Cloudera Enterprise Reference Architecture for Cloud 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.x, 6.x
Date: 9/15/2018

Table of Contents

Abstract

Cloudera in the Cloud

Cloud Overview

Deployment Architecture

- System Architecture Best Practices
  - Java
  - Right-size Server Configurations
  - General Cluster Architecture
- Deployment Topologies
  - Public Subnet Deployments
  - Private Subnet Deployments
- Workloads, Roles, and Instance Types
  - Master Nodes
  - Utility Nodes
  - Worker Nodes
  - Basic Instance Definitions
  - Edge Nodes
- Regions and Availability Zones
- Networking, Connectivity, and Security
  - Enhanced Networking
  - VPC
  - Connectivity to Other the Cloud Services
  - Connectivity to the Internet or Outside of VPC
  - Private data center Connectivity
  - Security Groups
  - Example Deployment
- Storage Options and Configuration
  - Ephemeral/Instance Storage
  - Remote Block Storage (RBS)
    - RBS Volume Selection
  - Root Device

- Capacity Planning

Installation and Software Configuration

- Provisioning Instances
- Setting Up Instances
- Deploying Cloudera Enterprise
- Cloudera Enterprise Configuration Considerations
  - HDFS
  - ZooKeeper
  - Flume
- Security Integration

Appendix A: Spanning Cloud Availability Zones

- Cloud Provisioning
- CDH Deployment
- Considerations

Appendix B: Customer Self-test framework

- Objective
- Distributed iperf3 HOWTO
- Teragen/Terasort/Teravalidate
- Storage Acceptance Criteria Guide microbenchmarks

References

- Cloudera Enterprise

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 can now run the EDH in any Cloud, leveraging the power of the Cloudera Enterprise platform and the flexibility of the chosen Cloud platform.

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 in the Cloud

Cloudera makes it possible for organizations to deploy the Cloudera solution as an EDH in cloud computing platforms. Provisioning Cloudera EDH in Clouds has the following advantages:

Flexible Deployment, Faster Time to Insight

Running Cloudera Enterprise in a Cloud 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 Clouds provide 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 Clouds, 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 Clouds 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 Clouds 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.

Cloud Overview

Cloud offerings consist 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. In this document, Cloud refers to both Public as well as Private Cloud services. Given that there are many Public and Private Cloud solutions, this document refers to key concepts in Cloud (IaaS) solutions in as generic a language as possible. The reader is urged to read through this document first and then map their implemented/selected Cloud solution for specific terminology to the generic concepts provided here.

Infrastructure as a Service (IaaS) solutions

With IaaS, users can rent virtual machines of different configurations, on demand, for the time required. For this deployment, IaaS instances are the equivalent of servers that run Hadoop. For Cloudera Enterprise deployments, each individual node in the cluster conceptually maps to an individual IaaS instance.

Remote Block Store (RBS)

Remote Block Storage provides block-level storage volumes that can be used as network attached disks with IaaS instances. Users can provision volumes of different capacities with varying IOPS and throughput guarantees[1]. These volumes can be mounted as network attached storage to IaaS instances and have an independent persistence lifecycle; that is, they can be made to persist even after the IaaS instance has been shut down. At a later point, the same RBS volume can be attached to a different IaaS instance.

On-Prem to Cloud Connectivity

Clouds have provisions to allow dedicated connectivity from Customer data centers to their own data centers[2]. The terminology varies depending on the Cloud vendor, but conceptually as long as a guaranteed and dedicated network pipe is available between your data center and the Cloud regional data center(s), you can consider the IaaS infrastructure as an extension to your data center.

Virtual Private Cloud[3]

VPC is typically provided by all Cloud vendors. VPC uses the concept of a virtualized overlay network enabled using software defined networking (SDN) features such as VXLAN. Using these overlay networks, you can logically isolate a section of the IaaS cloud and provision services within that isolated network. If available, it is recommended to provision services using these VPCs. VPCs may offer various configuration options for accessibility to the Internet and other cloud services. You can create public-facing subnets, wherein the instances can have direct access to the public Internet gateway and other cloud services. Instances can be provisioned in private subnets too, where their access to the

[1] Will vary depending on the Cloud and its offerings

[2] If the Cloud is a Private Cloud, there might still be connectivity implications with scenarios such as - The legacy data center and private cloud data centers are physically separate and require some connectivity using dedicated Network connections.

[3] The concept of Virtual Private Clouds might be implemented under other names depending on the cloud provider.

Internet and other cloud services can be restricted or managed through some form of network address translation (NAT).

IaaS Service Limits

Clouds typically place per-location default limits on most IaaS services. A few examples include:

- IaaS: *N*[4] Large instances per region/location (IaaS instance SKUs vary from region to region)
- RBS: *N* TB of RBS volumes per region
- VPC: *N* VPCs per region/location

The default limits might impact your ability to create even a moderately sized cluster, so work with your Cloud vendor and plan ahead. These limits can be increased by submitting a request to the Cloud Provider, although these requests typically take a few days to process.

[4] Here “*N*” would vary depending on the Public Cloud provider or Private Cloud solution. For instance, OpenStack default quota for any given project is 10 instances, 20 cores, 512GB of RAM, 1TB of storage and so on.

Deployment Architecture

System Architecture Best Practices

This section describes Cloudera recommendations and best practices applicable to Hadoop cluster system architecture.

Java

Cloudera Manager and CDH are certified to run on Oracle JDK. At this time OpenJDK is not supported. Cloudera distributes a compatible version of the Oracle JDK through the Cloudera Manager repository. Customers are also free to install a compatible version of the Oracle JDK distributed by Oracle.

Refer to Cloudera Enterprise Requirements and Supported Versions for a list of supported JDK versions.

Right-size Server Configurations

Cloudera recommends deploying three or four machine types into production:

- **Master Node.** Runs the Hadoop master daemons: NameNode, Standby NameNode, YARN Resource Manager and History Server, the HBase Master daemon, Sentry server, and the Impala StateStore Server and Catalog Server. Master nodes are also the location where ZooKeeper and JournalNodes are installed. The daemons can often share a single pool of servers, but depending on the cluster size the roles may each be run on a dedicated server. Kudu Master Servers should also be deployed on master nodes.
- **Worker Node.** Runs the HDFS DataNode, YARN NodeManager, HBase RegionServer, Impala impalad, Search worker daemons, and Kudu Tablet Servers.
- **Utility Node.** Runs Cloudera Manager and the Cloudera Management Services. It can also host a MySQL (or another supported) database instance, which is used by Cloudera Manager, Hive, Sentry, and other Hadoop-related projects.
- **Edge Node.** Contains all client-facing configurations and services, including gateway configurations for HDFS, YARN, Impala, Hive, and HBase. The edge node is also a good place for Hue, Oozie, HiveServer2, and Impala HAProxy. HiveServer2 and Impala HAProxy serve as a gateway to external applications such as Business Intelligence (BI) tools.

For more information refer to Cluster Hosts and Role Assignments.

> **Note:**
>
> The edge and utility nodes can be combined in smaller clusters, however in cloud environments it’s often more practical to provision dedicated instances for each.

General Cluster Architecture

This section contains general guidelines for cluster layout, assuming the usage of HDFS, YARN, Hive, Impala, Hue, Oozie, and ZooKeeper.

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 host types that are suitable for a diverse set of workloads. As service offerings change, these requirements may change to specify host types that are unique to specific workloads. You choose host types based on the workload you run on the cluster. You should also do a cost-performance analysis.

The Cloudera documentation on Recommended Cluster Hosts and Role Distribution covers scenarios of how to allocate roles and the recommended number of nodes and so on, for varying cluster sizes.

Encryption Infrastructure (for all cluster sizes)

- 4 Utility nodes
  - Key Management Server (2)
  - Key Trustee Server (2)

Further detail can be found in the Data at Rest Encryption documentation.

Note:

Edge nodes in the medium and large cluster architectures are mostly driven by use cases. Complex ingest pipelines and/or lots of client access will require more edge nodes to handle the load.

Note:

Without three master servers, Kudu does not provide HA. Five masters can be utilized; in this case, the loss of 2 Kudu Master Servers will be tolerated.

Deployment Topologies

Two kinds of Cloudera Enterprise deployments are supported in the Cloud, 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

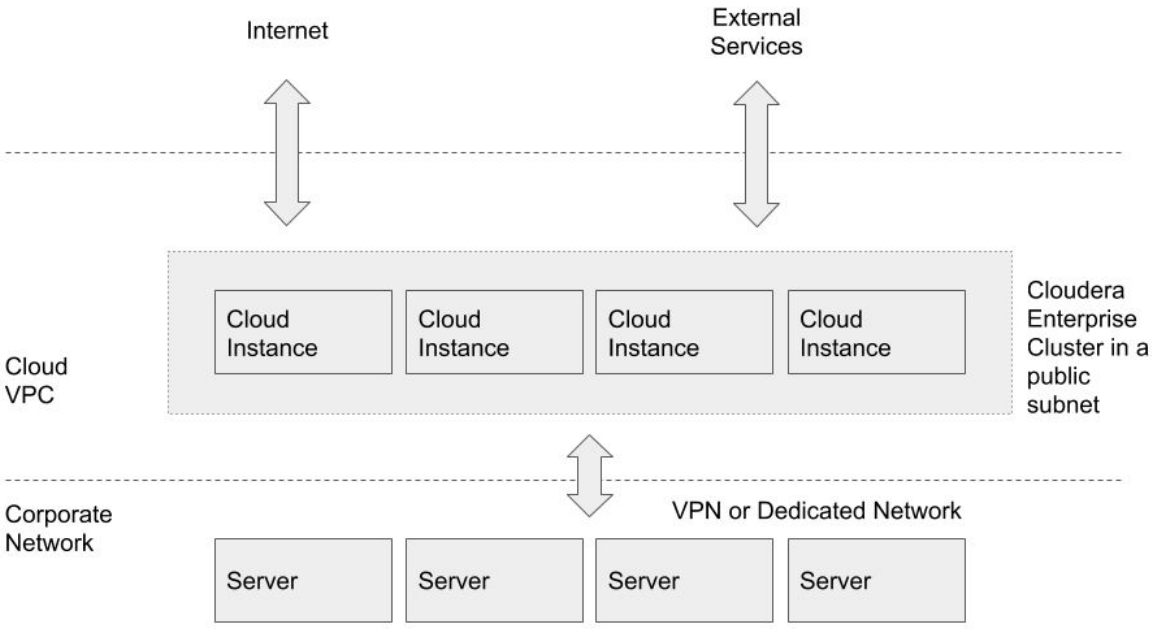
A public subnet in this context is a subnet with a route to the Internet gateway. Instances provisioned in public subnets inside VPC can have direct access to the Internet as well as to other external services such as the Cloud Services in another region. If your cluster requires high-bandwidth access 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

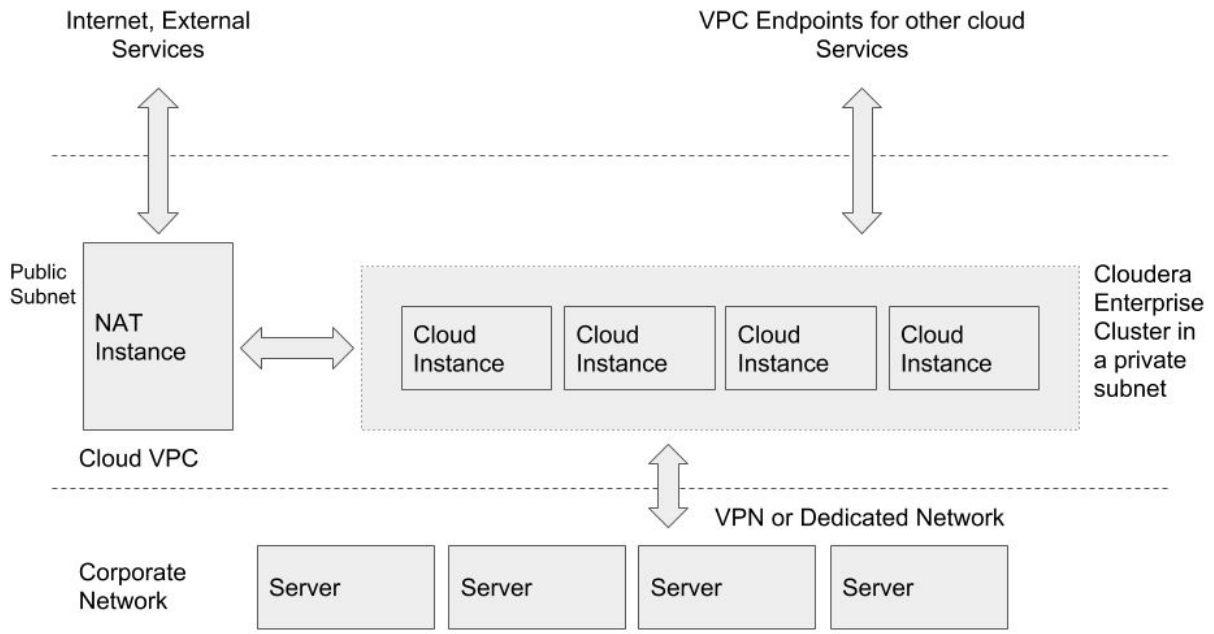
Instances provisioned in private subnets inside VPC don't have direct access to the Internet or to other the Cloud Services, except when a VPC endpoint is configured for that service. To access the Internet, they must go through a NAT gateway or NAT instance in the public subnet; NAT gateways provide better availability, higher bandwidth, and require less administrative effort. 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 the Cloud Services.

In both cases, you can set up VPN or dedicated network connection between your corporate network and the Cloud. This makes the cloud 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.

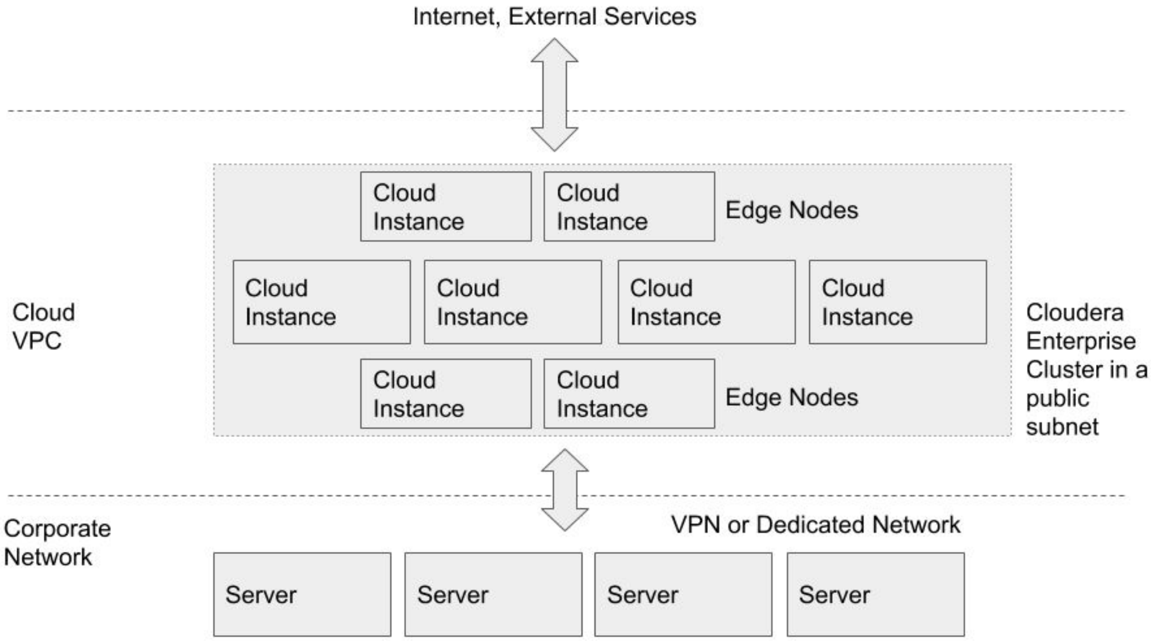
Deployment in the public subnet looks like this:



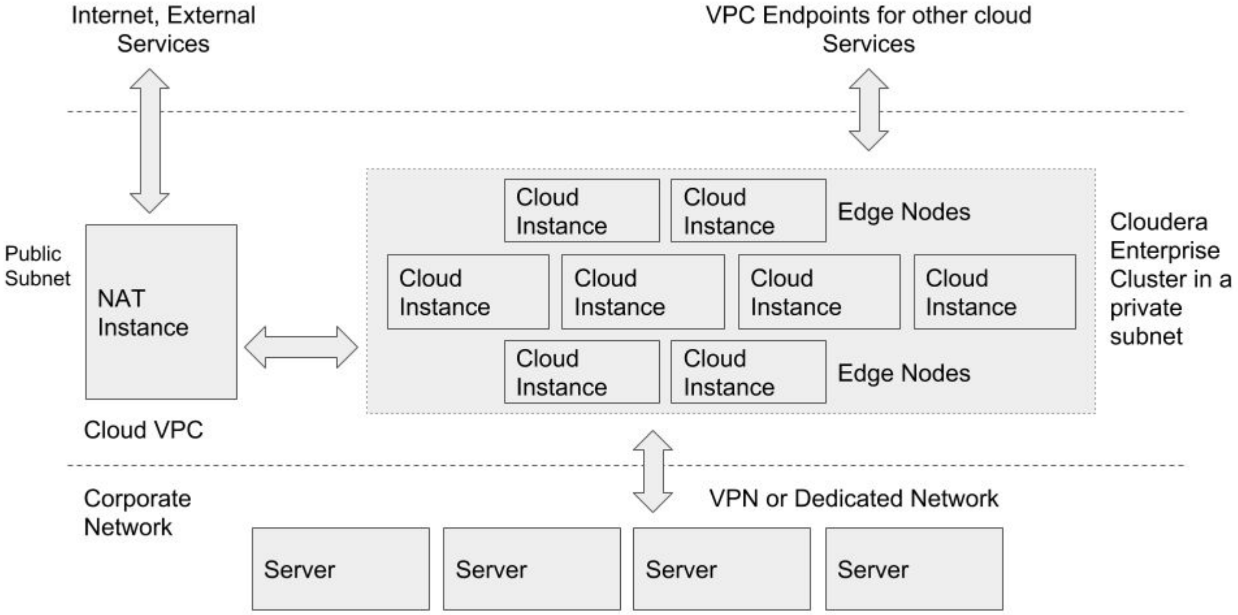
Deployment in the private subnet looks like this:



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 IaaS 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.

The following document identifies service roles for different node types -- Recommended Cluster Hosts and Role Distribution.

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

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 and coordination services, which may include:

- ResourceManager
- NameNode
- Standby NameNode
- JournalNodes
- ZooKeeper

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 RBS-backed instances.

Ephemeral

When deploying to instances using ephemeral disk for cluster metadata, ensure that the instance types you choose have at least two HDD or SSD, one each dedicated for DFS metadata and ZooKeeper data.

Smaller instances 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 different placement groups[5], such that they are not physically on the same hardware (or equivalent cloud-specific) or deploying to dedicated nodes such that each master node is placed on a separate physical host.

RBS

Per the Cloudera Enterprise Storage Device Acceptance Criteria Guide, the minimum supportable throughput per Worker node (irrespective of whether VM or Bare-metal) is 200 MB/s. This implies that at least 4 Gb/s of East-West network bandwidth is available to the node, for proper performance.

Minimum Per-VM throughput (MB/s)	Minimum per-VM network throughput (Gb/s) (EW)	Recommended per-VM throughput (MB/s)	Recommended per-VM network throughput (Gb/s) (EW)
200	4	800	16

The minimum throughput per Master node is, 120 MB/s. The table below shows the minimum and recommended.

Minimum Per-VM throughput (MB/s)	Minimum per-VM network throughput (Gb/s) (EW)	Recommended per-VM throughput (MB/s)	Recommended per-VM network throughput (Gb/s) (EW)
120	2	240	4

These parameters than can be utilized to build the infrastructure that supports VMs that provide these minimum characteristics. For more details, refer to the Cloudera Enterprise Storage Device Acceptance Criteria Guide.

Utility Nodes

Utility nodes for a Cloudera Enterprise deployment run management, coordination, and utility services, which may include:

- Cloudera Manager
- JournalNode
- ZooKeeper
- Oozie
- Hive Server
- Impala Catalog Server
- Impala State Store
- Job History Server
- Cloudera Management Services

[5] A Placement Group concept is essentially a grouping of instances with a specific objective in mind - provide low latency networking, or distinctly separate hardware.

Refer to Master node requirements.

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 RBS storage, so there are a variety of instances that can be utilized for Worker nodes.

Basic Instance Definitions

Instance Name/Type	vCPUs	Memory	Root Disk	Additional Storage
cdh-worker	>=8 or as sized[6]	>= 32GB or As sized	400GB	N x Storage Volumes
cdh-master	16	>= 64GB	400GB	N x Master Volumes

NOTE:

- The Cloudera Hardware Requirements Guide provides a good starting point for ascertaining minimum dimensions of these instances.

Ephemeral

[6] "As Sized" here is predicated on a sizing exercise undertaken, or at least based on the recommendations provided in the Cloudera Hardware Requirements Guide.

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.

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 the memory and disk requirements of Impala, we recommend large instances with sufficient RAM and vCPUs.

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.

Regions and Availability Zones

Regions are self-contained geographical locations where the Cloud 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 the cloud service select for you.

Cloudera EDH deployments are restricted to single regions. Single clusters spanning regions are not supported.

Networking, Connectivity, and Security

Enhanced Networking

Most cloud service providers provide enhanced networking capacities on certain instance types, resulting in higher performance, lower latency, and lower jitter. In order to take advantage of enhanced networking, you should launch an OS image tailored for the specific instance type (eg: para-virtualized or with SR-IOV support) in the VPC and install the appropriate driver.

VPC

VPC has several different configuration options, choose one 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 the Cloud Services

For private subnet deployments, connectivity between your cluster and other the Cloud Services in the same region should be configured to make use of VPC endpoints[7]. VPC endpoints allow configurable, secure, and scalable communication without requiring the use of public IP addresses, NAT or Gateway instances.

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 or NAT gateway in the public subnet, allowing access outside the private subnet into the public domain. Cloudera does not recommend using NAT instances or NAT gateways 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 the Cloud 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 or gateway 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 a dedicated network connection. We recommend using a dedicated network connection 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 the Cloud, connecting to IaaS through the Internet is sufficient and a dedicated network connection may not be required.

[7] If the Cloud solution supports VPC endpoints.

Security Groups

Security Groups are analogous to host firewalls. You can define rules for IaaS 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.

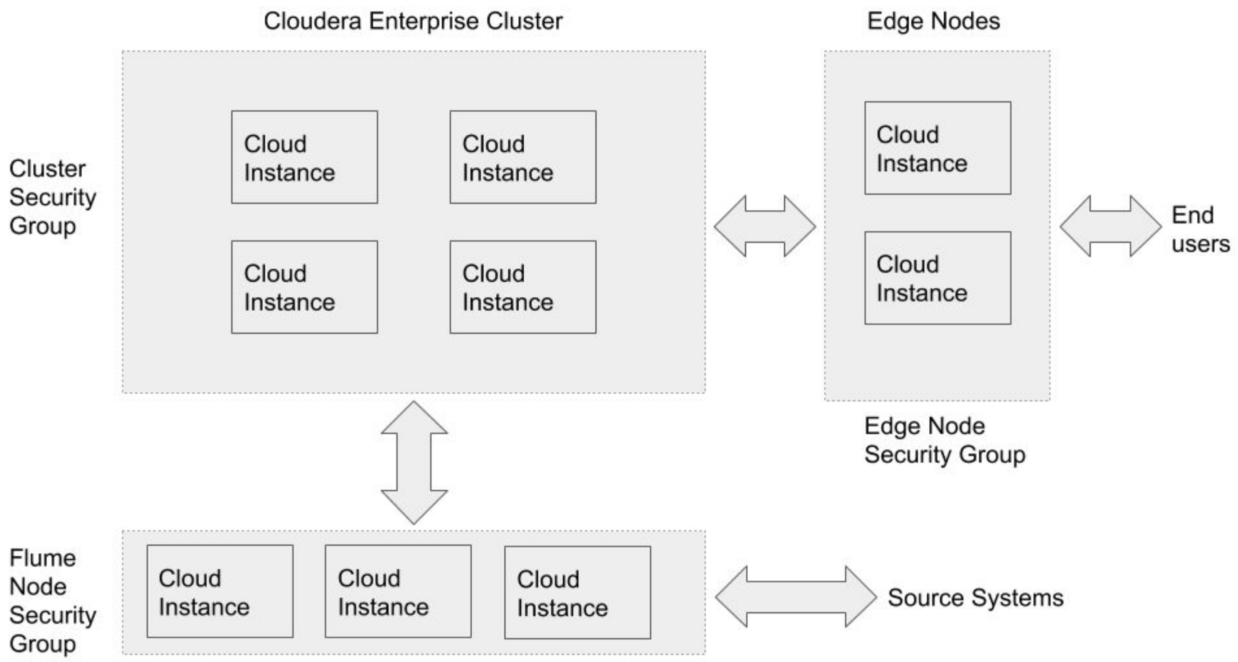
Edge Nodes

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.

Storage Options and Configuration

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

Ephemeral/Instance Storage

IaaS 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 IaaS instance. If you stop or terminate the IaaS instance, the storage is lost. The storage is not lost on restarts, however. Different IaaS 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 data center events. You can set up a scheduled distcp operation to persist data to a cluster or a supported[8] object store (see the examples in the distcp documentation) or leverage the Cloudera Manager Backup and Data Recovery (BDR) features to backup data on another running cluster.

Remote Block Storage (RBS)

Follow the Storage Device Acceptance Criteria Guide for guidance on RBS device selection.

RBS Volume Selection

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

Root Device

We require using RBS volumes as root devices for the IaaS 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 a Cloud platform service 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 IaaS 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 storage volumes and usage.

Installation and Software Configuration

Provisioning Instances

Altus Director can be used to provision IaaS instances on currently supported platforms. Alternately, other tools for orchestration such as Terraform, Chef, SaltStack, Puppet, or Ansible and the Cloudera Manager API (CM API) can be leveraged to simplify the process of provisioning.

To provision IaaS instances manually, first define the VPC configurations based on your requirements for aspects like access to the Internet, other the Cloud Services, and connectivity to your corporate network. Most Cloud platforms have tools that you can leverage to script/automate resource creation and allocation. You can use the respective tools to provision instances. You must create a keypair with which you will later log into the instances.

[8] Currently only Amazon S3 and Azure ADLS are supported

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

- Root device size of at least 400 GB
- Ephemeral storage devices or recommended RBS volumes to be used for master metadata
- Ephemeral storage devices or recommended RBS volumes to be attached to the instances

Along with instances, relational databases must be provisioned. You must set up database instances on IaaS 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 RBS 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 a Cloud provider time service (if available) as a time source.

For more information on operating system preparation and configuration, see Installing Cloudera Manager, CDH, and Managed Services.

Deploying Cloudera Enterprise

If you are using Cloudera Manager, log into the instance that you have elected to host Cloudera Manager and follow the instructions found in Installing Cloudera Manager, CDH, and Managed Services.

If you are using Altus Director[9], see Altus Director installation instructions.

In general, the Cloudera Enterprise Reference Architecture for Bare Metal Deployments provides greater in-depth details of various Cloudera Enterprise components and should be referred to.

Cloudera Enterprise Configuration Considerations

HDFS

Durability

For Cloudera Enterprise deployments in the Cloud, the recommended storage options are ephemeral storage or RBS 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 (`dfs.replication`) at three (3).

DFS block replication can be reduced to two (2) when using RBS-backed[10] data volumes to save on monthly storage costs, but be aware:

[9] Not every Cloud is supported in Altus Director at the time of writing this document.
[10] The reader must validate that the RBS solution has its internal fault tolerance/redundancy mechanism in place (either via replicated blocks or RAID or similar mechanisms).

- read-heavy workloads may take longer to run due to reduced block availability
- reducing replica count effectively migrates durability guarantees from HDFS to RBS
- smaller instances have less network capacity; it will take longer to re-replicate blocks in the event of an RBS volume or IaaS instance failure, meaning longer periods where you're at-risk of losing your last copy of a block

Cloudera does not recommend lowering the replication factor.

Data stored on RBS 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, if applicable for the cloud platform.

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. The Flume 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 RBS. If the IaaS instance goes down, the data on the ephemeral storage is lost. For guaranteed data delivery, use RBS-backed storage for the Flume file channel.

Security Integration

The Cloudera Security guide is intended for system administrators who want to secure a cluster using data encryption, user authentication, and authorization techniques.

It provides conceptual overviews and how-to information about setting up various Hadoop components for optimal security, including how to setup a gateway to restrict access. The guide assumes that you have basic knowledge of Linux and systems administration practices, in general.

Appendix A: Spanning Cloud Availability Zones

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

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

Cloud Provisioning

Provision all IaaS instances in a single VPC/virtual network. 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 Cloud regions. We do not recommend or support spanning clusters across regions. For minimum latency requirements, refer to the Cloudera Enterprise Reference Architecture for Bare Metal Deployments.

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-virginia you would deploy your standby NameNode to us-nyc or us-philly. 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.

Considerations

There are typically data transfer costs[11] associated with IaaS network data sent between AZ.

DFS throughput will be less than if cluster nodes were provisioned within a single AZ.

[11] Contact your Cloud vendor for details.

Network throughput and latency vary based on AZ and IaaS instance size and neither might be guaranteed by the Cloud provider. 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.

Appendix B: Customer Self-test framework

Objective

Objective of this document is to itemize and describe a suite of tests that customers running Cloudera implementations on new platforms such as Multi-cluster, Private cloud platforms and public cloud platforms which don't have specific reference architectures published.

Following tests should be run --

1) Distributed iperf3 to check bandwidth
2) Teragen, Terasort and Teravalidate
3) The Storage Device Acceptance Criteria Guide distributed fio test (for Remote Block Storage validation)

Distributed iperf3 HOWTO

We need the following for this methodology to work --

- iperf3 installed on all nodes
- moreutils installed on all nodes
- A clush setup (clustershell); for example, on a laptop.

The objective is that, given a known set of Iperf3 Server instances, randomly run iperf3 client sessions against them simultaneously from multiple hosts. Towards that end, the same hosts can be set up as servers as well as clients (with the caveat that we avoid running a test from any given node against itself). The clush setup involves creating an entry in the file /etc/clustershell/groups as follows --

```
mc_all:host[0102,0104,0106,0108,0110,0112,0114,0116,0118,0120,0122,0124,0126,0128,01
30,0132,0134,0136,0138,0140].my.company.com
host[0202,0204,0206,0208,0210,0212,0214,0216,0218,0220,0222,0224,0226,0228,0230,0232
,0234,0236,0238,0240].my.company.com host[0302,0304,0305-0333].my.company.com

mc_rack1:host[0102,0104,0106,0108,0110,0112,0114,0116,0118,0120,0122,0124,0126,0128,
0130,0132,0134,0136,0138,0140].my.company.com

mc_rack2:host[0202,0204,0206,0208,0210,0212,0214,0216,0218,0220,0222,0224,0226,0228,
0230,0232,0234,0236,0238,0240].my.company.com
 mc_rack3:host[0302,0304,0305-0333].my.company.com
```

```
# clush -g mc_all 'sudo yum install -y iperf3 moreutils'
```

After this is done, launch iperf3 in Daemon mode on all the hosts as follows --

```
clush -g mc_all -l jdoe sudo iperf3 -sD -p 5001
clush -g mc_all -l jdoe sudo iperf3 -sD -p 5002
clush -g mc_all -l jdoe sudo iperf3 -sD -p 5003
```

Verify that the iperf3 Daemons are running --

```
$ clush -g mc_all -l jdoe 'ps -ef|grep iperf3|grep -v grep'
 host0102.my.company.com: root      10245      1 14 12:50 ?        00:06:42 iperf3
-sD -p 5001
 host0106.my.company.com: root       9495      1 12 12:50 ?        00:05:36 iperf3
-sD -p 5001
```

```
 host0104.my.company.com: root       9554      1  9 12:50 ?        00:04:20 iperf3
-sD -p 5001
 <truncated for readability>
 host0315.my.company.com: root      33247      1  0 12:57 ?        00:00:00 iperf3
-sD -p 5001
 host0224.my.company.com: root      33136      1  0 12:57 ?        00:00:00 iperf3
-sD -p 5001
 host0323.my.company.com: root      33257      1  0 12:57 ?        00:00:00 iperf3
-sD -p 5001
 host0318.my.company.com: root      32868      1  0 12:57 ?        00:00:00 iperf3
-sD -p 5001
 host0236.my.company.com: root      33470      1  0 12:57 ?        00:00:00 iperf3
-sD -p 5001
 host0236.my.company.com: jdoe   33734  33492 22 13:34 ?        00:00:10 iperf3 -c
```

After all the nodes have been setup to run iperf3 server instances in Daemon mode, create multiple nodelist files (in this case, we are testing cross-rack bandwidth, and so set up the nodelist per rack as follows --

```
$ ls -lrt
 total 32
 -rw-r--r--  1 jdoe  staff  806 Aug  9 14:52 nodes.rack3
 -rw-r--r--  1 jdoe  staff  520 Aug  9 14:52 nodes.rack2
 -rw-r--r--  1 jdoe  staff  520 Aug  9 14:52 nodes.rack1
 -rwxr-xr-x  1 jdoe  staff  221 Aug  9 14:52 random_net.sh
```

Distribute the files to all nodes --

```
$ clush -g mc_all -c nodes.rack*  --dest=/home/jdoe/
$ clush -g mc_all -c random*.sh --dest=/home/jdoe/
```

Each file contains a list of all nodes in the specific rack. Eg:

```
$ cat nodes.rack1
 host0102.my.company.com
 host0104.my.company.com
 host0106.my.company.com
 host0108.my.company.com
 host0110.my.company.com
 host0112.my.company.com
 host0114.my.company.com
 host0116.my.company.com
 host0118.my.company.com
 host0120.my.company.com
 host0122.my.company.com
 host0124.my.company.com
 host0126.my.company.com
 host0128.my.company.com
 host0130.my.company.com
 host0132.my.company.com
 host0134.my.company.com
 host0136.my.company.com
 host0138.my.company.com
 host0140.my.company.com
```

The shell script random_net.sh is intended to randomly select a target node against which to run the iperf test. This is to be run on the client side, say from Rack2 (client) to Rack1 (server) --

```
#!/bin/sh

# Read nodelist into an array
IFS=$'\r\n' GLOBIGNORE='*' command eval 'nodes=($(cat $1))'

 ports=(5001 5002 5003)

 psize=${#ports[@]}
size=${#nodes[@]}

 for i in $(seq $size)
do
       index=$(( $RANDOM % size ))
       pindex=$(( $RANDOM % psize ))
       target=${nodes[$index]}
       ptarget=${ports[$pindex]}
       iperf3 -c $target -p $ptarget |ts |tee -a ~/iperf3.log
       # Run payload from Client to server
       iperf3 -c $target -R -p $ptarget |ts |tee -a ~/iperf3.log  # Reverse the
direction
done
```

In order to run against a single iperf3 server instance per node, run this script instead

```
#!/bin/sh

 # Read nodelist into an array
IFS=$'\r\n' GLOBIGNORE='*' command eval 'nodes=($(cat $1))'

size=${#nodes[@]}

 for i in $(seq $size)
do
       index=$(( $RANDOM % size ))
       target=${nodes[$index]}
       iperf3 -c $target -p 5001 |ts |tee -a ~/${target}_iperf3.log    # Run payload
from Client to server
       iperf3 -c $target -R -p 5001|ts |tee -a  ~/${target}_iperf3.log  # Reverse the
direction
done
```

Run the script(s) as follows –

```
$ clush -g mc_rack2 -l jdoe 'sh random_net.sh nodes.rack1'
```

Or

```
$ clush -g mc_rack2 -l jdoe 'sh randomnet_single.sh nodes.rack1'
```

This example shows the test running in parallel on all nodes of Rack2 by randomly selecting a node in Rack1 as the target.

```
$ clush -g mc_rack2 -l jdoe 'sh random_net.sh nodes.rack1'
 host0224.my.company.com: Aug 09 14:31:08 iperf3: error - the server is busy running
a test. try again later
 host0240.my.company.com: Aug 09 14:31:08 Connecting to host
host0110.my.company.com, port 5002
 host0240.my.company.com: Aug 09 14:31:08 Reverse mode, remote host
host0110.my.company.com is sending
 host0240.my.company.com: Aug 09 14:31:08 [  4] local 192.168.1.70 port 42042
connected to 192.168.1.15 port 5002
 host0240.my.company.com: Aug 09 14:31:08 iperf3: error - control socket has closed
unexpectedly
 host0210.my.company.com: Aug 09 14:31:08 iperf3: error - the server is busy running
a test. try again later
 host0218.my.company.com: Aug 09 14:31:17 Connecting to host
host0108.my.company.com, port 5001
```

```
 host0218.my.company.com: Aug 09 14:31:17 [  4] local 192.168.1.59 port 54962
connected to 192.168.1.14 port 5001
 host0218.my.company.com: Aug 09 14:31:17 [ ID] Interval           Transfer
Bandwidth       Retr  Cwnd
 host0218.my.company.com: Aug 09 14:31:17 [  4]   0.00-1.00   sec   646 MBytes  5.42
Gbits/sec   80    595 KBytes
 host0218.my.company.com: Aug 09 14:31:17 [  4]   1.00-2.00   sec   542 MBytes  4.55
Gbits/sec   50    561 KBytes
 host0218.my.company.com: Aug 09 14:31:17 [  4]   2.00-3.00   sec   574 MBytes  4.81
Gbits/sec   29    577 KBytes
```

Teragen/Terasort/Teravalidate

To use these benchmarking tools follow the instructions provided in the Cloudera Enterprise Reference Architecture for Bare Metal Deployments document.

Storage Acceptance Criteria Guide microbenchmarks

The Cloudera Enterprise Storage Device Acceptance Criteria Guide provides a validation mechanism that should be followed, along with running the microbenchmarks referenced.

References

Cloudera Enterprise

Cloudera

Cloudera Product Documentation

Cloudera Services & Support

Cloudera Enterprise Storage Device Acceptance Criteria Guide

Cloudera Enterprise Reference Architecture for Bare Metal Deployments

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and
2. You must cause any modified files to carry prominent notices stating that You changed the files; and
3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

```
  Copyright [yyyy] [name of copyright owner]

   Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
   You may obtain a copy of the License at

     http://www.apache.org/licenses/LICENSE-2.0

   Unless required by applicable law or agreed to in writing,
software
   distributed under the License is distributed on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied.
   See the License for the specific language governing permissions
and
   limitations under the License.
```