Cloudera Public Cloud

Azure Reference Network Architecture

Date published: 2019-08-22
Date modified: 2025-08-18

https://docs.cloudera.com/

Legal Notice

Contents

Cloudera reference network architecture on Azure

This topic includes a conceptual overview of the Cloudera on cloud network architecture for Azure, its use cases, and personas who should be using it.

Overview

Cloudera on cloud allows customers to set up cloud Data Lakes and compute workloads in their cloud accounts on AWS, Azure, and Google Cloud. It maps a cloud account to a concept called the environment into which all Cloudera workloads, including Cloudera Data Hub clusters (compute workload clusters)and data services (such as Cloudera Data Engineering, Cloudera Data Warehouse, Cloudera AI, Cloudera Operational Database, Cloudera DataFlow) are launched. For these Data Lakes, Cloudera Data Hub clusters, and data services to function correctly, several elements of the cloud architecture need to be configured appropriately: access permissions, networking setup, cloud storage and so on. Broadly, these elements can be configured in one of two ways:

- Cloudera can set up these elements for the customer

  Usually, this model helps to set up a working environment quickly and try out Cloudera. However, many enterprise customers prefer or even mandate specific configurations of a cloud environment for Infosec or compliance reasons. Setting up elements such as networking and cloud storage requires prior approvals and they would generally not prefer, or even actively prevent, a third party vendor like Cloudera to set up these elements automatically.
- Cloudera can work with pre-created elements provided by the customer

  In this model, the flow for creating the cloud Data Lakes accepts pre-created configurations of the cloud environment and launches workloads within those boundaries. This model is clearly more aligned with enterprise requirements. However, it brings with it the risk that the configuration might not necessarily play well with Cloudera requirements. As a result, customers might face issues launching Cloudera workloads and the turnaround time to get to a working environment might be much longer and involve many tedious interactions between Cloudera and the customer cloud teams.

From our experience in working with several enterprise customers, the most complicated element of the cloud environment setup is the cloud network configuration. The purpose of this document is to clearly articulate the networking requirements needed for setting up a functional Cloudera environment on cloud into which the Data Lakes and compute workloads of different types can be launched. It attempts to establish the different points of access to these workloads and establishes how the given architecture helps to accomplish this access.

Along with this document, you can use the Terraform Cloudera deployment to automatically set up a model of this reference architecture, which can then be reviewed for security and compliance purposes. For more information, see *Deploy Cloudera using Terraform*.

Note:

Currently this document only covers network architecture required for registering a Cloudera environment (with a Data Lake and FreeIPA) and deploying Cloudera Data Warehouse, and Cloudera AI in the environment. It does not currently cover Cloudera Data Hub clusters and the remaining data services (Cloudera Data Engineering, Cloudera DataFlow, and Cloudera Operational Database).

Use cases

Cloudera on cloud allows customers to process data in the cloud storage under a secure and governed Data Lake using different types of compute workloads that are provisioned via Cloudera Data Hub or data services. Typically the lifecycle of these workloads is as follows:

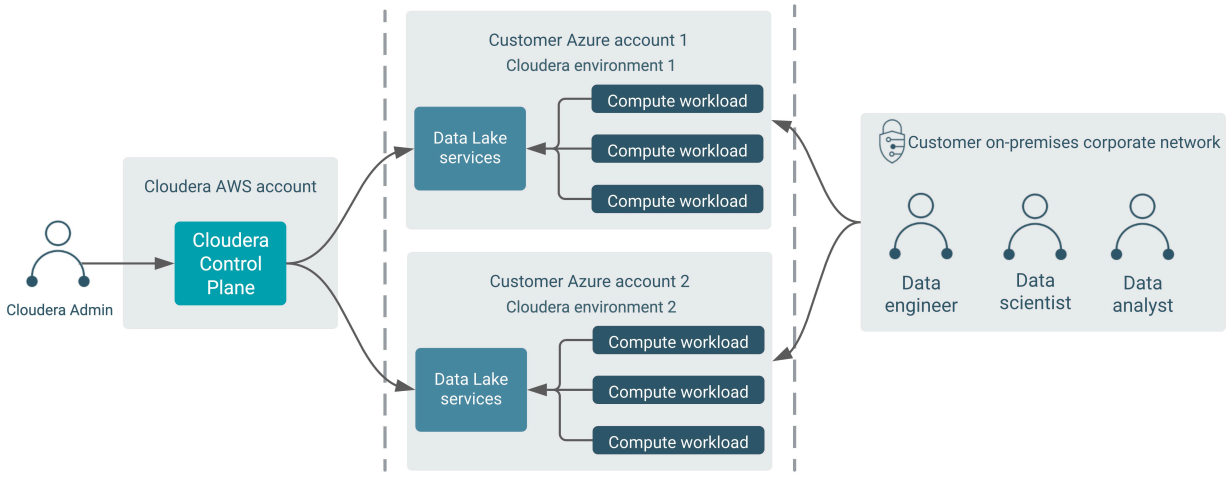
- A Cloudera environment is set up by a Cloudera admin using their cloud account. This sets up a cloud Data Lake cluster and FreeIPA cluster with security and governance services and an identity provider for this environment. The Cloudera admin may need to work with a cloud administrator to create all the cloud provider resources (including networking resources) that are required by Cloudera.

- Then one or more Cloudera Data Hub clusters and data services can be launched, linked to the Data Lake. Each of these Cloudera Data Hub clusters and data services typically serves a specific purpose such as data ingestion, analytics, machine learning and so on.
- These Cloudera Data Hub clusters and data services are accessed by data consumers such as data engineers, analysts or scientists. This is the core purpose of using Cloudera on the public cloud.
- These compute workload clusters and data services can be long-running or ephemeral, depending on the customer needs.

There are two types of Cloudera users who interact with the product for different purposes:

- Cloudera admins - These persons are usually concerned with the launch and maintenance of the cloud environment, and the Data Lake, FreeIPA, Cloudera Data Hub clusters, and data services running inside the environment. They use a Cloudera Management Console running in the Cloudera AWS account to perform these operations of managing the environment.
- Data consumers - These are the data scientists, data analysts, and data engineers who use the Cloudera Data Hub clusters and data services to process data. They mostly interact directly with the compute workloads (Cloudera Data Hub clusters and data services) running in their cloud account. They could access these either from their corporate networks (typically through a VPN) or other cloud networks their corporate owns.

These two types of users and their interaction with Cloudera are represented in the following diagram:



Related Information

Deploy Cloudera using Terraform

Taxonomy of network architectures

This topic provides a high-level overview of each type of network architecture that Cloudera supports.

At a high level, there are several types of network architectures that Cloudera supports. As can be expected, each type brings a unique trade-off among various aspects, such as ease of setup, security provided, workloads supported, and so on. This section only provides a high level overview of each type. The characteristics of each type are explained under appropriate sections in the rest of the document. The users must review the advantages and disadvantages of each of these taxonomies in detail before making a choice suitable to their needs.

Name	Description	Trade-offs
Publicly accessible networks	Deploys customer workloads to hosts with public IP addresses. Security groups must be used to restrict access only to corporate networks as needed.	Easy to set up for POCs. Low security levels.

Name	Description	Trade-offs
Semi-private networks	Deploys customer workloads to private subnets, but exposes services to which data consumers need access over a load balancer with a public IP address. Security groups or allow-lists (of IP addresses or ranges) on load balancers must be used to restrict access to these public services only to corporate networks as needed.	This option is fairly easy to set up too, but it may not solve all the use cases of access (in semi private networks). The surface of exposure is reduced, and it is reasonably secure.
Fully private networks	Deploys customer workloads to private subnets, and the services to which data consumers need access are only on private IPs. Requires connectivity to corporate networks to be provided using solutions like VPN gateways, and so on.	Complex to set up depending on prior experience of establishing such connectivity, primarily due to the way the customer has to solve the corporate network peering problem. But it is very secure.
Fully private outbound restricted networks	This is the same as fully private networks; Except, in addition, Cloudera provides a mechanism for users to configure an outbound proxy or firewall to monitor or restrict the communication outside their networks.	Most complex to set up, mainly considering the varied needs that data consumers would have to connect outside the VNet on an evolving basis. It is also the most secure for an enterprise.

Cloudera Management Console to customer cloud network

This topic explains the possible ways in which the Cloudera Control Plane can communicate with the compute infrastructure in the customer network, in the context of the Cloudera Management Console.

As described previously, the Cloudera admin would typically use the Cloudera Management Console that runs in the Cloudera Control Plane to launch Cloudera environments with Data Lakes, FreeIPA, Cloudera Data Hub clusters, and data services into their cloud accounts. In order to accomplish this, the Cloudera Control Plane and the compute infrastructure in the customer network (such as VMs, AKS clusters) should be able to communicate with each other. Depending on the chosen network architecture, this communication can occur in the ways described below.

Publicly accessible networks

In this model of publicly accessible networks, the compute infrastructure must be reachable over the public internet from the Cloudera Management Console. While this is fairly easy to set up, it is usually not preferred by enterprise customers, as it implies that the VM nodes or AKS nodes are assigned public IP addresses. While the access control rules for these nodes can still be restricted to the IP addresses of the Cloudera Management Console components, it is still considered insecure for each of the network architectures described earlier.

Semi-private networks

Publicly accessible networks are easy to set up for connectivity, both from the Cloudera Control Plane and the customer on-prem network, but have a large surface area of exposure as all compute infrastructure has public IP addresses. In contrast, fully private networks need special configuration to enable connectivity from the customer on-prem network, due to having no surface area of exposure to any of the compute infrastructure. While very secure, it is more complex to establish.

There is a third configuration supported by Cloudera, semi-private networks, that provides some trade-offs between these two options. In this configuration, the user deploys the worker nodes of the compute infrastructure on fully private networks as described above. However, the user chooses to expose UIs or APIs of the services fronting these worker nodes over a public network load balancer. By using this capability, the data consumers can access the UIs or APIs of the compute infrastructure through these load balancers. It is also possible to restrict the IP ranges from which such access is allowed using security groups.

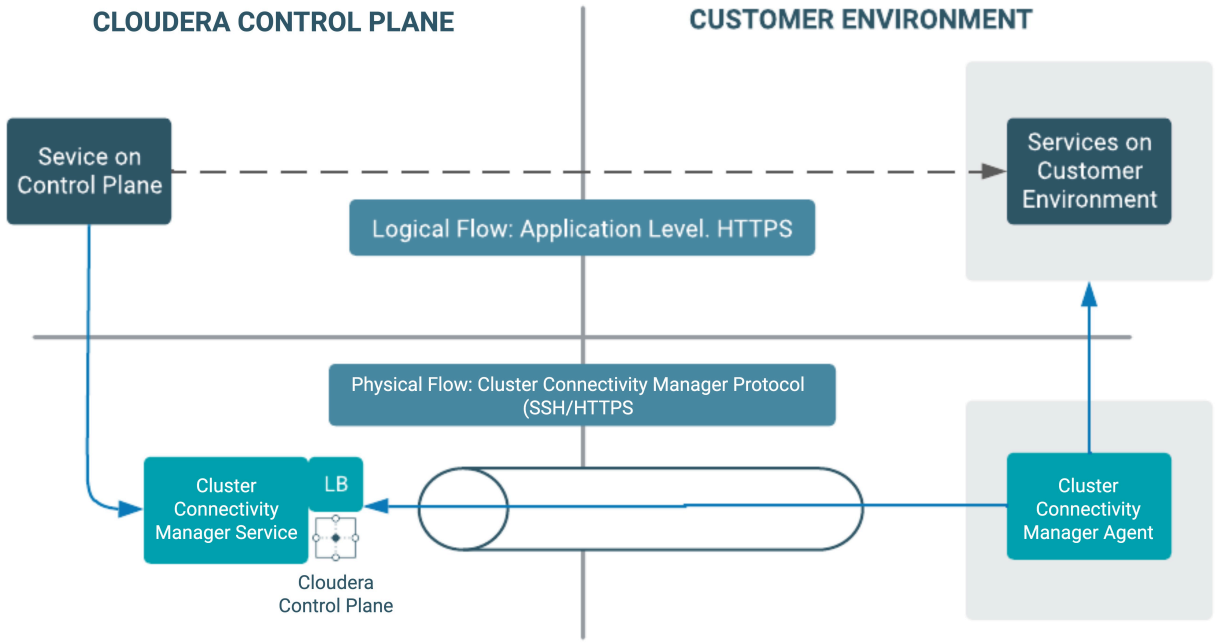
While this option provides a trade-off between ease of setup and exposure levels, it may not satisfy all use cases related to communication between various endpoints. For example, some compute workloads involving Kafka or NiFi would not benefit from having a simple publicly exposed load balancer. It is recommended that customers evaluate their use cases against the trade-off and choose an appropriately convenient and secure model of setup.

Fully private networks

In this model of fully private networks, the compute infrastructure is not assigned any public IP addresses. In this case, communication between the Cloudera Control Plane and compute infrastructure is established using a “tunnel” that originates from the customer network to the Cloudera Control Plane. All communication from the Cloudera Control Plane to the compute nodes is then passed through this tunnel. From experience, Cloudera has determined that this is the preferred model of communication for customers.

To elaborate on the tunneling approach, Cloudera uses a solution called “Cluster Connectivity Manager”. At a high level, the solution uses two components, an agent (Cluster Connectivity Manager agent) that runs on a VM provisioned in the customer network and a service (CCM service) that runs on the Cloudera Control Plane. The Cluster Connectivity Manager agent, at start-up time, establishes a connection with the Cluster Connectivity Manager service. This connection forms the tunnel. This tunnel is secured by asymmetric encryption. The private key is shared with the agent over cloud specific initialization mechanisms, such as a user-data script in Azure.

When any service on the Cloudera Control Plane wants to send a request to a service deployed on the customer environment (depicted in the below diagram as the “logical flow”), it physically sends a request to the Cluster Connectivity Manager service running in the Cloudera Control Plane. The Cluster Connectivity Manager agent and Cluster Connectivity Manager service collaborate over the established tunnel to accept the request, forward it to the appropriate service, and send a response over the tunnel to be handed over the calling service on the Cloudera Control Plane.



Currently, all AKS clusters provisioned by various Cloudera data services are enabled with public and private cluster endpoints. The AKS public endpoint is needed to facilitate the interactions between Cloudera Control Plane and the AKS cluster while worker nodes and Kubernetes control plane interact over private API endpoints. Cloudera Data Warehouse supports private AKS endpoints today (see “Enabling a private Cloudera Data Warehouse environment in Azure Kubernetes Service”). There are plans to support private AKS endpoints for other data services in the future. When this occurs, the documentation will be updated to reflect the same.

Fully private outbound restricted networks

Fully private outbound restricted networks is a variant of the fully private network where customers would like to pass outbound traffic originating from their cloud account through a proxy or firewall and explicitly allow-list URLs that are allowed to pass through. Cloudera on cloud supports such configuration. If such network architecture is chosen, the customer must ensure the following:

- Users configure a proxy for the environment via Cloudera, as documented in “Using a non-transparent proxy”.
- Compute resources (such as VMs used by Cloudera Data Hub clusters and data services) can connect to the proxy or firewall via appropriate routing rules.
- The proxy or firewall is set up to allow connections to all hosts, IP ranges, ports, and protocol types that are documented in “Azure outbound network access destinations”.

Note:

Given that fully private networks is the recommended option of connectivity in most cases, this document describes the architecture assuming a fully private network setup.

Related Information

Cluster Connectivity Manager

Enabling a private Cloudera Data Warehouse environment in Azure Kubernetes Service

Using a non-transparent proxy

Azure outbound network access destinations

Customer on-prem network to cloud network

After Cloudera Data Hub clusters and data services are launched in the customer’s cloud network, data consumers such as data engineers, data scientists, and data analysts access services running in these Cloudera data services. Sometimes, Cloudera admins who set up and operate these clusters might need this access to diagnose any issues the clusters face.

Examples of these include:

- Web UIs such as:
  - Hue: For running SQL queries in Hive tables
  - Cloudera AI Workbenches: For accessing Cloudera AI projects, models, notebooks, and so on
  - Cloudera Manager: For Cloudera Data Hub clusters and Data Lakes
  - Atlas and Ranger: For metadata, governance, and security in the Data Lake
- JDBC endpoints: Customers can connect tools such as Tableau using a JDBC URL pointing to the Hive server.
- SSH access: Data engineers might log in to nodes on the compute workload clusters and data services to run data processing jobs using YARN, Spark, or other data pipeline tools.
- Kube API access: Cloudera data services that run on AKS (such as Cloudera Data Warehouse and Cloudera AI) also provide admin access to Kubernetes for purposes of diagnosing issues.
- API access: Customers can use APIs for accessing many of the services exposed via the web UIs for purposes of automation and integration with other tools, applications, or other workloads they have. For example, Cloudera AI exposes the Cloudera AI API v2 to work with Cloudera AI projects and other entities. See Cloudera AI API v2.

These services are accessed by these consumers from within a corporate network inside a VPN. These services typically have endpoints that have a DNS name, the format of which is described more completely in the DNS section of this reference architecture documentation. These DNS names resolve to IP addresses assigned to the nodes, or load balancers fronting the ingest controllers of Kubernetes clusters. Note that these IP addresses are usually private IPs; Therefore, in order to be able to connect to these IPs from the on-premise network within a VPN, some special connectivity setup would be needed, typically accomplished using technologies like VPN peering, DirectConnect, transit gateways, and so on. While there are many options possible here, this document describes one concrete option of achieving this connectivity.

Related Information

Cloudera AI API v2

Network architecture

Cloudera recommends that customers configure their cloud networks as fully private networks, as described in this chapter. This will help on-boarding Data Lakes, Cloudera Data Hub cluster, and data services smoothly.

Note:

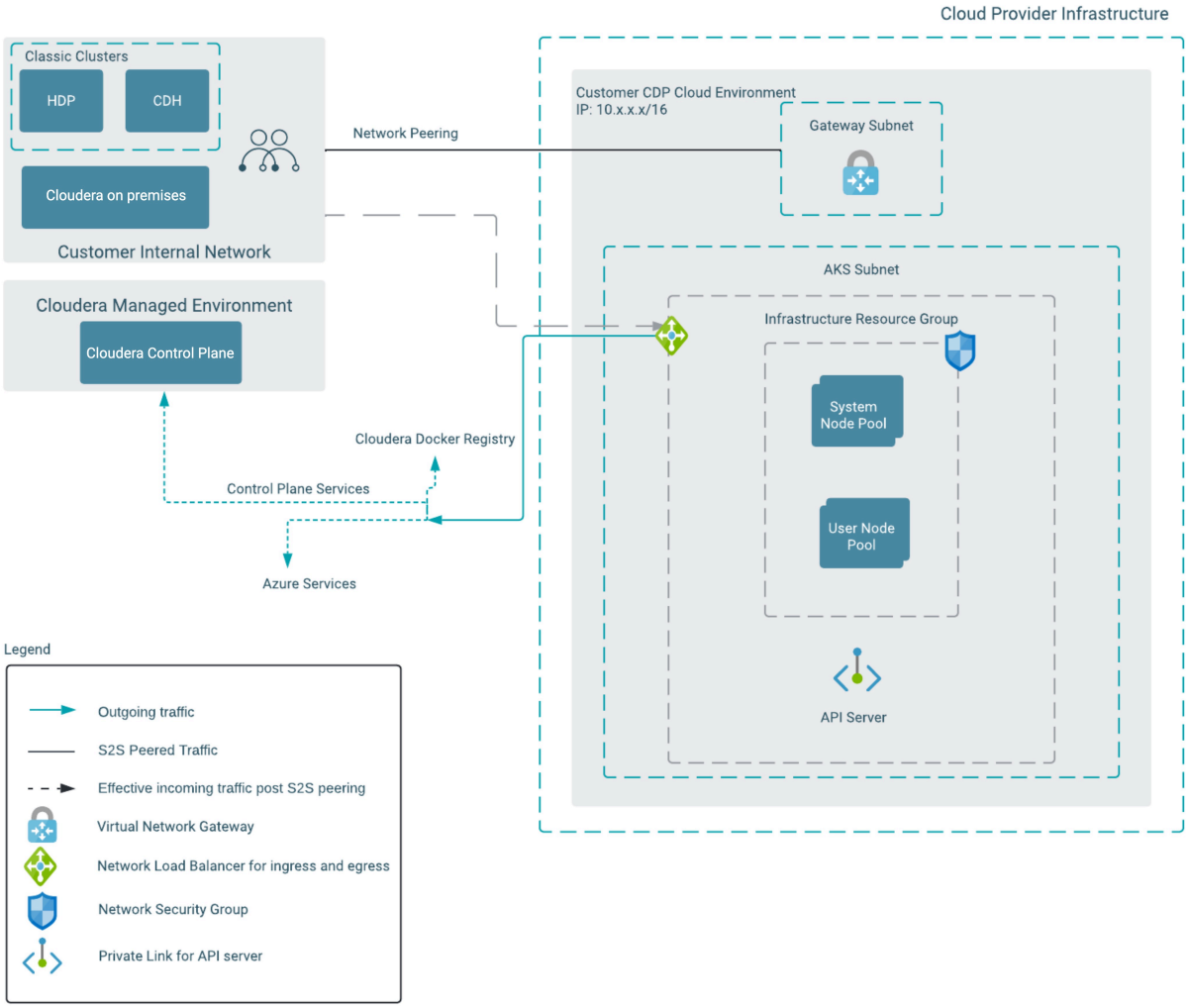
This network architecture only covers the fully private networks and assumes unrestricted outbound access.

The cloudera-deploy tool, which is released along with this document can be used to automatically set up a model of this reference architecture, which can then be reviewed for security and compliance purposes.

Architecture diagrams

This topic includes diagrams illustrating the various elements of the network architecture in the customer's cloud account into which Cloudera environments with Data Lakes, Cloudera Data Hub clusters, and data services will be launched.

Cloudera recommends that customers configure their cloud networks as described in this chapter and illustrated in the following diagrams. This will help onboarding Data Lakes, Cloudera Data Hub cluster, and data services smoothly. The following diagram illustrates the configuration for a fully private network that can be configured by the customer. This configuration can be provided by the Cloudera admins when they are setting up Cloudera environments and workloads which will get launched into this configuration.

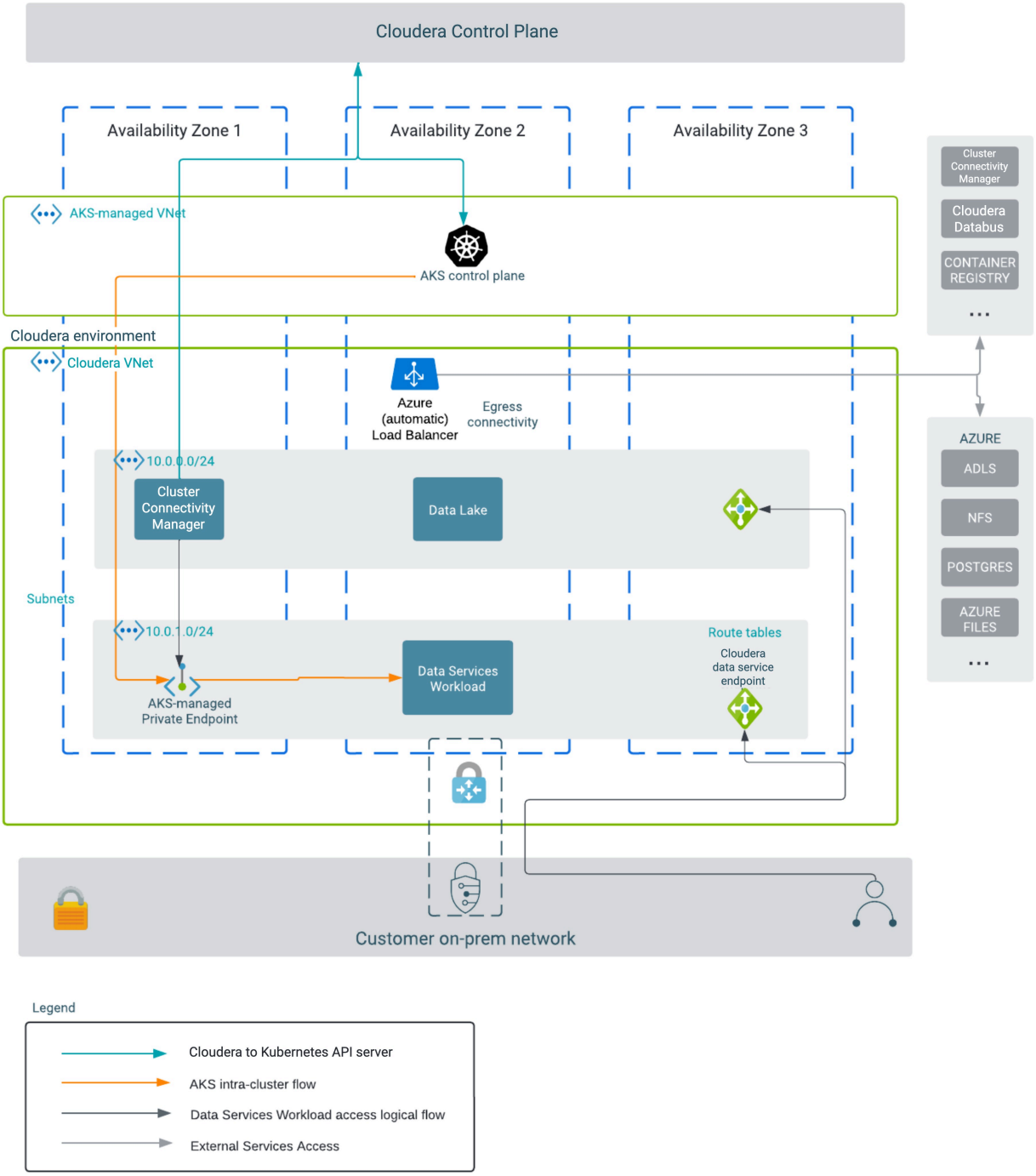


Note the following points about this architecture:

- The configuration is a fully private network configuration - that is, the workloads are launched on nodes that do not have public IP addresses.
- Workloads connect to the Cloudera Control Plane over a fixed IP and port range.
- For users to be able to connect from the customer on-prem network to the Cloudera workloads in the private subnet, some network connectivity setup is required. In this case, a customer's network peered to Azure VNet via Azure VPN gateway is shown.

Some of the Cloudera data services are based on Azure AKS clusters. Azure's AKS manages the Kubernetes control plane nodes while the worker nodes that make up the Cloudera workload cluster get provisioned in the customer's VPC. The AKS control plane has an API endpoint for administrative purposes which is commonly referred to as API server address. The data service itself is accessible through a service endpoint ELB.

This is illustrated in the following diagram:



As can be seen in the above diagram, Cloudera workloads have dependencies on some Azure cloud services such as ADLS, Azure Postgres and so on. A full list of these services, described in the context of each workload is specified in “Azure outbound network access destinations”.

In this example of a fully private deployment with an existing network of your own configuration, you must configure your own egress connectivity through a NAT gateway setup or user-defined routing. For more information see VNet and subnets and Azure Load Balancers in Data Lakes and Cloudera Data Hub clusters.

In the chapters that follow, we detail the elements of this architecture, including specifying the configuration and options in each of the components.

Related Information

Azure outbound network access destinations

Component description

This section includes an overview of the VNet, subnets, gateways and route tables, and security groups required for Cloudera on Azure.

VNet

An Azure Virtual Network (VNet) is needed for deploying Cloudera workloads into the customer's cloud account. VNet is similar to a traditional network that you would operate in your own data center, but brings with it additional benefits of Azure's infrastructure such as scale, availability, and isolation.

Cloudera recommends that the VNet used for Cloudera is configured with the properties specified below:

- The CIDR block for the VNet should be sufficiently large for supporting all the Cloudera Data Hub clusters and data services that you intend to run. Refer to "VNet and subnet planning" to understand how to compute the CIDR block range.
- In addition, you may want to ensure that the CIDR ranges assigned to the VNet do not overlap with any of your on-premise network CIDR ranges, as this may be a requirement for setting up connectivity from your on-premise network to the subnets.

Related Information

VNet and subnet planning

Subnets

A subnet is a partition of the virtual network in which Cloudera workloads are launched.

It is recommended that the subnets be configured with the following properties:

- The CIDR block for the subnet should be sufficiently large for supporting the Cloudera workload targeted to run inside it. Refer to "VNet and subnet planning" to understand how to compute the CIDR block range.
- Several Cloudera data services run on Kubernetes and use Kubenet CNI plugin for networking. Azure's Kubenet CNI plugin requires that the subnet is not shared between multiple Kubernetes clusters as it adds custom routes (see "Bring your own subnet and route table with kubenet"). Therefore, as many subnets as the expected number of workloads need to be created.
- In addition, you may want to ensure that the CIDR ranges assigned to the subnets do not overlap with any of your on-premise network CIDR ranges, as this may be a requirement for setting up connectivity from your on-prem network to the subnets.
- A subnet can be associated with a Network Security Group (NSG). However, since Cloudera works with a fully private network configuration where communication is always initiated from VMs within the subnets, an NSG at subnet level is generally not useful for this configuration.

Related Information

VNet and subnet planning

Bring your own subnet and route table with kubenet

Gateways and route tables

This topic covers recommended gateway and route table configurations for Cloudera on Azure.

Connectivity from Cloudera Control Plane to Cloudera workloads

- As described in the "Subnets" section above, each Cloudera data service workload requires its own subnet and a non-shared route table associated with it (see "Bring your own subnet and route table with kubenet").

- As described in the “Taxonomy of network architectures”, nodes in the Cloudera workloads need to connect to the Cloudera Control Plane over the internet to establish a “tunnel” over which the Cloudera Control Plane can send instructions to the workloads.
- Private AKS cluster is a feature that lets Cloudera access a Kubernetes workload cluster over a private IP address (see “Enabling a private Cloudera Data Warehouse environment in Azure Kubernetes Service”). When it is enabled for a Cloudera Data Warehouse workload, Cloudera Data Warehouse requires the user to have already set up internet connectivity for the subnet (see “Outbound type of userDefinedRouting)”.
- Cloudera data services such as Datalake and Cloudera AI create a public load balancer for internet connectivity.
- If a firewall is configured, the destinations described in “Azure outbound network access destinations” need to be allowed for the Cloudera workloads to work.

Connectivity from customer on-prem to Cloudera workloads

- As described in the “Use cases” section (see “Cloudera on cloud reference network architecture for Azure”>“Use Cases”), data consumers need to access data processing or consumption services in the Cloudera workloads.
- Given that these are created with private IP addresses in private subnets, the customers need to arrange for access to these addresses from their on-prem or corporate networks in specific ways.
- There are several possible solutions for achieving this, but one that is depicted in the architectural diagram, uses “Azure VPN Gateway”.
- Each workload provides an option to connect to the workload endpoint over a public or a private IP. It's recommended that the public IPs are disabled and that users rely on the corporate network connectivity for accessing the workloads with private IPs.

Related Information

Subnets

Bring your own subnet and route table with kubenet

Taxonomy of network architectures

Enabling a private Cloudera Data Warehouse environment in Azure Kubernetes Service

Outbound type of userDefinedRouting

Azure outbound network access destinations

Cloudera reference network architecture on Azure

Azure VPN Gateway

Private endpoints

Cloudera workloads can be configured to access the Azure resources over a private IP (which is called a private endpoint) or over a public IP (which is called a service endpoint).

The private endpoint setup requires a private DNS zone which the VNet is linked to and resolves the Azure resources to private IP addresses. Cloudera supports a private endpoint configuration for Azure postgres only. Cloudera admin can choose to either create the private DNS zone and link it to the VNet as described in “Bringing your own private DNS”, or let the Cloudera create them when provided with necessary configuration described in “Using Cloudera-managed private DNS”.

Cloudera supports only service endpoint configuration for other Azure resources(such as Microsoft Storage). The subnets need to be enabled to support the service endpoints. See “Service endpoint for Azure Postgres” for detailed steps.

Related Information

Bringing your own private DNS

Using Cloudera-managed private DNS

Service endpoint for Azure Postgres

Security groups

During the specification of a VNet to Cloudera, the Cloudera admin specifies security groups that will be associated with all the Cloudera workloads launched within that VNet. These security groups will be used in allowing the incoming traffic to the hosts.

Security groups for Data Lakes

During the specification of a VNet to Cloudera, the Cloudera admin can either let Cloudera create the required security groups, taking a list of IP address CIDRs as input; or create them in Azure and then provide them to Cloudera.

When getting started with Cloudera, the Cloudera admin can let Cloudera create security groups, taking a list of IP address CIDRs as input. These will be used in allowing the incoming traffic to the hosts. The list of CIDR ranges should correspond to the address ranges from which the Cloudera workloads will be accessed. In a VPN-peered VNet, this would also include address ranges from customer's on-prem network. This model is useful for initial testing given the ease of set up.

Alternatively, the Cloudera admin can create security groups on their own and select them during the setup of the VNet and other network configuration. This model is better for production workloads, as it allows for greater control in the hands of the Cloudera admin. However, note that the Cloudera admin must ensure that the rules meet the requirements described below.

For a fully private network, network security groups should be configured according to the types of access requirements needed by the different services in the workloads. The "Network security groups" section includes all the details of the necessary rules that need to be configured.

Note that for a fully private network, even specifying an open access here (such as 0.0.0.0/0) is restrictive because these services are deployed in a private subnet without a public IP address and hence do not have an incoming route from the Internet. However, the list of CIDR ranges may be useful to restrict which private subnets of the customer's on-prem network can access the services.

Rules for AKS based workloads are described separately in the following section.

Related Information

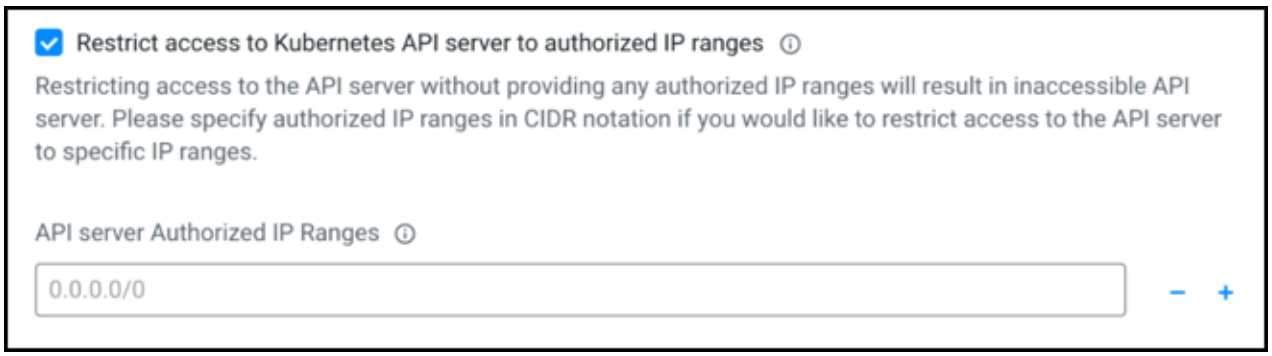
Network security groups

Additional rules for AKS-based workloads

At the time of enabling a Cloudera data service, the Cloudera admin can specify a list of CIDR ranges that will be used in allowing the incoming traffic to the workload Load Balancer.

This list of CIDR ranges should correspond to the address ranges from which the Cloudera data service workloads will be accessed. In a VPN peered VNet, this would also include address ranges from customer's on-prem network. In a fully private network setup, 0.0.0.0/0 implies access only within the VNet and the peered VPN network which is still restrictive.

Cloudera Data Warehouse data service currently supports fully private AKS clusters, which means the Kubernetes API server will have a private IP address. Cloudera AI uses a public endpoint by default for all AKS cluster control planes. It is highly recommended to provide a list of outbound public CIDR ranges at the time of provisioning a data service to restrict access to the AKS clusters. In addition, the data service adds Cloudera public CIDR range to allow access between workloads and Cloudera Control Plane. An example configuration section for a data service looks like below:



Specific guidelines for restricting access to Kubernetes API server and workloads are detailed in “Restricting access for Cloudera services that create their own security groups on Azure” by each data service.

Within the AKS cluster, security groups are defined to facilitate AKS control plane-pod communication, inter-pod and inter-worker node communication as well as workload communication through Load Balancers.

Related Information

Restricting access for Cloudera services that create their own security groups on Azure

Outbound connectivity requirements

Outbound traffic from the worker nodes is unrestricted and is targeted at other Azure services and Cloudera services. The comprehensive list of services that get accessed from a Cloudera environment can be found in “Azure outbound network access destinations”.

Related Information

Azure outbound network access destinations

Domain names for the endpoints

The previous sections dealt with how connectivity is established to the workload infrastructure. This section deals with “addressability”.

The workloads launched by Cloudera contain a few services that need to be accessed by the Cloudera admins and data consumers. These include services such as Cloudera Manager; metadata services such as the Hive Metastore, Atlas, or Ranger; and data processing or consumption services such as Oozie server, Hue, and so on. Given the nature of the cloud infrastructure, the IP addresses for the nodes running these services may change (for example if the infrastructure is restarted or repaired). However, these should have statically addressable DNS names so that users can access them with the same names.

In order to help with this, Cloudera assigns DNS names to these nodes. These naming schemes have the following properties:

- The DNS name is of the following format for each Data Lake node and the Data Lake cluster endpoint:

  ```
  <CLUSTER_NAME>-{<HOST_GROUP><i>}.<ENVIRONMENT-IDENTIFIER>.<CUSTOMER_IDEN
  TIFIER>.cloudera.site
  ```

  An example could be:

  ```
  my-dataeng-master0.my-envir.aaaa-1234.cloudera.site
  ```

  This name has the following components:

  - The base domain is cloudera.site. This is a publicly registered “DNS suffix”. It is also a registered Route53 hosted zone in a Cloudera owned AWS account.
  - The <CUSTOMER_IDENTIFIER> is unique to a customer account on Cloudera made of alphanumeric characters and a “-” (dash).
  - The <ENVIRONMENT_IDENTIFIER> is generated based on the environment name and is truncated to 8 characters.
  - The <CLUSTER_NAME> is the cluster name given to the Data Lake or Cloudera Data Hub. It is appended with a <HOST_GROUP> name such as “gateway”, “master”, “worker”, and so on, depending on the role that the node plays in the cluster. If there are more than one of these nodes playing the same role, they are appended with a serial number <i>.
- The DNS name of the endpoints of the Cloudera data services is of the following format:
  - For a Virtual Warehouse in Cloudera Data Warehouse, it is:

    ```
    <VIRTUAL_WAREHOUSE_NAME>.dw-<CDW_ENVIRONMENT_IDENTIFIER>.<CUSTOMER_IDENT
    IFIER>.cloudera.site
    ```

    - <VIRTUAL_WAREHOUSE_NAME> is the name of the Virtual Warehouse created. There could be multiple virtual warehouses for a given Cloudera environment.
    - <CDW_ENVIRONMENT_IDENTIFIER> is the identifier for the Cloudera environment.
  - For a Session Terminal in Cloudera AI workbench, it is:

    ```
    <TTY_SESSION_ID>.<CML_WORKSPACE_ID>.<ENVIRONMENT_IDENTIFIER>.<CUSTOMER_I
    DENTIFIER>.cloudera.site
    ```

    - <TTY_SESSION_ID> is the ID of the Cloudera AI Terminal Session ID.
    - <CML_WORKSPACE_ID> is the ID of the Cloudera AI workspace created.
    - The <ENVIRONMENT_IDENTIFIER> is generated based on the environment name and is truncated to 8 characters. If the 8th character is a “-” (dash), then it is truncated to 7 characters instead.
  - For all the Cloudera data services listed above, the common portions of the DNS include:
    - The base domain is cloudera.site. This is a publicly registered “DNS Suffix”. It is also a registered Route53 hosted zone in a Cloudera owned AWS account.
    - The <CUSTOMER_IDENTIFIER> is unique to a customer account on Cloudera made of alphanumeric characters and a “-” (dash).
  - The length of the DNS name is restricted to 64 characters due to some limitations with Hue workloads.
  - These names are stored as A records in the Route53 hosted zone in the Cloudera managed Cloudera Control Plane AWS account. Hence, you can resolve these names from any location outside of the VPC. However, note that they would still resolve to private IP addresses and hence are constrained by the connectivity setup described in preceding sections.
  - Within a Cloudera environment, the DNS resolution happens differently. Every Cloudera environment has a DNS server that is played by a component called FreeIPA. This server is seeded with the hostnames of the nodes of all workload clusters in the environment. Every node in a Data Lake, Cloudera Data Hub and data service is configured to look up the FreeIPA DNS service for name resolution within the cluster.
  - FreeIPA nodes are internal to the Cloudera environment both for resolving and use. They don't have public DNS records.

Related Information

DNS Suffix

DNS

Azure-provided DNS is recommended for the VNet. This is required to enable private AKS clusters and private access to Postgres databases among others, as described in “Private endpoints” above.

Cloudera Private Links Network for Azure

Cloudera Private Links Network enables you to connect privately and securely to the Cloudera Control Plane without traversing the internet. You can use Cloudera Private Links Network for end-to-end encryption of your workloads between Cloudera Control Plane and Azure private endpoints.

This documentation provides the following details and steps about Cloudera Private Links Network:

- High-level options of Virtual Network (VNet) endpoint placement
- Cloudera Private Links Network deployment process
- Instructions of how to set up both Private Link options:
  - VNet: Setup of Cloudera Private Links Network for a workload VNet through CDP CLI
  - Authorization: Authorization with CDP CLI to enable the setup of Cloudera Private Links Network through your automation tools
- References for proxy profile configuration and considerations, and Cloudera Private Links Network commands

Comparison of connectivity setup without and with Cloudera Private Links Network

Without Cloudera Private Links Network, your workload environment communicates with the Cloudera Control Plane through the internet. This traffic may optionally flow through a managed egress proxy. The following two diagrams illustrate this:

Figure 1: Connectivity from workload environment to Cloudera Control Plane through the internet

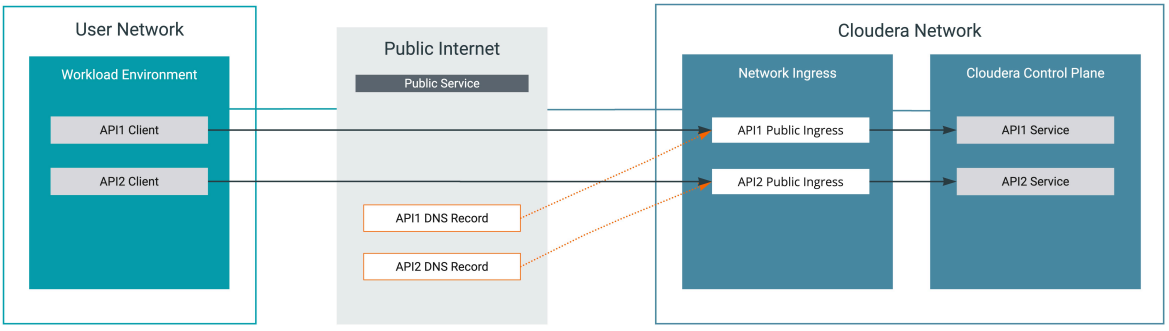
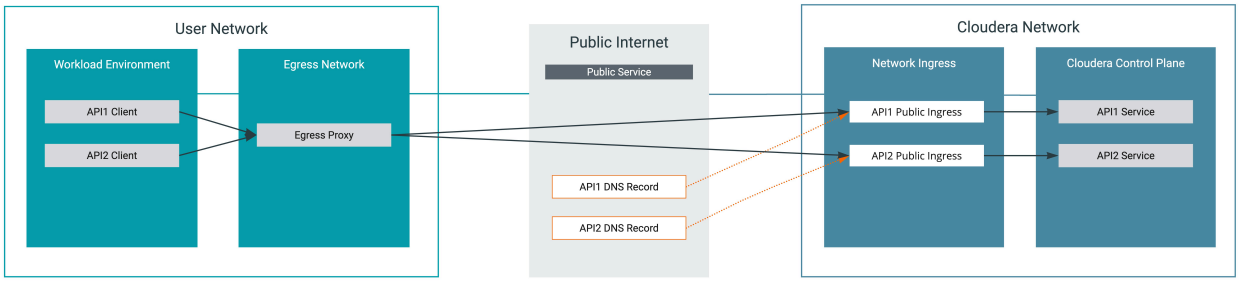


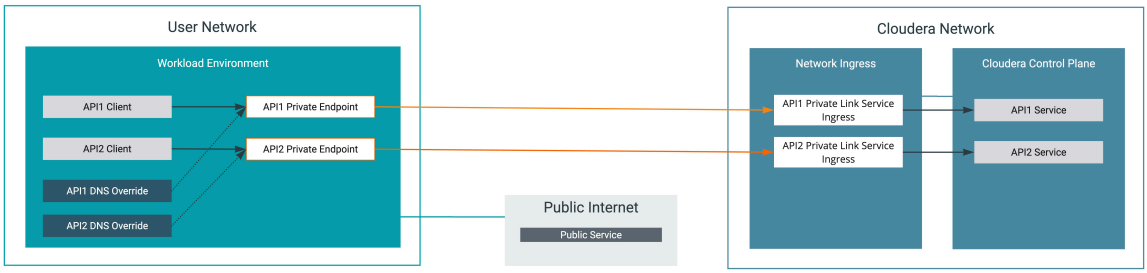
Figure 2: Connectivity from workload environment to Cloudera Control Plane through the internet and egress proxy



With Cloudera Private Links Network, the Cloudera Control Plane is accessed as if the Cloudera Control Plane would be on your network. This means that IP addresses are assigned to the Cloudera Control Plane services from your network, and DNS lookups will return your local IP addresses.

To ensure private connectivity through network ingress between the workload environment and Cloudera Control Plane, private endpoints can be added. The following illustration details the scenario where the private endpoints are in the same VNet as your workload environment. In this case, the private endpoints receive IPs from the workload environment VNet subnets:

Figure 3: Private endpoints in workload environment VNet



The following options are available for DNS overrides:

- DNS is a regional or global view: Installing overrides at a regional or global scope will impact DNS resolution for other VNets, these other VNets will attempt to use the local private endpoints of the VNet.

This section does not include an exhaustive list of design options, but should cover most cases. For more information about more advanced use cases, see the Additional VNet scenarios section.

Supported regions and hostnames

Review the list of supported regions, and the list of hostnames that are covered by the Private Links.

The following table describes the available service components for each region:

Cloudera Control Plane Region	Private network service region in Azure
us-west-1 (USA)	westus1
ap-1 (Australia)	australiaeast

Note: In case you need to use a region that is not mentioned in the supported region table, reach out to Cloudera Support.

Supported hostnames for Private Links

The following list details the hostnames where the traffic goes through the private endpoints when Private Links are created for the Azure VMs:

- '*.altus.cloudera.com'
- '*.ap-1.cdp.cloudera.com'
- '*.us-west-1.ccm.cdp.cloudera.com'
- '*.us-west-1.cdp.cloudera.com'

Setting up Cloudera Private Links Network for Azure environments

Learn about how to set up Cloudera Private Links Network using CDP CLI for both the VNet and Authorization options.

You can configure private links from the workload environment to Cloudera Control Plane using Cloudera Private Links Network. You have the following options to choose from:

- VNet option: Using CDP CLI to set up private links in a Cloudera environment network. Cloudera automation assists you with authorizing and end-to-end setup.
- Authorization option: Using your own network automation to create private links in a Cloudera environment. CDP CLI is used to authorize your cloud provider account to connect to Cloudera.

Prerequisites

Before you begin, review the following prerequisites and requirements.

The following requirements must be met before creating Cloudera Private Links Network:

- Ensure that you have completed the Azure credential requirements described in Prerequisites for the provisioning credential to be able to create Private Links.
- The Private Links should be set up before creating an environment.
- Configuring Private Links in Cloudera can only be performed through CDP CLI. You need to have CDP CLI installed and configured.
- In order to configure Private Links in Cloudera through CDP CLI, you must have the EnvironmentCreator or PowerUser role in Cloudera.

Creating Cloudera Private Links Network with VNet option

Learn about how to create Cloudera Private Links Network using the VNet option.

Required Role: EnvironmentCreator or PowerUser

You need to use the following CDP CLI command to create the Cloudera Private Links Network with VNet option:

```
cdp cloudprivatelinks create-private-link-endpoint
```

This command is used to create a private endpoint in your workload VNet. This establishes the private link connectivity between the created private endpoint and the respective VNet service present in the Cloudera Control Plane. The private endpoint will be created for all supported components.

The following parameters should be specified:

Parameter	Description
azureAccountDetails	This should have the following fields • azureClientSecretCredential or credentialCrn - Either azureClientSecretCredential or credentialCrn is mandatory. azureClientSecretCredential consists of clientId, clientSecret and tenantId. credentialCrn is configured with default policy or reduced access policy for provisioning the Azure private endpoint • subscriptionId - Azure subscription where the private endpoint needs to be created • location - Azure location • vNetId - Azure Vnet ID in which private endpoint needs to be created • subnetIds - ID of the private subnet in which a private endpoint needs to be created.

The following is an example command for creating Cloudera Private Links Network:

For Using credentialCRN

```
cdp cloudprivatelinks create-private-link-endpoint
 --cli-input-json '{
 "cloudServiceProvider : azure,
  "enablePrivateDns": true,
  "azureAccountDetails": {
  "credentialCrn": "CRN:CDP:ENVIRONMENTS:WESTUS1:CBBE14AC-6DB2-444A-B2C3-
D71E8E4807FF:CREDENTIAL:781F3844-3FDB-4FB1-A5BC-8AF2E04F4691",
        "subscriptionId": "8E3AF657-A8FF-443C-A75C-2FE8C4BCB635",
        "resourceGroup": "RESOURCEGROUPNAME",
    "location": "WESTUS1",
    "vNetId": "VNETNAME",
    "subnetId": "SUBNETNAME"
  }
}'
```

For Using azureClientSecretCredential

```
cdp cloudprivatelinks create-private-link-endpoint
 --cli-input-json '{
"cloudServiceProvider : azure,
  "enablePrivateDns": false,
  "azureAccountDetails": {
    "azureClientSecretCredential": {
      "clientId": "153C0929-9CC0-4226-9DEB-6F6C87879ECB",
      "clientSecret": "SECRET",
      "tenantId": "51F13146-80E9-41F2-8735-B620FF5B6914"
    },
    "subscriptionId": "8E3AF657-A8FF-443C-A75C-2FE8C4BCB635",
    "resourceGroup": "RESOURCEGROUPNAME",
    "location": "WESTUS1",
    "vNetId": "VNETNAME",
    "subnetId": "SUBNETNAME"
  }
}'
```

The executed command performs the following sequence of steps:

1. Identifying the appropriate Private Link service for the request. Existing services are filtered for the requested region.
2. Authorization of your subscription ID for authorizing access to the Private Link service is performed. Acceptance settings on the Private Link service require the subscription ID that can access the endpoint to be added to the auto approved list and visibility list. The visibility setting determines who can request access to the Private Link service. Requests can be automatically approved on a subscription level, if the subscription ID is on the auto approved list.

The command returns a trackingId, which can be used to verify that the domains of the respective Cloudera service components are reachable and resolve to private IPs from your VNet using the following command with the returned tracking ID from creating Private Link endpoints:

```
cdp cloudprivatelinks list-private-link-endpoint-statuses
--tracking-id [***ID***]
```

You can check this by selecting the metric at Monitoring Metrics tab on the Azure Console.

Creating Cloudera Private Links Network with Authorization option

Learn about how to create Cloudera Private Links Network using the Authorization option.

Required Role: EnvironmentCreator or PowerUser

You need to use the following CLI command to create the Cloudera Private Links Network using your own network automation:

```
cdp cloudprivatelinks authorize-private-link-service-access
```

This command is used to authorize access to the Private Link services for your cloud account.

The following parameters should be specified:

Parameter	Description
cloudAccountId	Your Azure subscription ID where the private endpoints are created. The subscription ID needs to be provided, because Cloudera needs to authorize the subscription for Private Link service access.
region	Region of the Cloudera Control Plane.

The following is an example command for creating Cloudera Private Links Network:

```
cdp cloudprivatelinks authorize-private-link-service-access --cli-input-json
 '{
"cloudAccountId" : "8e3af657-a8ff-443c-a75c-2fe8c4bcb635",
"region": "westus1",
}'
```

The executed command performs the following sequence of steps:

1. Identifying the appropriate VNet endpoint service for the request. Existing VNet services are filtered for the requested Cloudera service component and region.
2. Authorization of your subscription ID for authorizing access to the Private Link service is performed. Acceptance settings on the Private Link service require the subscription ID that can access the endpoint to be added to the auto approved list and visibility list. The visibility setting determines who can request access to the Private Link service. Requests can be automatically approved on a subscription level, if the subscription ID is on the auto approved list.

The command returns the Private Link service name, Cloudera service component and the authorization status as shown in the following example:

```
{
"authorizePrivateLinkServiceAccessResults": [
{
"privateLinkService": "cdp-privatelink-enterprise-westus1",
"serviceComponent": "cdp_control_plane",
"authorizationStatus": "SUCCESS",
"vpceClientTcpPortList": "[443]",
"hostname": "[*.altus.cloudera.com, *.us-west-1.ccm.cdp.cloudera.com, *.us-
west-1.cdp.cloudera.com]"
}
],
"status": "SUCCESS"
}
```

After executing the command, you have to manually complete the following steps:

1. Create private links using the private endpoints. For more information, see the Azure documentation.
2. Create private DNS zones as required. For more information, see the Azure documentation.

You can verify that the domains of the respective Cloudera service components are reachable and resolve to private IPs from your VNet using the following command with the returned tracking ID from creating Private Link endpoints:

```
cdp cloudprivatelinks list-private-link-endpoint-statuses
--tracking-id [***ID***]
```

You can check this by selecting the metric at Monitoring Metrics tab on the Azure Console.

Deleting Cloudera Private Links Network

You have the option to delete the created Cloudera Private Links Network using CDP CLI. The command first deletes the security groups and then removes the private link endpoints.

Required Role: EnvironmentCreator or PowerUser

Warning: In case the Cloudera Private Links Network is deleted, traffic will be routed through the internet.

If you want to delete the Cloudera Private Links Network, use the following command:

```
cdp cloudprivatelinks delete-private-link-endpoint
--cloud-service-provider AZURE
--azure-account-info [***ACCOUNT DETAILS***]
```

You can verify that the Cloudera Private Links Network is removed with the following command with the tracking ID returned from deleting the private link endpoints:

```
cdp cloudprivatelinks list-private-link-endpoint-statuses
--tracking-id [***ID***]
```

Troubleshooting Cloudera Private Links Network

Learn more about how to troubleshoot certain errors when creating Cloudera Private Links Network.

The following error message is displayed if the environment creation is failing with a connectivity issue:

```
Please check your connection and proxy settings and make sure the instance c
an reach *.v2.ccm.cdp.cloudera.com
```

In this case, you can login to the FreeIPA instance, and confirm if the respective domain resolves to a private IP. The following domains are available for the Cloudera service component:

Service component	Endpoint
API	api.*[***CLOUDERA CONTROL PLANE REGION***]*.cdp.cl
CCMv2	v2.ccm.*[***CLOUDERA CONTROL PLANE REGION***]*.cc
DBUSAPI	dbusapi.*[***CLOUDERA CONTROL PLANE REGION***]*.s
CONSOLEAUTH	consoleauth.altus-*[***ENVIRONMENT NAME***]*.cloudera.c
MONITORING	api.monitoring.*[***CLOUDERA CONTROL PLANE REGION*

The *[***CLOUDERA CONTROL PLANE REGION***]* should be one of the following:

- US control plane: us-west-1
- EU control plane: eu-1
- AP control plane: ap-1

References

Learn more about additional information related to Cloudera Private Links Network.

CLI commands for Cloudera Private Links Network

Learn more about the available CLI command for Cloudera Private Links Network.

Command	Description
create-private-link-endpoint	Creating private endpoints.
authorize-private-link-service-access	Authorize private link service's access for the Azure subscription ID. It returns the list of private link service names and their respective authorization status.
list-private-link-endpoint-statuses	This command can be used after creating private endpoints using the cdp create-private-link-endpoint command, after deleting private endpoints by using the cdp delete-private-link-endpoint command. Pass the tracking-id returned from the create or delete command to get the status of request.
list-private-link-services-for-region	Returns a list of the private endpoints services that are supported for the Azure region. It returns the list of private endpoint service names and the mapped service component. You can use this command to check which private endpoint services are supported in your Azure region.
delete-private-link-endpoint	Delete a previously created private link endpoint. This first deletes the associated private dns zones and VNet links, and then deletes the endpoints specified.

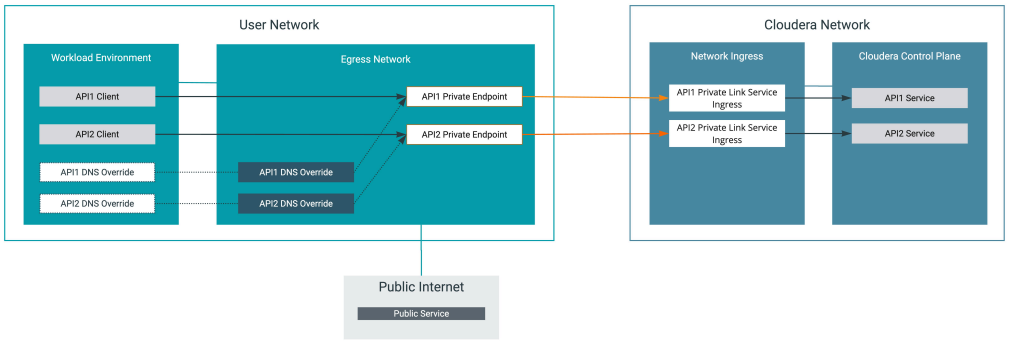
For the full list of command references, see CDP CLI documentation.

Additional VNet scenarios

Learn more about the additional VNet scenarios that show how private endpoints can be configured between your workload environment and Cloudera Control Plane.

Scenario 1: Private endpoint in your egress VNet with no HTTP proxy

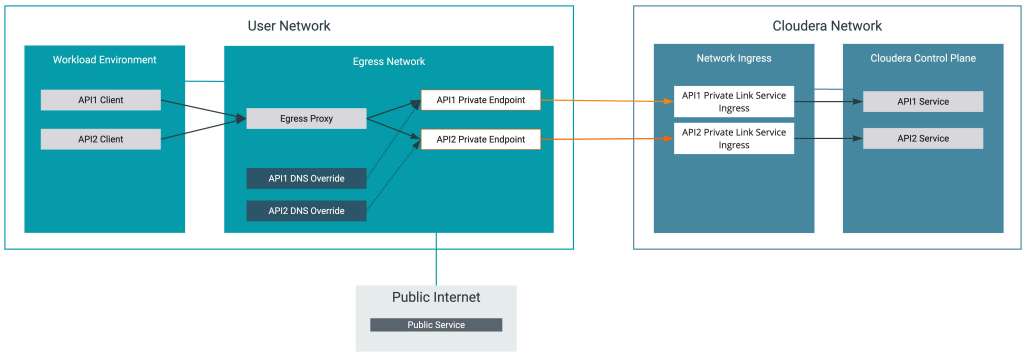
This scenario has the private endpoints in your egress VNet, allowing the private endpoints to be shared by multiple VNets in your network. In this setup, private endpoints receive IPs from egress VNet.



- DNS overrides where DNS is a per-VNet view:
  - DNS override zones and records can be deployed in each workload environment VNet, or a single set of DNS override zones and records are deployed and the zones are associated with each workload environment VNet.
- DNS overrides where DNS is a regional or global view:
  - The DNS override will have a regional or global impact to resolution of the Cloudera hostnames, clients in the region/globally will receive these private endpoints.

Scenario 2: Private endpoint in your egress VNet, HTTP proxy

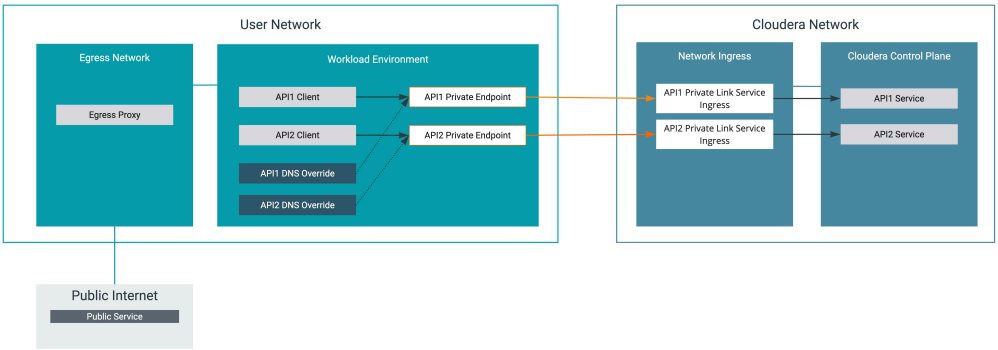
Similar to scenario 1, except you have egress traffic flowing through an HTTP proxy in the egress VNet. In this setup, Private endpoints receive IPs from egress VNet.



- DNS overrides:
  - Transparent proxy or egress firewall policy configurations may require the original destination IP to match the DNS resolution. If this is the case, the override zones/records/ VNet associations can be deployed as described in scenario 1.

Scenario 3: Private endpoint in your workload environment VNet, HTTP proxy

Private endpoints deployed in your workload environment network. In this setup, private endpoints receive IPs from egress VNet.



- DNS overrides where DNS is a regional or global view:
  - The overrides will impact resolution for clients elsewhere in the region and globally.
  - Traffic to these hostnames from outside this VNet will attempt to use these Private Endpoints, which may not be a desired configuration
- HTTP forward proxy or non-transparent proxy
  - Workload environment will be configured to use an HTTP proxy profile.
  - The no_proxy configuration of the profile must include the hostnames of the APIs reachable through private endpoint. HTTP requests for destinations in the no_proxy list will not be forwarded to the proxy, local DNS and therefore the private endpoints will be used for that traffic