc.cloudera.com	10.65.198.225	No	Host was successfully scanned.
✓	dh-centos79um-2.vpc.cloudera.com	dh-centos79um-2.vpc.cloudera.com	10.65.195.145	No	Host was successfully scanned.
✓	dh-centos79um-3.vpc.cloudera.com	dh-centos79um-3.vpc.cloudera.com	10.65.193.53	No	Host was successfully scanned.

1 - 3 of 3

🗄 Parcels

⌛ Running Commands

⊗ Support

Ⓐ admin

7.11.3   ≪

Cancel   ← Back   **Continue →**

After you have finished specifying the ECS hosts, click Continue.

7. On the Select JDK page, select any one from the below options:

a) Manually manage JDK

b) Install a Cloudera-provided version of OpenJDK

c) Install a system-provided version of OpenJDK

8. On the Enter Login Credentials page, All hosts accept the same password is selected by default. Enter the user name in the SSH Username box, and type in and confirm the password. You can also select the All hosts accept the same private key option and provide the Private Key and passphrase.

9. The Install Agents page appears.

Add Private Cloud Containerized Cluster

✓ Getting Started	**Install Agents**
✓ Cluster Basics	**Installation in progress.**
✓ Specify Hosts	
✓ Select JDK	0 of 3 host(s) completed successfully. Abort Installation
✓ Enter Login Credentials	
⑥ **Install Agents**	
⑦ Assign Roles	
⑧ Configure Docker Repository	
⑨ Configure Data Services	
⑩ Configure Databases	
⑪ Install Parcels	
⑫ Inspect Cluster	
⑬ Install Data Services	
⑭ Summary	

Hostname	IP Address	Progress	Status	
	10.65.6.9		C Installing openjdk8 package...	Details
	10.65.10.73		C Installing openjdk8 package...	Details
	10.65.9.254		C Installing openjdk8 package...	Details

10. On the Assign Roles page, you can customize the roles assignment for your new Private Cloud Containerized cluster.

> ⚠️ **Important:** Cloudera does not recommend altering assignments unless you have specific requirements such as having selected a specific host for a specific role.
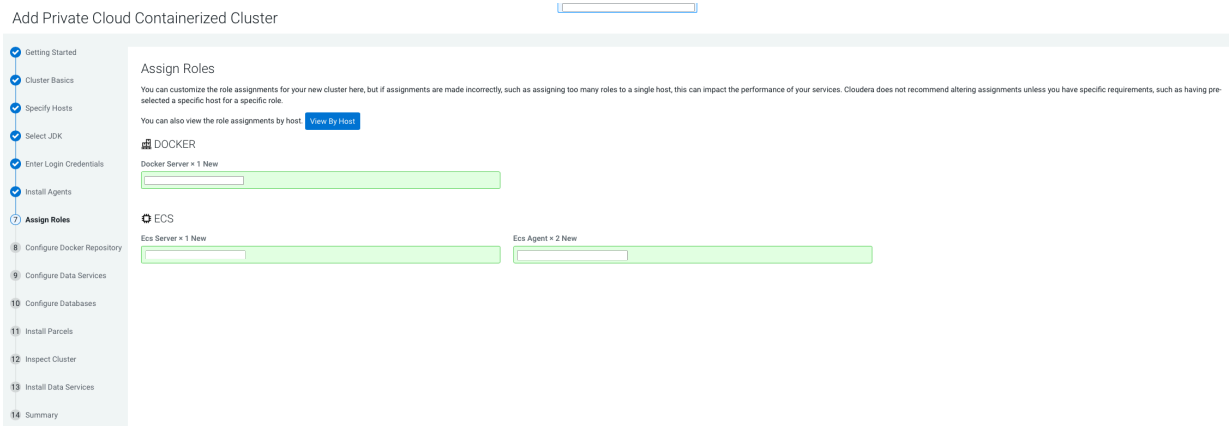
Single node ECS installation is supported, but is only intended to enable CDSW to CML migration. If you are installing ECS on a single node, only the Docker and ECS Server roles are assigned. The ECS Agent role is not required for single node installation.
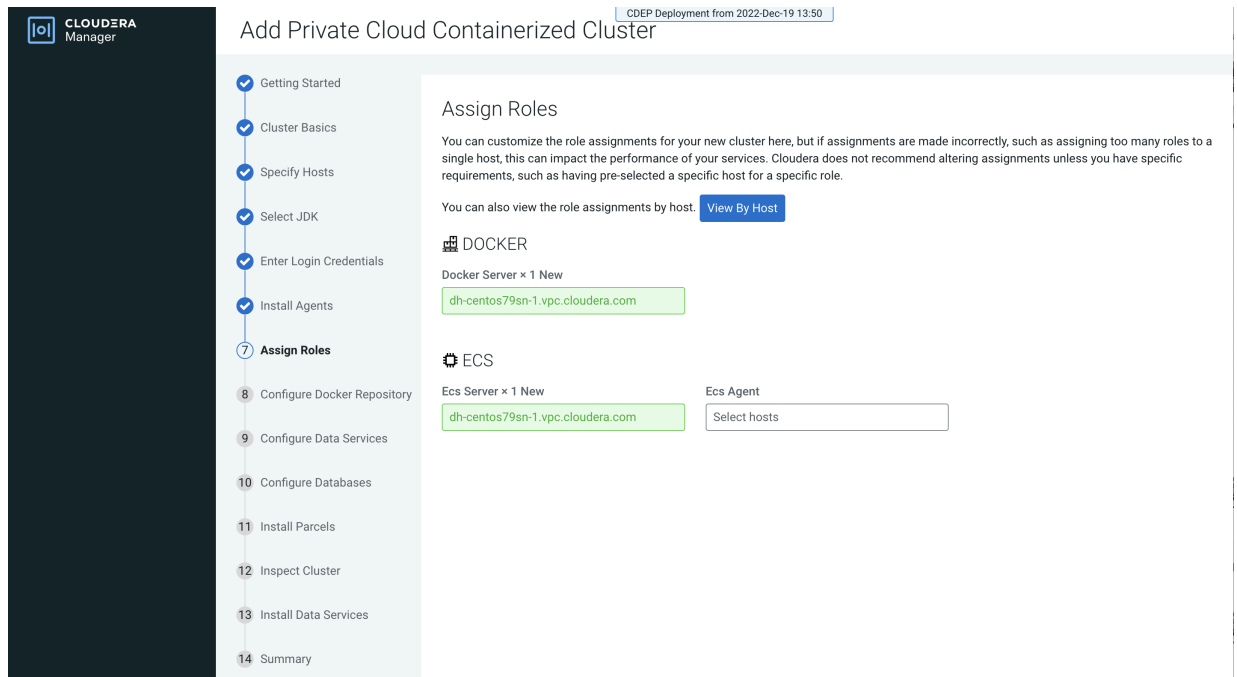
Click Continue.

11. Configure a Docker Repository.

There are several options for configuring a Docker Repository. For more information about these options, see Docker repository access on page 11.

Add Private Cloud Containerized Cluster

✓ Getting Started

✓ Cluster Basics

✓ Specify Hosts

✓ Select JDK

✓ Enter Login Credentials

✓ Install Agents

✓ Assign Roles

⑧ **Configure Docker Repository**

⑨ Configure Data Services

⑩ Configure Databases

⑪ Install Parcels

⑫ Inspect Cluster

⑬ Install Data Services

⑭ Summary

Configure Docker Repository

Cloudera uses a Docker Repository to deliver CDP Private Cloud Data Services. Learn more about how to set up custom Docker Repository for CDP Private Cloud Data Services.

○ Use an embedded Docker Repository
○ Use Cloudera's default Docker Repository
○ Use a custom Docker Repository

The following ports must be opened and allowed no matter which Docker repository option you choose.

- Ports required for Cloudera Manager/Cloudera Manager agent (port 5000 is required for Cloudera Machine Learning):

Protocol	Port
TCP	7180-7192
TCP	19001
TCP	5000
TCP	9000

- Inbound rules for ECS Server nodes (Kubernetes/RKE2):

Protocol	Port
TCP	9345
TCP	6443

Protocol	Port
UDP	8472
TCP	10250
TCP	2379
TCP	2380
TCP	30000-32767

- Inbound Rules for the ECS Agent (Kubernetes/RKE2):

Protocol	Port
UDP	4789

On the Configure Docker Repository page, select one of these options:

- Embedded Docker Repository

If you select the Internet Install Method option on the Getting Started page, images are copied over the internet from the Cloudera repository.

If you select the Air Gapped option, images are copied from a local http mirror you have set up in your environment.

Select Default to deploy all of the default Docker images to the repository, or select Select the Optional Images to choose which images to deploy. If you will be deploying Cloudera Machine Learning (CML), toggle the Cloudera Machine Learning switch on to copy the images for CML.

- Cloudera default Docker Repository

This option requires that cluster hosts have access to the internet and you have selected Internet as the install method.

- Custom Docker Repository

This option requires that you set up a Docker Repository in your environment and that all cluster hosts have connectivity to the repository.

> **Note:** If you are installing ECS on a single node, you should select the Use a Custom Docker Repository option. Single node ECS installation is supported, but is only intended to enable CDSW to CML migration.

You must enter the following options:

- Custom Docker Repository – Enter the URL for your Docker Repository
- Docker Username – Enter the username for the Docker Repository.
- Docker Password – Enter the password for the Docker Repository.

> **Important:** Do not use the $ character for this password.

- Docker Certificate – Click the Choose File button to upload a TLS certificate to secure communications with the Docker Repository.

Click the Generate the copy-docker script button to generate and download a script that copies the Docker images from Cloudera, or (for air-gapped installation) from a local http mirror in your network.

Run the script from a machine that is running Docker locally and has access to the Docker images using the following commands:

```
docker login [***URL for Docker Repository***] -u [***username of user
 with write access***]

bash copy-docker.txt
```

The copying operation may take 4 - 5 hours.

12. On the Configure Data Services page, you can modify configuration settings such as the data storage directory, number of replicas, and so on. If there are multiple disks mounted on each host with different characteristics (HDD and SSD), then Local Path Storage Directory must point to the path belonging to the optimal storage. Ensure that you have reviewed your changes. If you want to specify a custom certificate, place the certificate and the private key in a specific location on the Cloudera Manager server host and specify the paths in the input boxes labelled as Ingress Controller TLS/SSL Server Certificate/Private Key File below. This certificate will be copied to the Control Plane during the installation process.

Note:

The "Ingress Controller TLS/SSL Server Certificate File (PEM Format)" must only contain -----BEGIN C ERTIFICATE----- through -----END CERTIFICATE----- (inclusive) for the server and CA certs. It cannot include any preamble text and, and must not include a private key.

The "Ingress Controller TLS/SSL Server Private Key File (PEM Format)" must only contain the unencrypted key, and only the header through the footer, with no preamble text.

Both of these files must be readable by the "cloudera-scm" account.

For information on the required entries that must be present in DNS and TLS certificates when not using wildcards, refer to 'No Wildcard DNS/TLS Setup'

Click Continue.

13. On the Configure Databases page, click Continue.

14. On the Install Parcels page, the selected parcel is downloaded to the Cloudera Manager server host, distributed, unpacked, and activated on the ECS cluster hosts. Click Continue.

15. On the Inspect Cluster page, you can inspect your network performance and hosts. If the inspect tool displays any issues, you can fix those issues and run the inspect tool again.

Click Continue.

16. The installation progress is displayed on the Install Data Services page.

17. When the installation is complete, you will see the Summary image. You can now launch CDP Private Cloud.
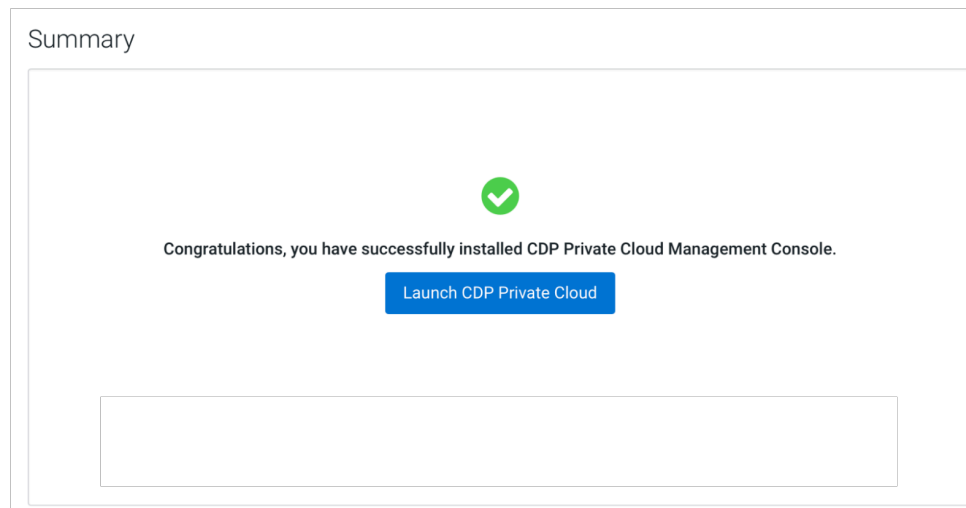
18. When the installation is complete, you can access your Private Cloud Data Services instance from Cloudera Manager. Click Data Services, then click Open Private Cloud Data Services for the applicable Data Services cluster.

If the installation fails, and you see the following error message in the stderr output during the Install Longhorn UI step, retry the installation by clicking the Resume button.

```
++ openssl passwd -stdin -apr1 + echo 'cm-longhorn:$apr1$gp2nrbtq$1KYPGI0QNl
FJ2lo5sV62l0' + kubectl -n longhorn-system create secret generic basic-auth
--from-file=auth + rm -f auth + kubectl -n longhorn-system apply -f /opt/clo
udera/cm-agent/service/ecs/longhorn-ingress.yaml Error from server (Internal
Error): error when creating "/opt/cloudera/cm-agent/service/ecs/longhorn-ing
ress.yaml":
Internal error occurred: failed calling webhook "validate.nginx.ingress.kub
ernetes.io": Post "https://rke2-ingress-nginx-controller-admission.kube-syst
em.svc:443/networking/v1/ingresses?timeout=10s": x509: certificate signed by
 unknown authority
```

What to do next

- If you specified a custom certificate, select the ECS cluster in Cloudera Manager, then select Actions > Update Ingress Controller. This command copies the cert.pem and key.pem files from the Cloudera Manager server host to the ECS Management Console host.
- Click Open Private Cloud Data Services to launch your CDP Private Cloud Data Services instance.
- Log in using the default username and password admin.
- On the Welcome to CDP Private Cloud page, click Change Password to change the Local Administrator Account password.
- Set up external authentication using the URL of the LDAP server and a CA certificate of your secure LDAP. Follow the instructions on the Welcome to CDP Private Cloud page to complete this step.

- Click Test Connection to ensure that you are able to connect to the configured LDAP server.
- Create your first Virtual Warehouse in the CDW Data Service
- Provision an ML Workspace in the CML Data Service
- Add a CDE service in the CDE Data Service

Related Information
No Wildcard DNS/TLS Setup

ECS Server High Availability

ECS Server High Availability (HA) is not enabled by default – you must enable it after installing ECS. If you do not wish to enable ECS HA, you can safely ignore this section. If you are enabling ECS HA, you should review the following notes and supported ECS Server scenarios before proceeding.

Note:

- Longhorn replication defaults to two replicas. This can be set only during the installation time. Three or more replicas potentially have performance issues.
- Kubectl delete node <host> permanently removes host from cluster and any data on the host is lost. You must reformat the host before reading to the cluster.
- Single node failure may cause the Control Plane or any other management service to be unavailable. In 1.3.4 or later, it will take several minutes to recover automatically.

ECS Server scenarios

Clusters with only two servers are not supported. This is only for the temporary transition from a single server cluster to a three server cluster.

1. Three or more servers

    - Redundancy requirements:

        - One failure requires three or more servers
        - Two failures require five or more servers
        - For more information see, Fault Tolerance
    - To recover, you must scale-up the ECS Server roles. For more information on adding ECS node to a cluster, see the following section.

2. Two servers to one server

    - Only after a double failure in a three server cluster

    - To recover:

        - Stop the ECS service
        - Remove both the failed ECS server roles and hosts from cluster
        - On the surviving server, run the following command `/opt/cloudera/parcels/ECS/bin/rke2 server --cluster-reset`
        - Start the ECS service

3. Single server

    - No failure supported

Enable ECS Server HA Post ECS Installation

If you want to enable ECS Server for High Availability after installing ECS, then you must proceed with this section. If you do not want to enable ECS HA, you can safely ignore this section.

As a prerequisite, during the installation, you must have installed ECS with 1 master (with app_domain as Load Balancer URL) + agents. When you are adding more masters, ensure that you add Docker server as well.

Install iptables on the new ECS master nodes

You must install iptables on all of the additional ECS master nodes.

If your ECS hosts are running the CentOS 8.4, OEL 8.4, or RHEL 8 operating systems, you must install iptables on all the ECS hosts. Run the following command on each additional ECS master node:

```
yum --setopt=tsflags=noscripts install -y iptables
```

Adding hosts to the containerized cluster

You must add hosts to the containerized cluster.

1. Log in to Cloudera Manager.
2. Navigate to the ECS service.
3. Click the Actions drop-down.
4. Click the Add Hosts button. The Add Hosts page appears.
5. Select the Add hosts to cluster option.
6. Select the cluster where you want to add the host from the drop-down list. Click Continue.
7. In the Specify Hosts page, provide a list of available hosts or you can add new hosts. You can provide the Fully Qualified Domain Name (FQDN) in the following patterns: You can specify multiple addresses and address ranges by separating them by commas, semicolons, tabs, or blank spaces, or by placing them on separate lines. Use this technique to make more specific searches instead of searching overly wide ranges.

   For example, use host[1-3].network.com to specify these hosts: host1.network.com, host2.network.com, host3.network.com.

   Click Continue.
8. In the Select Repository page, you must specify the repository location. Choose any one of the following:

   a. Cloudera Repository (Requires direct internet access on all hosts)
   b. Custom Repository
9. In the Select JDK page, select any one from the below options:

   a. Manually manage JDK
   b. Install a Cloudera-provided version of OpenJDK
   c. Install a system-provided version of OpenJDK
10. In the Enter Login Credentials page select the SSH Username and provide the password.
11. The Install Agents page appears. Click Continue.
12. In the Install Parcels page, the selected parcels are downloaded and installed on the host cluster. Click Continue.
13. In the Inspect Hosts page, you can inspect your hosts. If the inspect tool displays any issues, you can fix those issues and run the inspect tool again. Click Continue.
14. In the Select Host Template page, select the hosts.
15. The Deploy Client Config page appears. Click Finish.

Adding Role Instances to Docker Server

You must add role instances to the docker server.

1. Log in to Cloudera Manager.
2. Navigate to the ECS service.
3. Open Docker Server.
4. Click the Actions drop-down.
5. Click the Add Role Instances button.
6. Select the hosts.
7. Click OK.

Adding Role Instances to Containerised Cluster

You must add the role instances to the containerised cluster.

1. Log in to Cloudera Manager.
2. Navigate to the ECS service.
3. Click the Actions drop-down.

4. Click the Add Role Instances button. The Add Role Instances page appears.
5. In the Assign Roles page, specify the role assignments for your new roles. Click Continue.
6. In the Review Changes page, click Finish.

Starting Docker Server on Nodes

You must start the Docker server on nodes.

1. Log in to Cloudera Manager.
2. Navigate to the ECS service.
3. Open Docker Server.
4. Click the Actions for Selected drop-down.
5. Click Start. Docker Server starts.

Starting ECS Server on Nodes

You must start the ECS server on nodes.

1. Log in to Cloudera Manager.
2. Navigate to the ECS service.
3. Click the Instances tab.
4. Select the nodes by clicking the checkbox
5. Click the Actions for Selected drop-down.
6. Click Start. ECS Server starts.

Refreshing ECS

You must refresh the ECS servers.

1. Log in to Cloudera Manager.
2. Navigate to the ECS service.
3. Click the Actions drop-down.
4. Click the Refresh button.

Checking Nodes and Pods in the UI

You must check the nodes and pods in the UI.

1. Log in to Cloudera Manager.
2. Navigate to the ECS service.
3. Click the Web UI drop-down.
4. Click ECS Web UI. The Kubernetes web UI page opens in a new tab.
5. Check the Nodes and Pods on the Web UI.

Enable ECS Server HA and promote agents Post ECS Installation

If you want to enable ECS Server for High Availability after installing ECS, then you must proceed with this section. If you do not want to enable ECS HA, you can safely ignore this section.

As a prerequisite, during the installation, you must have installed ECS with 1 master (with app_domain as Load Balancer URL) + agents. This allows you to promote Agents as masters.

Enabling ECS Server deployment for High Availability

You can enable ECS Server deployment for High Availability by installing a Load Balancer and promoting the existing ECS Agents to ECS Server. By performing this procedure, you will be able to deploy HA on your existing ECS Server. You must have an ECS cluster installed and configured with a single ECS Server.

If you have a production quality ECS cluster, Cloudera recommends that you configure ECS Server High Availability. You can also consider having an ECS Server HA for any non-production ECS cluster that you expect to be available long-term.

Enabling ECS Server deployment for High Availability involves preparing your cluster, configuring a DNS wildcard entry, adding a Load Balancer into the topology, and promoting ECS Agents to the ECS Server. An ECS High Availability cluster must consist of:

- An odd number of server nodes that will run etcd, the Kubernetes API, and other control plane services. Cloudera recommends a minimum of three ECS Server nodes.
- Two or more agent nodes that are designated to run CDP data services.
- A software or hardware Load balancer using TCP mode (non-terminating https).

> **Note:** A Load Balancer is required for the ECS Server HA. This documentation uses HAProxy as an example. However, Cloudera recommends that you use your production quality Load Balancer technology from commercial vendors.

Architecture of CDP Private Cloud Data Services on a single ECS Server:
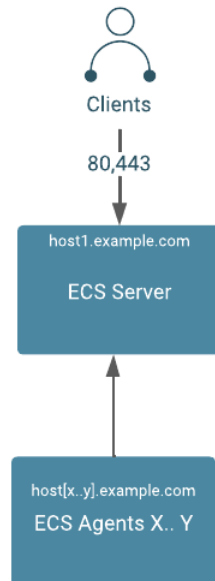
Architecture of CDP Private Cloud Data Services with High Availability:
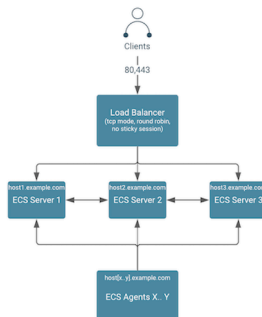
Preparing the cluster for High Availability:
Review the table to understand the requirements for enabling the High Availability.

1. This process has been tested with a minimum of five ECS hosts. However, Cloudera recommends six or more hosts.
2. DNS requirements for ECS High Availability must be fulfilled.

Hostname	Subdomain	Expected Roles	DNS ForwardZone	Reverse Zone PTR
"Wildcard" (hostname = *)	apps.ecs.example.com The string "apps" is required, "ecs" is up to user	Virtual app domain wildcard	"A Record" wildcard (hostname = *), may be a CNAME on certain DNS systems that use text-based config. Resolves to fixed IP of ha_proxy (or VIP of some commerical LB's)	N
"apps alias"	apps.ecs.example.com	Virtual app domain alias	"CNAME" alias points to A Record of ha_proxy (or VIP). Alternatively, this can be an ARecord with IP of ha_proxy (or VIP)	N/A
HAProxy (or commercial LB)	<domain of your LB>	HA Load Balancer	Depends on vendor/ software	
ecs-master1	example.com	ECS Server 1 Docker server	"A Record" resolves to IP of ecs-master1	Y
ecs-master2	example.com	ECS Server 2 Docker server	"A Record" resolves to IP of ecs-master2	Y
ecs-master3	example.com	ECS Server 3 Docker server	"A Record" resolves to IP of ecs-master3	Y
ecs-agentN	example.com	ECS Agent N Docker server N	"A Record" resolves to IP of ecs-agentN	Y

Note:

1. The above table uses a consistent subdomain ("example.com") but this is not mandatory. To support multiple domains, you must follow certain steps to ensure that the domains are forward and reverse resolvable using DNS, from all Base cluster and ECS cluster hosts (that is through forest/domain level trusts and/or hosts level /etc/resolv.conf config). You must avoid the use of /etc/hosts entries.
2. A predefined wildcard DNS record allows the resolution of *.apps.<app domain name> to the IP address of the Load Balancer. You cannot proceed further until this is in place.

High Level steps to enable an ECS High Availability cluster
Review the high level steps to understand the steps in enabling High Availability.

Enabling ECS High Availability Cluster

1 Verifying DNS Setup

2 Installing Load Balancer

3 Promoting ECS Agents to ECS Servers

4 Refreshing ECS Cluster

Note:

1. You must have installed an ECS with one ECS server and other nodes that are ECS Agents.
2. You must have a DNS wildcard record that has an IP address pointing to your Load Balancer (hostname or VIP). For more information, see the KB article.

Verifying DNS setup

You must verify the DNS setup to ensure that the app domain DNS hostname points to the Load Balancer.

Procedure

1. Verify that the app domain DNS hostname has moved from single non-HA ECS Server to the Load Balancer.

Hostname	Expected Roles	DNS
ecs-loadbalancer.example.com	Load Balancer	Resolves to IP of LB host (or VIP). The example uses 10.10.0.99. Both *.apps.ecs.example.com and apps.ecs.example.com resolve to 10.10.0.99.

2. Verify the DNS setup with nslookup.

Note: You must verify that a random hostname resolves in the wildcard entry. In this example, Cloudera uses foobar.apps.ecs.example.com as the random name. Both entries should resolve to the same IP address.

For example,

```
$ hosts="apps.ecs.example.com foobar.apps.ecs.example.com"
$ for target in $hosts; do nslookup $target; done

Server: 10.10.xx.xx
Address: 10.10.xx.xx#53

apps.ecs.example.com canonical name = ecs-loadbalancer.example.com.
Name: ecs-loadbalancer.example.com
```

```
Address: 10.10.0.99

Server: 10.10.xx.xx
Address: 10.10.xx.xx#53

Name: foobar.apps.ecs.example.com
Address: 10.10.0.99
```

Results

DNS setup is verified.

What to do next

You must now install the Load Balancer.

Installing Load Balancer

To install the HAProxy Load Balancer, Cloudera uses an example that uses a single instance of HAProxy, configured with round robin balancing and TCP mode. This allows for non-terminating https (https passthrough). The HAProxy service can be configured for High Availability using keepalived.

Before you begin

You must consult your operating system vendor's documentation for requirements and the install guide for configuring HAproxy with keepalived.

To install a HAProxy Load Balancer, you must ssh into the HAProxy host, install, and then configure HAProxy:

Procedure

1. `sudo su -`
2. `yum install haproxy -y`
3. `cp /etc/haproxy/haproxy.cfg /etc/haproxy/haproxy.cfg.bak`
4. `cat > /etc/haproxy/haproxy.cfg << EOF`

   global

log	127.0.0.1 local2
chroot	/var/lib/haproxy
pidfile	/var/run/haproxy.pid
user	haproxy
group	haproxy
daemon	

   defaults

mode	tcp
log	global
option	tcplog
option	dontlognull
option	redispatch
retries	3
maxconn	5000

timeout connect	5s
timeout client	50s
timeout server	50s

listen stats

bind *:8081
mode http
stats enable
stats refresh 30s
stats uri /stats
monitor-uri /healthz

frontend fe_k8s_80

bind *:80
default_backend be_k8s_80

backend be_k8s_80

balance roundrobin
mode tcp
server ecs-server1.example.com 10.10.0.1:80 check
server ecs-server2.example.com 10.10.0.2:80 check
server ecs-server3.example.com 10.10.0.3:80 check

frontend fe_k8s_443

bind *:443
default_backend be_k8s_443

backend be_k8s_443

balance roundrobin
mode tcp
server ecs-server1.example.com 10.10.0.1:443 check
server ecs-server2.example.com 10.10.0.2:443 check
server ecs-server3.example.com 10.10.0.3:443 check

EOF

| systemctl enable haproxy |
|---|
| systemctl restart haproxy |
| systemctl status haproxy |

5. You can verify that all the hosts are shown from the HAproxy UI. However, at this point the hosts are not listening to the configured ports.

> ⚠️ **Important:** Since you already have an ECS cluster running, you must alter your DNS wildcard to point to the IP address of the HAProxy server. You cannot change the Application Domain configured through the ECS wizard. So you must ensure that you send all ingress traffic to the HAProxy IP address by making that change in the IP address of your wildcard DNS Record.

> 📝 **Note:**
>
> a. Application Domain (app_domain property in Cloudera Manager) maps to your wildcard DNS record (For example, app_domain ecs.example.com maps to your DNS entry *.apps.ecs.example.com)
>
> b. The resolved IP address must be the host IP (or VIP) of your Load Balancer. For more information, see the Verify DNS Step 5 above.

Results

Load Balancer is now installed.

Promoting ECS Agents to ECS Servers

After installing the Load Balancer, you must reconfigure the existing Embedded Container Service (ECS) Agents to ECS Servers. This process is referred to as promoting the agents to servers. You must promote only one agent at a time.

About this task

In this example we will promote the ECS agent on agent1.example.com and then promote the ECS agent on agent2.example.com.

Procedure

1. In Cloudera Manager, select the ECS cluster, then click ECS. Stop the ECS agent running on agent1 and then delete the agent.

ECS-HACluster-01
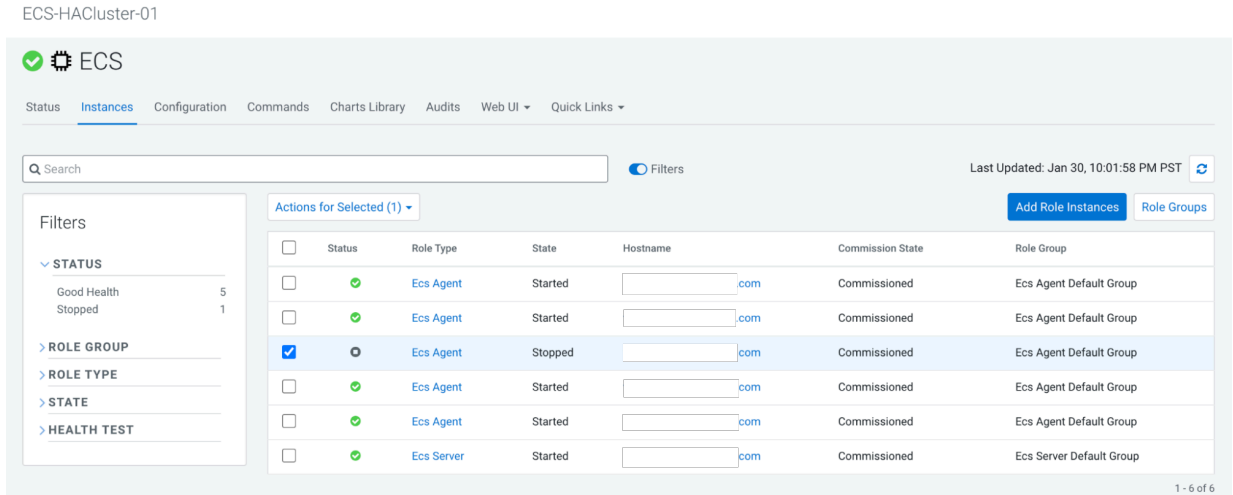
2. In ECS, click Add Role Instances.
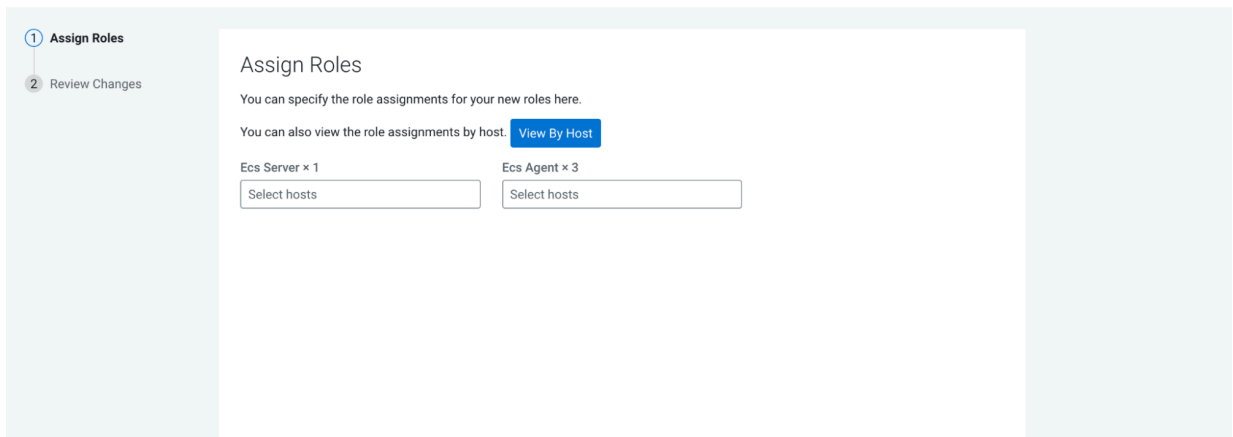
3. Add the available host agent1 as an ECS server in the Add Role Instances to ECS pop-up. Click OK.

4. Click Continue.

5. Start the new ECS server from the ECS Instances view. For example, start the ECS server on agent1.

6. Confirm the node's status from the Web UI or the command line by running the following command:

```
sudo /var/lib/rancher/rke2/bin/kubectl --kubeconfig=/etc/rancher/rke2/rke2.yaml get nodes
```

Note: Do not proceed until the node status is Ready. This may take several minutes.
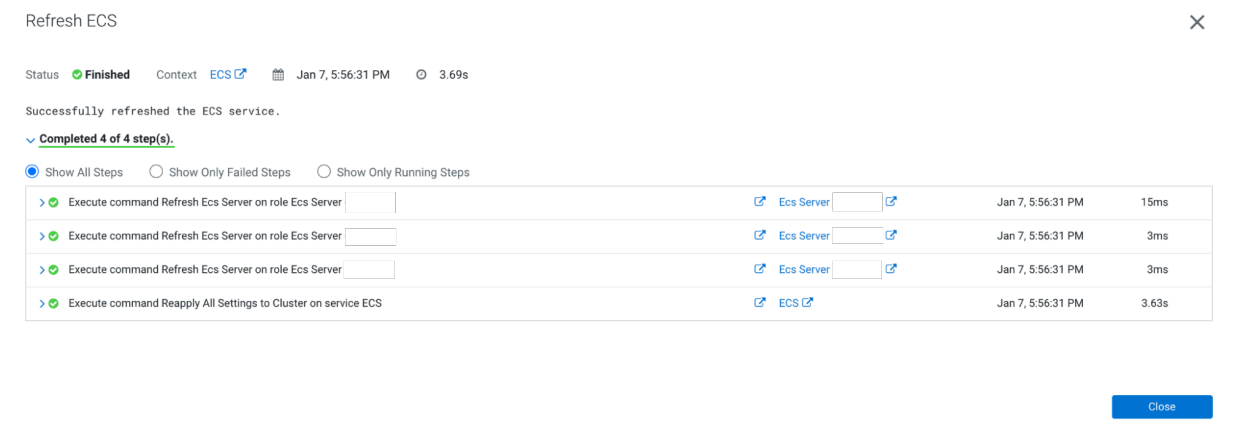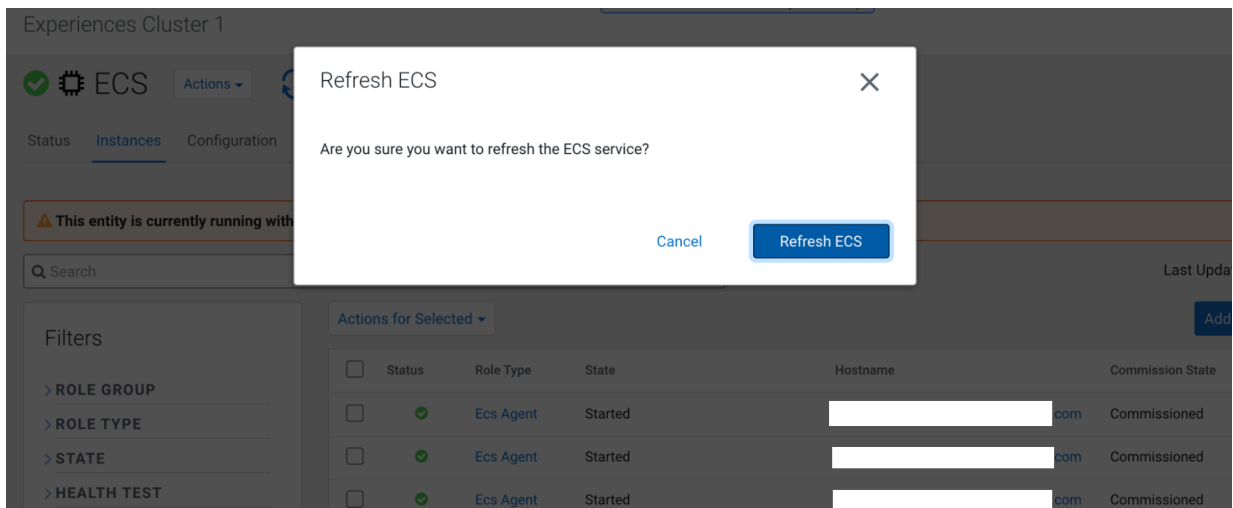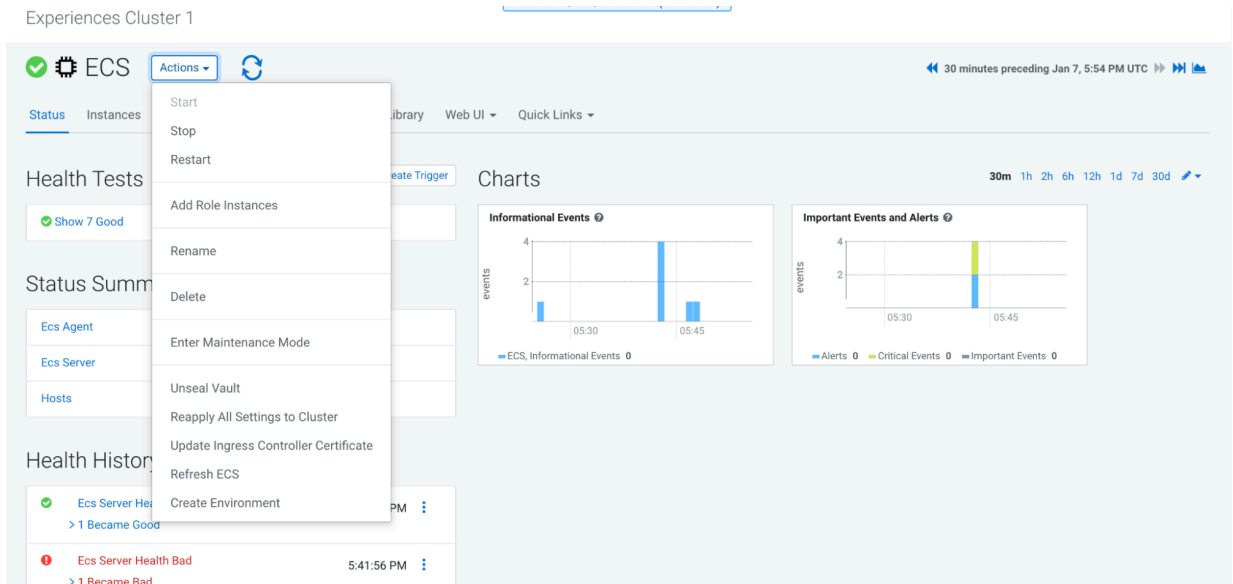

What to do next
When agent1 is ready, you can promote agent2. To promote agent2, perform steps 1-8 again using agent2.example.com.

Refreshing ECS
After all the ECS Agents are promoted to ECS Servers, you must log in to Cloudera Manager and refresh the ECS cluster.

Procedure

1. Navigate to ECS Cluster >> ECS view >> Actions >> Refresh ECS. This sets the ingress proxy so that all three servers are eligible to process incoming commands.

2. Confirm that all backends of HAProxy display the status UP. This may take several minutes.

3. Confirm that all pods are green in the ECS webUI >> (All Namespaces) >> Workloads.

4. Confirm that there are no alerts in the ECS service.

ECS1

Results

High Availability is now deployed on your ECS cluster.

Manually uninstalling ECS from a cluster

You can manually uninstall ECS from your cluster.

Before you begin

Before performing this procedure, ensure that you have activated the ECS parcel on the cluster hosts.

During the installation time of ECS, the directory for Longhorn and the LSO are decided by Cloudera Manager and defaults to /ecs.

Procedure

1. On each host in the cluster:
   a) /opt/cloudera/parcels/ECS/docker/docker container stop registry
   b) /opt/cloudera/parcels/ECS/docker/docker container rm -v registry
   c) /opt/cloudera/parcels/ECS/docker/docker image rm registry:2
2. Stop the ECS cluster in Cloudera Manager
3. On each host:
   a) cd /opt/cloudera/parcels/ECS/bin
   b) ./rke2-killall.sh # usually 2 times is sufficient
   c) Use umount to unmount all NFS disks.
   d) ./rke2-uninstall.sh
   e) rm -rf /ecs/* # assumes the default defaultDataPath and lsoDataPath
   f) rm -rf /var/lib/docker_server/* # deletes the auth and certs
   g) rm -rf /etc/docker/certs.d/* # delete the ca.crt
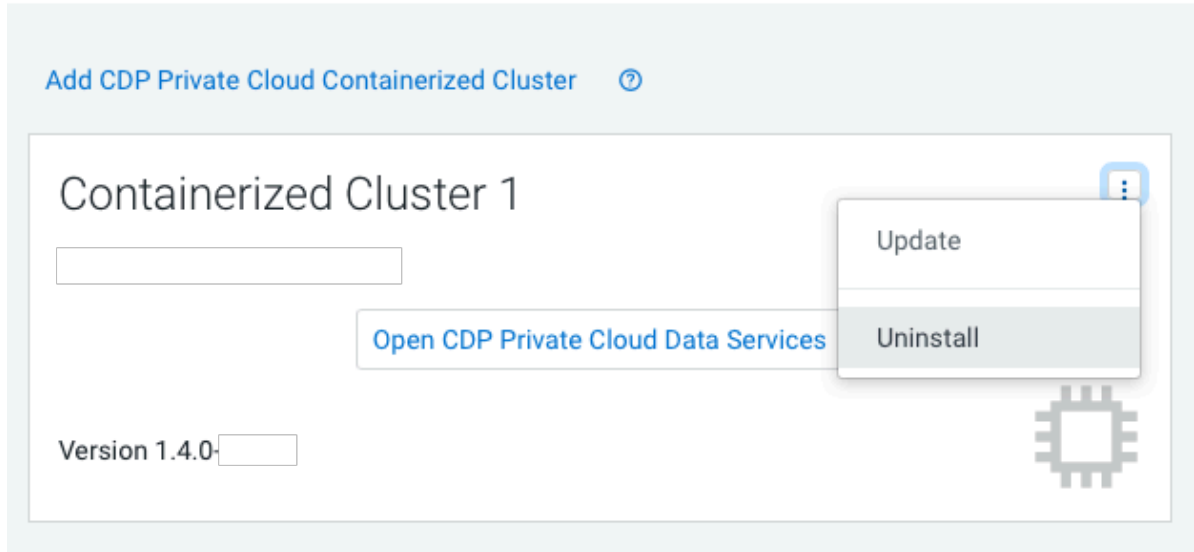   h) rm -rf /docker # assumes the default defaultDataPath for docker
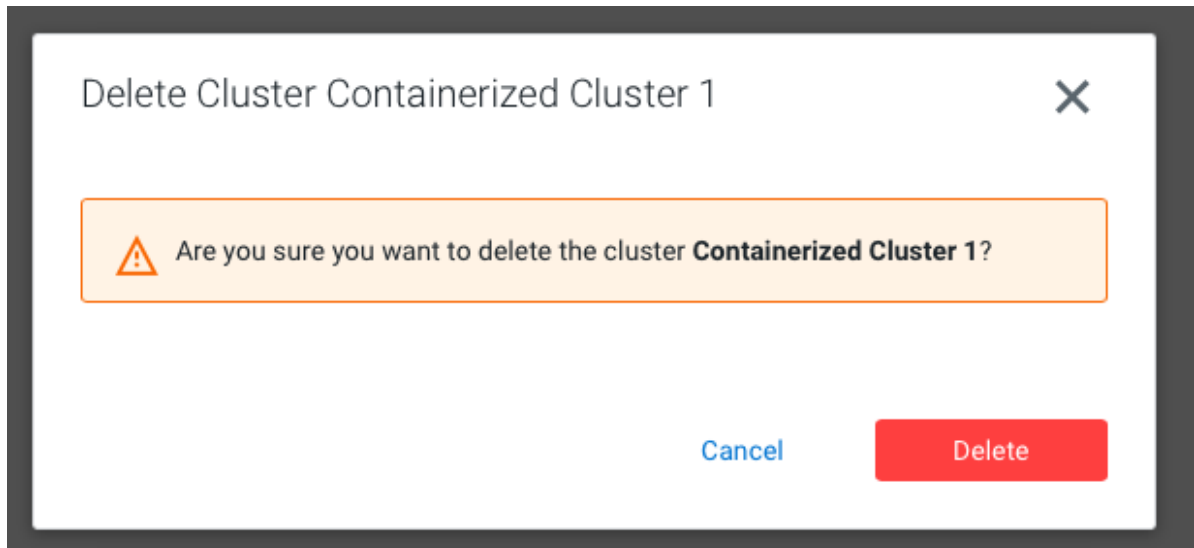
4. Delete the ECS cluster in Cloudera Manager.

a)

In Cloudera Manager, navigate to CDP Private Cloud Data Services and click ⋮ . Click Uninstall.

CDP Private Cloud Data Services

Add CDP Private Cloud Containerized Cluster ⑦

Containerized Cluster 1   ⋮

                                                          Update

Open CDP Private Cloud Data Services   Uninstall

Version 1.4.0-

b) The Delete Cluster wizard appears. Click Delete.

Delete Cluster Containerized Cluster 1   ✕

⚠ Are you sure you want to delete the cluster **Containerized Cluster 1**?

Cancel   Delete

5. Clean IPtables on each host:

```
echo "Reset iptables to ACCEPT all, then flush and delete all other chai
ns";
                        declare -A chains=( [filter]=INPUT:FORWARD:OUTPUT
 [raw]=PREROUTING:OUTPUT [mangle]=PREROUTING:INPUT:FORWARD:OUTPUT:POSTROUT
ING [security]=INPUT:FORWARD:OUTPUT [nat]=PREROUTING:INPUT:OUTPUT:POSTRO
UTING );
                        for table in "${!chains[@]}"; do
                        echo "${chains[$table]}" | tr : $"\n" | while IFS=
 read -r;
                        do sudo iptables -t "$table" -P "$REPLY" ACCEPT
                        done
```

```
sudo iptables -t "$table" -F
sudo iptables -t "$table" -X
done
```

> **Note:** Alternatively, an experimental script is available. This script combines steps three through five. The script is available here: https://github.com/cloudera-labs/snippets/blob/main/private-cloud/kill-2-rke .sh   script

6. Reboot the host(s).

7. Before you install ECS again, ensure that the IP tables list is empty by executing the following command: #iptable s -L