Cloudera Enterprise Reference Architecture for AWS Deployments

Important Notice

The information in this document is subject to change without notice. Cloudera shall not be liable for any damages resulting from technical errors or omissions which may be present in this document, or from use of this document.

Cloudera, Inc.

395 Page Mill Road
Palo Alto, CA 94306
info@cloudera.com

US: 1-888-789-1488
Intl: 1-650-362-0488
www.cloudera.com

Release Information

Version: 5.14
Date: January 26, 2018

Table of Contents

Abstract

An organization's requirements for a big-data solution are simple: Acquire and combine any amount or type of data in its original fidelity, in one place, for as long as necessary, and deliver insights to all kinds of users, as quickly as possible.

Cloudera, an enterprise data management company, introduced the concept of the enterprise data hub (EDH): a central system to store and work with all data. The EDH has the flexibility to run a variety of enterprise workloads (for example, batch processing, interactive SQL, enterprise search, and advanced analytics) while meeting enterprise requirements such as integrations to existing systems, robust security, governance, data protection, and management. The EDH is the emerging center of enterprise data management. EDH builds on Cloudera Enterprise, which consists of the open source Cloudera Distribution including Apache Hadoop (CDH), a suite of management software and enterprise-class support.

In addition to needing an enterprise data hub, enterprises are looking to move or add this powerful data management infrastructure to the cloud for operation efficiency, cost reduction, compute and capacity flexibility, and speed and agility.

As organizations embrace Hadoop-powered big data deployments in cloud environments, they also want enterprise-grade security, management tools, and technical support--all of which are part of Cloudera Enterprise.

Customers of Cloudera and Amazon Web Services (AWS) can now run the EDH in the AWS public cloud, leveraging the power of the Cloudera Enterprise platform and the flexibility of the AWS cloud. Cloudera Director enables users to manage and deploy Cloudera Manager and EDH clusters in AWS. Users can create and save templates for desired instance types, spin up and spin down Cloudera Manager and EDH as well as clone clusters. Users can also deploy multiple clusters and can scale up or down to adjust to demand.

Cloudera Reference Architecture documents illustrate example cluster configurations and certified partner products. The Cloud RAs are not replacements for official statements of supportability, rather they're guides to assist with deployment and sizing options. Statements regarding supported configurations in the RA are informational and should be cross-referenced with the latest documentation.

Cloudera on AWS

Cloudera makes it possible for organizations to deploy the Cloudera solution as an EDH in the AWS cloud. This joint solution combines Cloudera's expertise in large-scale data management and analytics with AWS' expertise in cloud computing.

This joint solution provides the following benefits:

Flexible Deployment, Faster Time to Insight

Running Cloudera Enterprise on AWS provides the greatest flexibility in deploying Hadoop. Customers can now bypass prolonged infrastructure selection and procurement processes to rapidly implement the Cloudera big data platform and realize tangible business value from their data immediately. Hadoop excels at large-scale data management, and the AWS cloud provides infrastructure services on demand.

Scalable Data Management

At large organizations, it can take weeks or even months to add new nodes to a traditional data cluster. By deploying Cloudera Enterprise in AWS, enterprises can effectively shorten rest-to-growth cycles to scale their data hubs as their business grows.

On-Demand Processing Power

While Hadoop focuses on collocating compute to disk, many processes benefit from increased compute power. Deploying Hadoop on Amazon allows a fast compute power ramp-up and ramp-down based on specific workloads—flexibility that is difficult to obtain with on-premise deployment.

Improved Efficiency and Increased Cost Savings

Deploying in AWS eliminates the need for dedicated resources to maintain a traditional data center, enabling organizations to focus instead on core competencies. As annual data growth for the average enterprise continues to skyrocket, even relatively new data management systems can strain under the demands of modern high-performance workloads. By moving their data-management platform to the cloud, enterprises can avoid costly annual investments in on-premises data infrastructure to support new enterprise data growth, applications, and workloads.

In this white paper, we provide an overview of best practices for running Cloudera on AWS and leveraging different AWS services such as EC2, S3, and RDS.

AWS Overview

AWS offerings consists of several different services, ranging from storage to compute, to higher up the stack for automated scaling, messaging, queuing, and other services. Cloudera Enterprise deployments can use the following service offerings.

Elastic Compute Cloud (EC2)

With Elastic Compute Cloud (EC2), users can rent virtual machines of different configurations, on demand, for the time required. For this deployment, EC2 instances are the equivalent of servers that run Hadoop. EC2 offers several different types of instances with different pricing options. For Cloudera Enterprise deployments, each individual node in the cluster conceptually maps to an individual EC2 instance. A list of vetted instance types and the roles that they play in a Cloudera Enterprise deployment are described later in this document.

Simple Storage Service (S3)

Simple Storage Service (S3) allows users to store and retrieve various sized data objects using simple API calls. S3 is designed for 99.999999999% durability and 99.99% availability. S3 provides only storage; there is no compute element. The compute service is provided by EC2, which is independent of S3.

Relational Database Service (RDS)

Relational Database Service (RDS) allows users to provision different types of managed relational database instances, including Oracle and MySQL. RDS handles database management tasks, such as backups for a user-defined retention period, point-in-time recovery, patch management, and replication, allowing users to pursue higher value application development or database refinements.

Elastic Block Store (EBS)

Elastic Block Store (EBS) provides block-level storage volumes that can be used as network attached disks with EC2 instances. Users can provision volumes of different capacities with varying IOPS and throughput guarantees. Unlike S3, these volumes can be mounted as network attached storage to EC2 instances and have an independent persistence lifecycle; that is, they can be made to persist even after the EC2 instance has been shut down. At a later point, the same EBS volume can be attached to a different EC2 instance. EBS volumes can also be snapshotted to S3 for higher durability guarantees. Encrypted EBS volumes can be provisioned to protect data in-transit and at-rest with negligible impact to latency.

Direct Connect

Use Direct Connect to establish direct connectivity between your data center and AWS region. You can configure direct connect links with different bandwidths based on your requirement. With this service, you can consider AWS infrastructure as an extension to your data center.

Virtual Private Cloud (VPC)

With Virtual Private Cloud (VPC), you can logically isolate a section of the AWS cloud and provision services inside of that isolated network. Using VPC is recommended to provision services inside AWS and is enabled by default for all new accounts. VPC has various configuration options for accessibility to the Internet and other AWS services. You can create public-facing subnets in VPC, where the instances can have direct access to the public Internet gateway and other AWS services. Instances can be provisioned in private subnets too, where their access to the Internet and other AWS services can be restricted or managed through network address translation (NAT). RDS instances can be accessed from within a VPC.

AWS Service Limits

Amazon places per-region default limits on most AWS services. A few examples include:

- EC2: 10 m4.4xlarge instances per region
- EBS: 20 TB of Throughput Optimized HDD (st1) per region
- S3: 100 buckets per account
- VPC: 5 VPCs per region

The default limits might impact your ability to create even a moderately sized cluster, so plan ahead. Some limits can be increased by submitting a request to Amazon, although these requests typically take a few days to process. For more information on limits for specific services, consult AWS Service Limits.

Deployment Architecture

Deployment Topologies

Two kinds of Cloudera Enterprise deployments are supported in AWS, both within VPC but with different accessibility:

1. Cluster inside a public subnet in VPC
2. Cluster inside a private subnet in VPC

Choosing between the public subnet and private subnet deployments depends predominantly on the accessibility of the cluster, both inbound and outbound, and the bandwidth required for outbound access.

Public Subnet Deployments

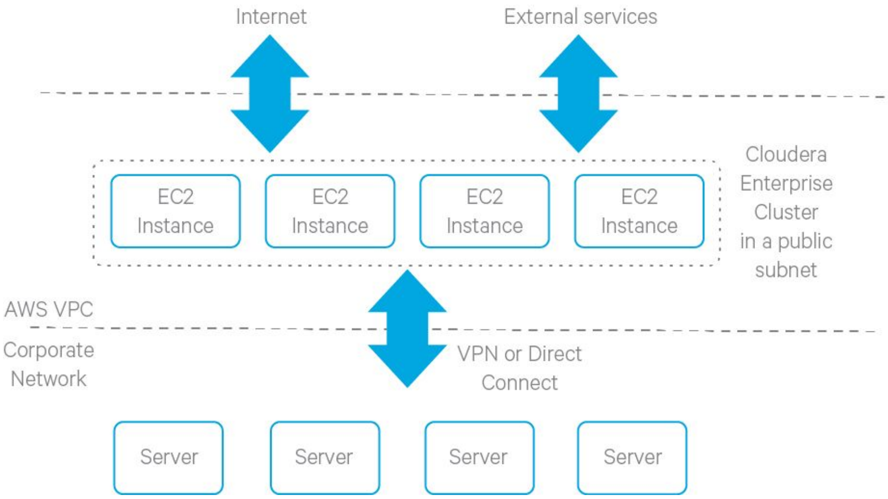
A public subnet in this context is a subnet with a route to the Internet gateway. Instances provisioned in public subnets inside VPC can have direct access to the Internet as well as to other external services such as AWS services in another region. If your cluster requires high-bandwidth access to data sources on the Internet or outside of the VPC, your cluster should be deployed in a public subnet. This gives each instance full bandwidth access to the Internet and other external services. Unless it's a requirement, we don't recommend opening full access to your cluster from the Internet. Using security groups (discussed later), you can configure your cluster to have access to other external services but not to the Internet, and you can limit external access to nodes in the public subnet.

Private Subnet Deployments

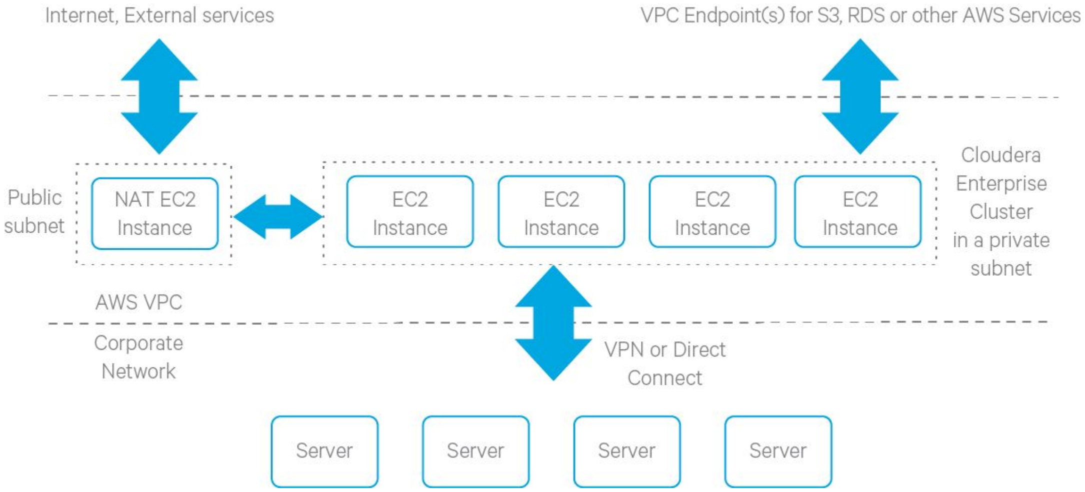
Instances provisioned in private subnets inside VPC don't have direct access to the Internet or to other AWS services, except when a VPC endpoint is configured for that service. To access the Internet, they must go through a NAT instance in the public subnet. If your cluster does not require full bandwidth access to the Internet or to external services, you should deploy in a private subnet. VPC endpoint interfaces or gateways should be used for high-bandwidth access to AWS services.

In both cases, you can set up VPN or Direct Connect between your corporate network and AWS. This makes AWS look like an extension to your network, and the Cloudera Enterprise deployment is accessible as if it were on servers in your own data center.

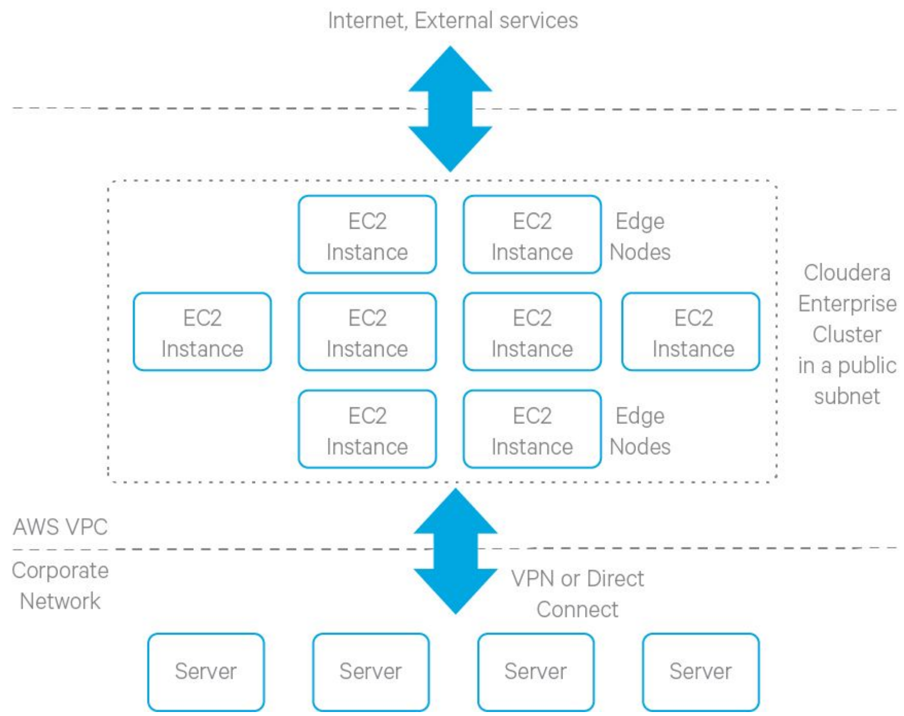
Deployment in the public subnet looks like this:



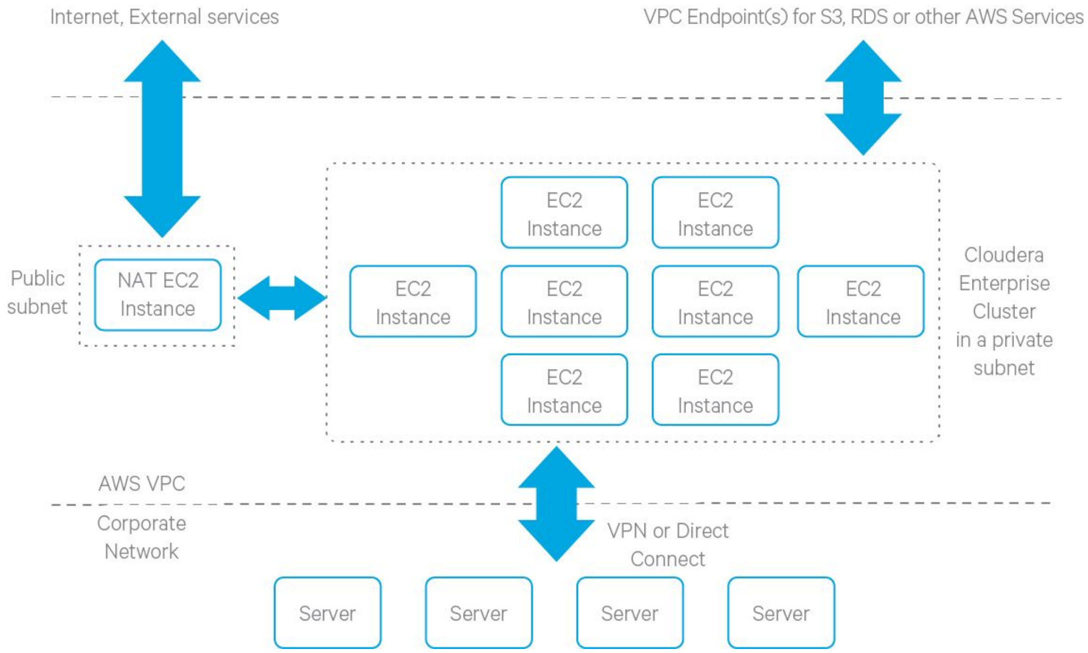
Deployment in the private subnet looks like this:



The accessibility of your Cloudera Enterprise cluster is defined by the VPC configuration and depends on the security requirements and the workload. Typically, there are edge/client nodes that have direct access to the cluster. Users go through these edge nodes via client applications to interact with the cluster and the data residing there. These edge nodes could be running a web application for real-time serving workloads, BI tools, or simply the Hadoop command-line client used to submit or interact with HDFS. The public subnet deployment with edge nodes looks like this:



Deployment in private subnet with edge nodes looks like this:



The edge nodes in a private subnet deployment could be in the public subnet, depending on how they must be accessed. The figure above shows them in the private subnet as one deployment option.

The edge nodes can be EC2 instances in your VPC or servers in your own data center. Cloudera recommends allowing access to the Cloudera Enterprise cluster via edge nodes only. You can configure this in the security groups for the instances that you provision.

Workloads, Roles, and Instance Types

In this reference architecture, we consider different kinds of workloads that are run on top of an Enterprise Data Hub. The initial requirements focus on instance types that are suitable for a diverse set of workloads. As service offerings change, these requirements may change to specify instance types that are unique to specific workloads. You choose instance types based on the workload you run on the cluster. You should also do a cost-performance analysis. Table 1 identifies service roles for node types.

Table 1: Service Roles and Node Types

	Master Node 1-2	Master Node 3	Worker Nodes	Edge Node(s)
HDFS	NN, QJN, FC	QJN, HttpFS	DataNode	
YARN	RM	Job History Server	NodeManager	
Hive		MetaStore WebHCat HiveServer2		
Hue		Hue Server		
Spark		History Server	(runs on YARN)	
Impala		Catalog Server State Store	Impala Daemon	
Cloudera Search			Solr	
HBase	HBase Master	Thrift Server	HBase Region Server	
ZooKeeper	ZooKeeper	ZooKeeper		
Flume				Flume Agent
Gateway Roles				HDFS, YARN, Hive, Sqoop, Hue, etc. as needed
Management Roles	Cloudera Manager Agent	Cloudera Manager Agent, Oozie, Cloudera, Management Services,	Cloudera Manager Agent	Cloudera Manager Agent

		Cloudera Manager[1]		

Cloudera currently recommends RHEL, CentOS, and Ubuntu AMIs on CDH 5.

All EC2 instance types should have:

- Networking Performance of High or 10+ Gigabit or faster (as seen on Amazon Instance Types)
- Enhanced Networking

In addition, instances utilizing EBS volumes -- whether root volumes or data volumes -- should be EBS-optimized OR have 10 Gigabit or faster networking. With the exception of EBS-optimized instances, there are no guarantees about network performance on shared hosts. If the instance type isn't listed with a 10 Gigabit or faster network interface, it's shared. For C4, H1, M4, M5, R4, and D2 instances, EBS optimization is enabled by default at no additional cost.

When sizing instances, allocate two vCPUs and at least 4 GB memory for the operating system. The more services you are running, the more vCPU and memory will be required; you will need to use larger instances to accommodate these needs.

Master Nodes

Management nodes for a Cloudera Enterprise deployment run the master daemons, management, and coordination services, which may include:

- Cloudera Manager
- ResourceManager
- NameNode
- Standby NameNode
- JournalNodes
- ZooKeeper
- Oozie
- Impala Catalog Server
- Impala State Store
- Job History Server

Allocate a vCPU for each master service. The more master services you are running, the larger the instance will need to be. For example, if running YARN, Spark, and HDFS, an instance with eight vCPUs is sufficient (two for the OS plus one for each YARN, Spark, and HDFS is five total and the next smallest instance vCPU count is eight). If you add HBase, Kafka, and Impala, you would pick an instance type with more vCPU and memory. The memory footprint of the master services tend to increase linearly with overall cluster size, capacity, and activity.

Cloudera supports running master nodes on both ephemeral- and EBS-backed instances.

Ephemeral

When deploying to instances using ephemeral disk for cluster metadata, the types of instances that are suitable are limited. Each of the following instance types have at least two HDD or SSD, one each dedicated for DFS metadata and ZooKeeper data.

[1] Cloudera Manager will be on a dedicated node for Director deployments.

- c3.8xlarge
- d2.8xlarge
- h1.8xlarge
- i2.8xlarge
- i3.16xlarge
- r3.8xlarge

Smaller instances in these classes can be used so long as they meet the aforementioned disk requirements; be aware there might be performance impacts and an increased risk of data loss when deploying on shared hosts. If you want to utilize smaller instances, we recommend provisioning in Spread Placement Groups or deploying to Dedicated Hosts such that each master node is placed on a separate physical host.

EBS

Cloudera requires using GP2 volumes when deploying to EBS-backed masters, one each dedicated for DFS metadata and ZooKeeper data. While EBS volumes don't suffer from the disk contention issues that can arise when using ephemeral disks, using dedicated volumes can simplify resource monitoring. Cloudera requires GP2 volumes with a minimum capacity of 100 GB to maintain sufficient IOPs, although volumes can be sized larger to accommodate cluster activity.

Note:

While less expensive per GB, the I/O characteristics of ST1 and SC1 volumes make them unsuitable for the transaction-intensive and latency-sensitive master applications.

When using EBS volumes for masters, use EBS-optimized instances or instances that include 10 Gb/s or faster network connectivity.

- **c4**: 2xlarge, 4xlarge, 8xlarge
- **m4**: xlarge, 2xlarge, 4xlarge, 8xlarge, 10xlarge, 16xlarge
- **m5**: xlarge, 2xlarge, 4xlarge, 12xlarge, 24xlarge
- **r4:** xlarge, 2xlarge, 4xlarge, 8xlarge, 16xlarge

We recommend a minimum Dedicated EBS Bandwidth of 1000 Mbps (125 MB/s).

Worker Nodes

Worker nodes for a Cloudera Enterprise deployment run worker services, which may include:

- HDFS DataNode
- YARN NodeManager
- HBase RegionServer
- Impala Daemons
- Solr Servers

Allocate a vCPU for each worker service. For example an HDFS DataNode, YARN NodeManager, and HBase Region Server would each be allocated a vCPU. You will need to consider the memory requirements of each service. Some services like YARN and Impala can take advantage of additional vCPUs to perform work in parallel. Consider your cluster workload and storage requirements, determine the vCPU and memory resources you wish to allocate to each service, then select an instance type that's capable of satisfying the requirements.

DFS is supported on both ephemeral and EBS storage, so there are a variety of instances that can be utilized for Worker nodes.

EBS

When selecting an EBS-backed instance, be sure to follow the EBS guidance.

- **c4**: 2xlarge, 4xlarge, 8xlarge
- **m4**: xlarge, 2xlarge, 4xlarge, 10xlarge
- **m5**: xlarge, 2xlarge, 4xlarge, 12xlarge, 24xlarge
- **r4**: xlarge, 2xlarge, 4xlarge, 8xlarge, 16xlarge

In addition, any of the D2, I2, or R3 instance types can be used so long as they are EBS-optimized and have sufficient dedicated EBS bandwidth for your workload. If EBS encrypted volumes are required, consult the list of EBS encryption supported instances.

For dedicated Kafka brokers we recommend m4.xlarge or m5.xlarge instances.

Ephemeral

Cloudera recommends the largest instances types in the ephemeral classes to eliminate resource contention from other guests and to reduce the possibility of data loss. Data loss can result from multiple replicas being placed on VMs located on the same hypervisor host. The impact of guest contention on disk I/O has been less of a factor than network I/O, but performance is still not guaranteed.

- c3.8xlarge
- d2.8xlarge
- h1.8xlarge, h1.16xlarge
- i2.8xlarge, i2.16xlarge
- i3.8xlarge, i3.16xlarge
- r3.8xlarge

Smaller instances in these classes can be used; be aware there might be performance impacts and an increased risk of data loss when deploying on shared hosts. We do not recommend using any instance with less than 32 GB memory.

To address Impala's memory and disk requirements, we recommend d2.8xlarge, h1.8xlarge, h1.16xlarge, i2.8xlarge, or i3.8xlarge instances.

Edge Nodes

Hadoop client services run on edge nodes. They are also known as gateway services. Some example services include:

- Third-party tools
- Hadoop command-line client
- Hive command-line client
- Impala command-line client
- Flume agents
- Hue Server
- HBase REST proxy

- HBase Thrift proxy

Edge node services are typically deployed to the same type of hardware as those responsible for master node services, however any instance type can be used for an edge node so long as it has sufficient resources for your use. Depending on the size of the cluster, there may be numerous systems designated as edge nodes.

A detailed list of configurations for the different instance types is available on the EC2 instance types page.

Regions and Availability Zones

Regions are self-contained geographical locations where AWS services are deployed. Regions have their own deployment of each service. Each service within a region has its own endpoint that you can interact with to use the service.

Regions contain availability zones, which are isolated locations within a general geographical location. Some regions have more availability zones than others. While provisioning, you can choose specific availability zones or let AWS select for you.

Cloudera EDH deployments are restricted to single regions. Single clusters spanning regions are not supported. Refer to Appendix A: Spanning AWS Availability Zones for more information.

Networking, Connectivity, and Security

Enhanced Networking

Amazon EC2 provides enhanced networking capacities on supported instance types, resulting in higher performance, lower latency, and lower jitter. In order to take advantage of enhanced networking, you should launch an HVM (Hardware Virtual Machine) AMI in VPC and install the appropriate driver. More details can be found in the Enhanced Networking documentation.

VPC

VPC has several different configuration options. See the VPC documentation for detailed explanation of the options and choose based on your networking requirements. You can deploy Cloudera Enterprise clusters in either public or private subnets. In both cases, the instances forming the cluster should not be assigned a publicly addressable IP unless they must be accessible from the Internet. If you assign public IP addresses to the instances and want to block incoming traffic, you can use security groups.

Connectivity to Other AWS Services

For private subnet deployments, connectivity between your cluster and other AWS services in the same region such as S3 or RDS should be configured to make use of VPC endpoints. VPC endpoints allow configurable, secure, and scalable communication without requiring the use of public IP addresses, NAT or Gateway instances. See the VPC Endpoint documentation for specific configuration options and limitations.

For public subnet deployments, there is no difference between using a VPC endpoint and just using the public Internet-accessible endpoint.

Connectivity to the Internet or Outside of VPC

Clusters that do not need heavy data transfer between the Internet or services outside of the VPC and HDFS should be launched in the private subnet. These clusters still might need access to services like software repositories for updates or other low-volume outside data sources. Do this by provisioning a NAT instance in the public subnet, allowing access outside the private subnet into the public domain. Cloudera does not recommend using the NAT instance for large-scale data movement.

If cluster instances require high-volume data transfer outside of the VPC or to the Internet, they can be deployed in the public subnet with public IP addresses assigned so that they can directly transfer data to and from those services. Configure the security group for the cluster nodes to block incoming connections to the cluster instances.

If you completely disconnect the cluster from the Internet, you block access for software updates as well as to other AWS services that are not configured via VPC Endpoint, which makes maintenance difficult. If you are required to completely lock down any external access because you don't want to keep the NAT instance running all the time, Cloudera recommends starting a NAT instance when external access is required and stopping it when activities are complete.

Private Data Center Connectivity

You can establish connectivity between your data center and the VPC hosting your Cloudera Enterprise cluster by using a VPN or Direct Connect. We recommend using Direct Connect so that there is a dedicated link between the two networks with lower latency, higher bandwidth, security and encryption via IPSec. If you don't need high bandwidth and low latency connectivity between your data center and AWS, connecting to EC2 through the Internet is sufficient and Direct Connect may not be required.

Security Groups

Security Groups are analogous to firewalls. You can define rules for EC2 instances and define allowable traffic, IP addresses, and port ranges. Instances can belong to multiple security groups. Cloudera Enterprise deployments require the following security groups:

- Cluster
- Flume Nodes
- Edge Nodes

Cluster

This security group blocks all inbound traffic except that coming from the security group containing the Flume nodes and edge nodes. You can allow outbound traffic for Internet access during installation and upgrade time and disable it thereafter. You can also allow outbound traffic if you intend to access large volumes of Internet-based data sources.

Flume Nodes

This security group is for instances running Flume agents. Outbound traffic to the Cluster security group must be allowed, and inbound traffic from sources from which Flume is receiving data must be allowed.

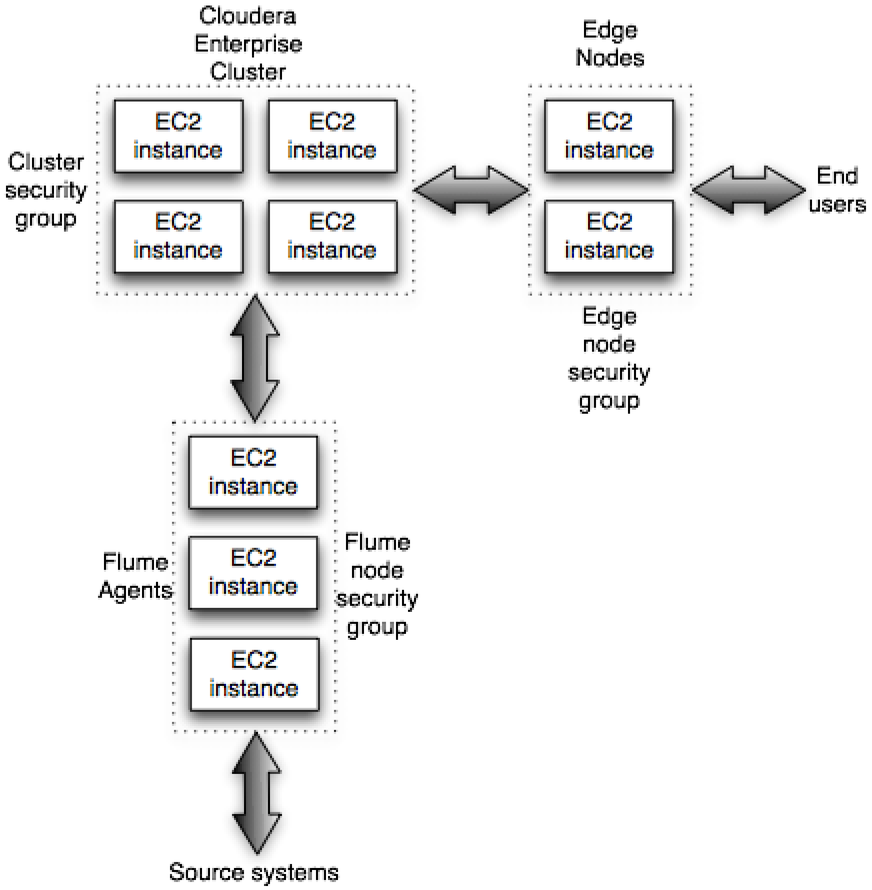
Edge Nodes

This security group is for instances running client applications. Outbound traffic to the Cluster security group must be allowed, and incoming traffic from IP addresses that interact with client applications as well the cluster itself must be allowed.

Each of these security groups can be implemented in public or private subnets depending on the access requirements highlighted above.

Example Deployment

A full deployment looks like the following:



Source systems are where the data is being ingested from using Flume. You'll have flume sources deployed on those machines.

End users are the end clients that interact with the applications running on the edge nodes that can interact with the Cloudera Enterprise cluster.

Placement Groups

As described in the AWS documentation, Placement Groups are a logical grouping of EC2 instances that determine how instances are placed on underlying hardware. Cluster Placement Groups are within a single availability zone, provisioned such that the network between them has higher throughput and lower latency. AWS accomplishes this by provisioning instances as close to each other as possible. This limits the pool of instances available for provisioning but guarantees uniform network performance. Spread Placement Groups ensure that each instance is placed on distinct underlying hardware; you can have a maximum of seven running instances per AZ per group.

Cloudera recommends provisioning the worker nodes of the cluster within a cluster placement group. Edge nodes can be outside the placement group unless you need high throughput and low latency between those and the cluster—for example, if you are moving large amounts of data or expect low-latency responses between the edge nodes and the cluster. Master nodes should be placed within a spread placement group to prevent master metadata loss.

Note:

Attempting to add new instances to an existing cluster placement group or trying to launch more than once instance type within a cluster placement group increases the likelihood of insufficient capacity errors. Restarting an instance may also result in similar failure. Spread Placement Groups aren't subject to these limitations. For more information, refer to the AWS Placement Groups documentation.

Recommended AMIs

Amazon Machine Images (AMIs) are the virtual machine images that run on EC2 instances. These consist of the operating system and any other software that the AMI creator bundles into them. Cloudera Enterprise deployments in AWS recommends Red Hat AMIs as well as CentOS AMIs. Both HVM and PV AMIs are available for certain instance types, but whenever possible Cloudera recommends that you use HVM.

You can find a list of the Red Hat AMIs for each region here. A list of supported operating systems for CDH can be found here, and a list of supported operating systems for Cloudera Director can be found here.

Cloudera Director is unable to resize XFS partitions, which makes creating an instance that uses the XFS filesystem fail during bootstrap. Workaround is to use an image with an ext filesystem such as ext3 or ext4.

To properly address newer hardware, D2 instances require RHEL/CentOS 6.6 (or newer) or Ubuntu 14.04 (or newer).

Storage Options and Configuration

AWS offers different storage options that vary in performance, durability, and cost.

Instance Storage

EC2 instances have storage attached at the instance level, similar to disks on a physical server. The storage is virtualized and is referred to as ephemeral storage because the lifetime of the storage is the same as the lifetime of your EC2 instance. If you stop or terminate the EC2 instance, the storage is lost. The storage is not lost on restarts, however. Different EC2 instances have different amounts of instance storage, as highlighted above. For long-running Cloudera Enterprise clusters, the HDFS data directories should use instance storage, which provide all the benefits of shipping compute close to the storage and not reading remotely over the network.

When using instance storage for HDFS data directories, special consideration should be given to backup planning. Since the ephemeral instance storage will not persist through machine shutdown or failure, you should ensure that HDFS data is persisted on durable storage before any planned multi-instance shutdown and to protect against multi-VM datacenter events. You can set up a scheduled distcp operation to persist data to AWS S3 (see the examples in the distcp documentation) or leverage Cloudera Manager's Backup and Data Recovery (BDR) features to backup data on another running cluster.

Elastic Block Storage

Amazon Elastic Block Store (EBS) provides persistent block level storage volumes for use with Amazon EC2 instances. HDFS data directories can be configured to use EBS volumes. Encrypted EBS volumes can be used to protect data in-transit and at-rest, with negligible impact to latency or throughput.

There are different types of volumes with differing performance characteristics: the Throughput Optimized HDD (st1) and Cold HDD (sc1) volume types are well suited for DFS storage. From the Amazon ST1/SC1 release announcement:

> These magnetic volumes provide baseline performance, burst performance, and a burst credit bucket. While [GP2] volumes define performance in terms of IOPS (Input/Output Operations Per Second), [these] volumes define it in terms of throughput (MB/s).

Baseline and burst performance both increase with the size of the provisioned EBS volume. For example, a 500 GB ST1 volume has a baseline throughput of 20 MB/s whereas a 1000 GB ST1 volume has a baseline throughput of 40 MB/s.

A few considerations when using EBS volumes for DFS:

AMI Selection

Use HVM, not paravirtualization (PV).

For kernels > 4.2 (which does not include CentOS 7.2) – set kernel option `xen_blkfront.max=256`

```
kernel /boot/vmlinuz-4.4.5-15.26.amzn1.x86_64 root=LABEL=/ console=ttyS0
xen_blkfront.max=256
```

In order to take advantage of Enhanced Networking, you should launch an HVM AMI in VPC and install the appropriate driver. Enhanced Networking is currently supported in C4, C3, H1, R3, R4, I2, M4, M5, and D2 instances.

Instance selection

When using EBS volumes for DFS storage, use EBS-optimized instances or instances that include 10 Gb/s or faster network connectivity.

- c4.2xlarge, c4.4xlarge, c4.8xlarge
- m4.xlarge, m4.2xlarge, m4.4xlarge, m4.8xlarge, m4.10xlarge
- m5.xlarge, m5.2xlarge, m5.4xlarge, m5.12xlarge, m5.24xlarge
- r4.xlarge, r4.2xlarge, r4.4xlarge, r4.8xlarge, r4.16xlarge

All of these instance types support EBS encryption.

We recommend a minimum Dedicated EBS Bandwidth of 1000 Mbps (125 MB/s).

Important:

When running Impala on M5 and C5 instances, use CDH 5.14 or later. Older versions of Impala can result in crashes and incorrect results on CPUs with AVX512; workarounds are available, but incur significant performance loss. See IMPALA-6291 for more details.

EBS Volume Selection

ST1 and SC1 volumes have different performance characteristics and pricing. The throughput of ST1 and SC1 volumes can be comparable, so long as they are sized properly. For example, to achieve 40 MB/s baseline performance the volume must be sized as follows:

- a 1,000 GB ST1 volume
- a 3,200 GB SC1 volume

With identical baseline performance, the SC1 burst performance provides slightly higher throughput than its ST1 counterpart.

We recommend a minimum size of 1,000 GB for ST1 volumes (3,200 GB for SC1 volumes) to achieve baseline performance of 40 MB/s.

Do not exceed an instance's dedicated EBS bandwidth! The sum of the mounted volumes' baseline performance should not exceed the instance's dedicated EBS bandwidth. For example, an m4.2xlarge instance has 125 MB/s of dedicated EBS bandwidth. Mounting four 1,000 GB ST1 volumes (each with 40 MB/s baseline performance) would place up to 160 MB/s load on the EBS bandwidth, exceeding the instance's capacity. Not only will the volumes be unable to operate to their baseline specification, the instance won't have enough bandwidth to benefit from burst performance.

To prevent device naming complications, do not mount more than 26 EBS volumes on a single instance. That includes EBS root volumes. For example, assuming one (1) EBS root volume do not mount more than 25 EBS data volumes.

Note:

On the largest instance type of each class -- where there are no other guest VMs -- dedicated EBS bandwidth can be exceeded to the extent that there is available network bandwidth. This can provide considerable bandwidth for burst throughput. This behavior has been observed on m4.10xlarge and c4.8xlarge instances.

EBS Volume Tuning

Per EBS performance guidance, increase read-ahead for high-throughput, read-heavy workloads on st1 and sc1:

```
$ sudo blockdev --setra 2048 /dev/<device>
```

To verify the read-ahead:

```
$ sudo blockdev --report /dev/<device>
```

These commands do not persist on reboot, so they'll need to be added to rc.local or equivalent post-boot script.

Also keep in mind, "for maximum consistency, HDD-backed volumes must maintain a queue length (rounded to the nearest whole number) of 4 or more when performing 1 MiB sequential I/O."

To avoid significant performance impacts, Cloudera recommends initializing EBS volumes when restoring DFS volumes from snapshot. Freshly provisioned EBS volumes are not affected.

Simple Storage Service

We strongly recommend using S3 to keep a copy of the data you have in HDFS for disaster recovery. The durability and availability guarantees make it ideal for a cold backup that you can restore in case the primary HDFS cluster goes down. For a hot backup, you need a second HDFS cluster holding a copy of your data.

You can also directly make use of data in S3 for query operations using Hive and Spark. Standard data operations can read from and write to S3. Hive does not currently support use of reference scripts or JAR files located in S3 or LOAD DATA INPATH operations between different filesystems (example: HDFS to S3).

Root Device

We require using EBS volumes as root devices for the EC2 instances. When instantiating the instances, you can define the root device size. The root device size for Cloudera Enterprise clusters should be at least 500 GB to allow parcels and logs to be stored. You should not use any instance storage for the root device.

Capacity Planning

Using AWS allows you to scale your Cloudera Enterprise cluster up and down easily. If your storage or compute requirements change, you can provision and deprovision instances and meet your requirements quickly, without buying physical servers. However, some advance planning makes operations easier. You must plan for whether your workloads need a high amount of storage capacity or not. The available EC2 instances have different amounts of memory, storage, and compute, and deciding which instance type and generation make up your initial deployment depends on the storage and workload requirement. The operational cost of your cluster depends on the type and number of instances you choose, the storage capacity of EBS volumes, and S3 storage and usage.

Low Storage Density

For use cases with lower storage requirements, using r3.8xlarge or c4.8xlarge is recommended. They provide a lower amount of storage per instance but a high amount of compute and memory resources to go with it. For more storage, consider h1.8xlarge.

High Storage Density

For use cases with higher storage requirements, using d2.8xlarge is recommended. These provide a high amount of storage per instance, but less compute than the r3 or c4 instances. The d2.8xlarge instances have 24 x 2 TB instance storage. h1.8xlarge and h1.16xlarge also offer a good amount of local storage with ample processing capability (4 x 2TB and 8 x 2TB respectively).

Reserved Instances

AWS offers the ability to reserve EC2 instances up front and pay a lower per-hour price. This is beneficial for users that are using EC2 instances for the foreseeable future and will keep them on a majority of the time. Reserving instances can drive down the TCO significantly of long-running Cloudera Enterprise clusters. There are different options for reserving instances in terms of the time period of the reservation and the utilization of each instance. See the AWS documentation to plan instance reservation.

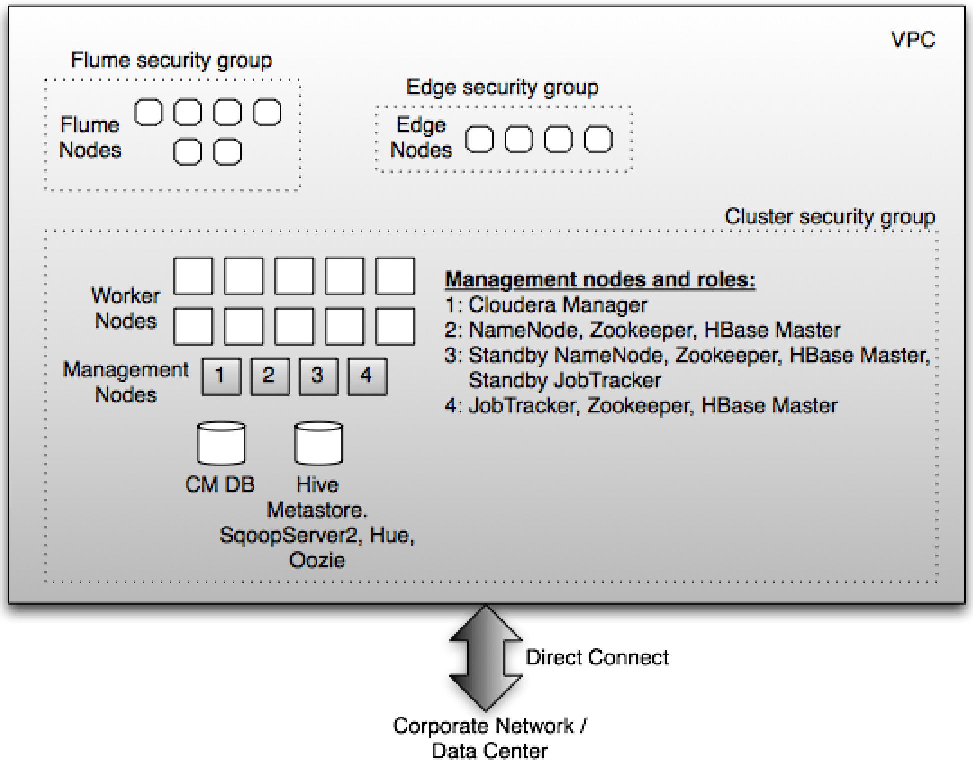
Relational Databases

Cloudera Enterprise deployments require relational databases for the following components:

- Cloudera Manager databases
- Hive and Impala metastore
- Hue database
- Oozie database

For operating relational databases in AWS, you can either provision EC2 instances and install and manage your own database instances, or you can use RDS. The list of supported database types and versions is available here.

With all the considerations highlighted so far, a deployment in AWS would look like (for both private and public subnets):


VPC
Flume security group
Flume Nodes
Edge security group
Edge Nodes
Cluster security group
Worker Nodes
Management Nodes
1
2
3
4
Management nodes and roles:
1: Cloudera Manager
2: NameNode, Zookeeper, HBase Master
3: Standby NameNode, Zookeeper, HBase Master, Standby JobTracker
4: JobTracker, Zookeeper, HBase Master
CM DB
Hive Metastore. SqoopServer2, Hue, Oozie
Direct Connect
Corporate Network / Data Center

Installation and Software Configuration

Provisioning Instances

Cloudera Director can be used to provision EC2 instances.

To provision EC2 instances manually, first define the VPC configurations based on your requirements for aspects like access to the Internet, other AWS services, and connectivity to your corporate network. You can then use the EC2 command-line API tool or the AWS management console to provision instances. You must create a keypair with which you will later log into the instances. In Red Hat AMIs, you will use this keypair to log in as ec2-user, which has sudo privileges.

No matter which provisioning method you choose, make sure to specify the following:

- Root device size of at least 500 GB
- Ephemeral storage devices or recommended GP2 EBS volumes to be used for master metadata
- Ephemeral storage devices or recommended ST1/SC1 EBS volumes to be attached to the instances
- Tags to indicate the role that the instance will play (this makes identifying instances easier).

Along with instances, relational databases must be provisioned (RDS or self managed). If you are provisioning in a public subnet, RDS instances can be accessed directly. If you are deploying in a private subnet, you either need to configure a VPC Endpoint, provision a NAT instance to access RDS instances, or you must set up database instances on EC2 inside the private subnet. The database credentials are required during Cloudera Enterprise installation.

Setting Up Instances

Once the instances are provisioned, you must perform the following to get them ready for deploying Cloudera Enterprise:

- Disable iptables
- Disable SELinux
- Format and mount the instance storage or EBS volumes
- Resize the root volume if it does not show full capacity

When enabling Network Time Protocol (NTP) for use in a private subnet, consider using Amazon Time Sync Service as a time source. The service uses a link local IP address (169.254.169.123) which means you don't need to configure external Internet access. Note: The service is not currently available for C5 and M5 instances.

For more information on operating system preparation and configuration, see the Cloudera Manager installation instructions.

Deploying Cloudera Enterprise

If you are using Cloudera Manager, log into the instance that you have elected to host Cloudera Manager and follow the Cloudera Manager installation instructions.

If you are using Cloudera Director, follow the Cloudera Director installation instructions.

Cloudera Enterprise Configuration Considerations

HDFS

Durability

For Cloudera Enterprise deployments in AWS, the recommended storage options are ephemeral storage or ST1/SC1 EBS volumes.

Data stored on ephemeral storage is lost if instances are stopped, terminated, or go down for some other reason. Data persists on restarts, however. Data durability in HDFS can be guaranteed by keeping replication at three (3). Cloudera does not recommend lowering the replication factor. A persistent copy of all data should be maintained in S3 to guard against cases where you can lose all three copies of the data. Do this by either writing to S3 at ingest time or distcp-ing datasets from HDFS afterwards.

Data stored on EBS volumes persists when instances are stopped, terminated, or go down for some other reason, so long as the “delete on terminate” option is not set for the volume.

Availability

HDFS availability can be accomplished by deploying the NameNode with high availability with at least three JournalNodes.

ZooKeeper

We recommend running at least three ZooKeeper servers for availability and durability.

Flume

For durability in Flume agents, use memory channel or file channel. Flume’s memory channel offers increased performance at the cost of no data durability guarantees. File channels offer a higher level of durability guarantee because the data is persisted on disk in the form of files. Cloudera supports file channels on ephemeral storage as well as EBS. If the EC2 instance goes down, the data on the ephemeral storage is lost. For guaranteed data delivery, use EBS-backed storage for the Flume file channel.

Summary

Cloudera and AWS allow users to deploy and use Cloudera Enterprise on AWS infrastructure, combining the scalability and functionality of the Cloudera Enterprise suite of products with the flexibility and economics of the AWS cloud. This white paper provided reference configurations for Cloudera Enterprise deployments in AWS. These configurations leverage different AWS services such as EC2, EBS, S3, and RDS.

Appendix A: Spanning AWS Availability Zones

Spanning a CDH cluster across multiple Availability Zones (AZs) can provide highly available services and further protect data against AWS host, rack, and datacenter failures.

We recommend the following deployment methodology when spanning a CDH cluster across multiple AWS AZs.

AWS Provisioning

Provision all EC2 instances in a single VPC but within different subnets (each located within a different AZ). In this way the entire cluster can exist within a single Security Group (SG) which can be modified to allow traffic to and from itself.

Deploy across three (3) AZs within a single region. This might not be possible within your preferred region as not all regions have three or more AZs.

Note: Network latency is both higher and less predictable across AWS regions. We do not recommend or support spanning clusters across regions.

CDH Deployment

Deploy HDFS NameNode in High Availability mode with Quorum Journal nodes, with each master placed in a different AZ. For example, if you've deployed the primary NameNode to us-east-1b you would deploy your standby NameNode to us-east-1c or us-east-1d. You should place a QJN in each AZ.

Although HDFS currently supports only two NameNodes, the cluster can continue to operate if any one host, rack, or AZ fails:

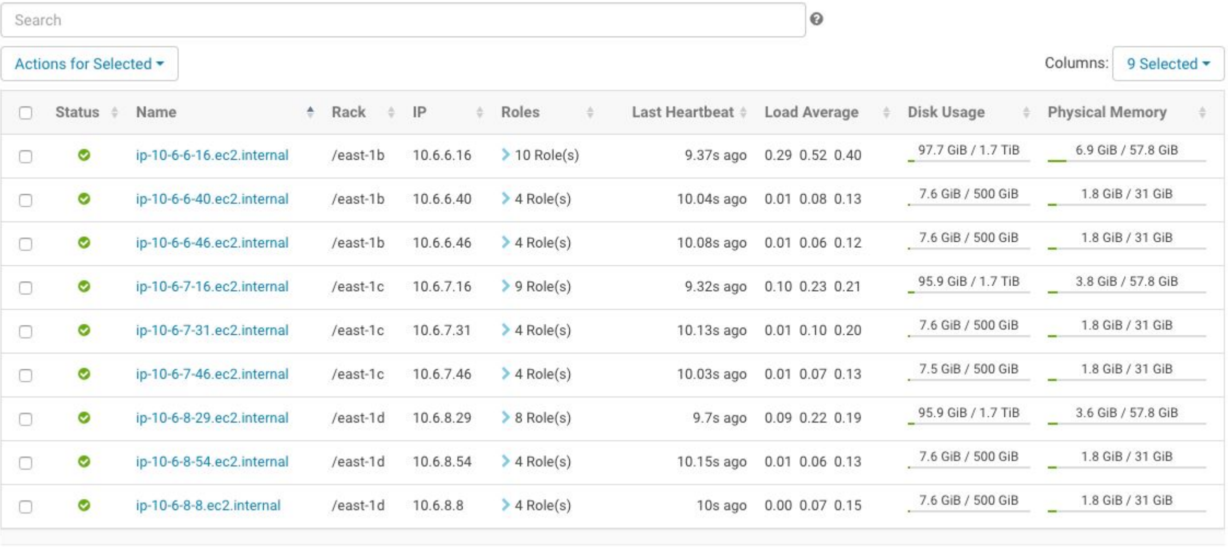
- lose active NameNode, standby NameNode takes over
- lose standby NameNode, active is still active; promote 3rd AZ master to be new standby NameNode
- lose AZ without any NameNode, still have two viable NameNodes

Deploy YARN ResourceManager nodes in a similar fashion.

Deploy a three node ZooKeeper quorum, one located in each AZ.

Deploy edge nodes to all three AZ and configure client application access to all three.

Configure rack awareness, one rack per AZ. See the following screenshot for an example.



□	Status	Name	Rack	IP	Roles	Last Heartbeat	Load Average	Disk Usage	Physical Memory
□	✓	ip-10-6-6-16.ec2.internal	/east-1b	10.6.6.16	10 Role(s)	9.37s ago	0.29 0.52 0.40	97.7 GiB / 1.7 TiB	6.9 GiB / 57.8 GiB
□	✓	ip-10-6-6-40.ec2.internal	/east-1b	10.6.6.40	4 Role(s)	10.04s ago	0.01 0.08 0.13	7.6 GiB / 500 GiB	1.8 GiB / 31 GiB
□	✓	ip-10-6-6-46.ec2.internal	/east-1b	10.6.6.46	4 Role(s)	10.08s ago	0.01 0.06 0.12	7.6 GiB / 500 GiB	1.8 GiB / 31 GiB
□	✓	ip-10-6-7-16.ec2.internal	/east-1c	10.6.7.16	9 Role(s)	9.32s ago	0.10 0.23 0.21	95.9 GiB / 1.7 TiB	3.8 GiB / 57.8 GiB
□	✓	ip-10-6-7-31.ec2.internal	/east-1c	10.6.7.31	4 Role(s)	10.13s ago	0.01 0.10 0.20	7.6 GiB / 500 GiB	1.8 GiB / 31 GiB
□	✓	ip-10-6-7-46.ec2.internal	/east-1c	10.6.7.46	4 Role(s)	10.03s ago	0.01 0.07 0.13	7.5 GiB / 500 GiB	1.8 GiB / 31 GiB
□	✓	ip-10-6-8-29.ec2.internal	/east-1d	10.6.8.29	8 Role(s)	9.7s ago	0.09 0.22 0.19	95.9 GiB / 1.7 TiB	3.6 GiB / 57.8 GiB
□	✓	ip-10-6-8-54.ec2.internal	/east-1d	10.6.8.54	4 Role(s)	10.15s ago	0.01 0.06 0.13	7.6 GiB / 500 GiB	1.8 GiB / 31 GiB
□	✓	ip-10-6-8-8.ec2.internal	/east-1d	10.6.8.8	4 Role(s)	10s ago	0.00 0.07 0.15	7.6 GiB / 500 GiB	1.8 GiB / 31 GiB

Considerations

There are data transfer costs associated with EC2 network data sent between AZ.

DFS throughput will be less than if cluster nodes were provisioned within a single AZ and considerably less than if nodes were provisioned within a single Cluster Placement Group.

Network throughput and latency vary based on AZ and EC2 instance size and neither are guaranteed by AWS. Expect a drop in throughput when a smaller instance is selected and a slight increase in latency as well; both ought to be verified for suitability before deploying to production.

References

Cloudera Enterprise

Cloudera

Cloudera Product Documentation

Cloudera Services & Support

Amazon Web Services

Amazon Web Services

- Developer Tools
- Direct Connect
- Red Hat on AWS
- Support

Amazon EC2

- Instance Lifecycle
- Network & Security

Amazon S3

Amazon Relational Database Service (RDS)

Amazon Virtual Private Cloud (VPC)

v5.14-20180126