



CLOUDERA AND ISILON BEST PRACTICES AND CONFIGURATION GUIDE

Best Practices for configuring and tuning Cloudera

November 2015

To learn more about how EMC products, services, and solutions can help solve your business and IT challenges, [contact](#) your local representative or authorized reseller, visit www.emc.com, or explore and compare products in the [EMC Store](#)

Copyright © 2014 EMC Corporation. All Rights Reserved.

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided “as is.” EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

For the most up-to-date listing of EMC product names, see EMC Corporation Trademarks on EMC.com.

Part Number HXXXXX <required, see Part numbers below for more info>

TABLE OF CONTENTS

INTRODUCTION.....	5
AUDIENCE	5
APACHE HADOOP PROJECTS.....	5
CLOUDERA CDH AND CLOUDERA MANAGER	6
ISILON SCALE-OUT NAS FOR HDFS.....	6
OVERVIEW OF ISILON SCALE-OUT NAS FOR BIG DATA	6
CLOUDERA AND EMC JOINT SUPPORT STATEMENT	6
ENVIRONMENT	7
Versions	7
HOSTS	8
INSTALLATION OVERVIEW.....	9
PREREQUISITES	9
Isilon	9
Networking	9
DNS.....	10
Other	10
PREPARE ISILON.....	10
Assumptions	10
SmartConnect For HDFS	10
OneFS Access Zones	11
Sharing Data Between Access Zones	12
User and Group IDs	12
Configure Isilon for HDFS.....	13
Create DNS Records For Isilon.....	15
Prepare NFS Clients	17
Tune Isilon for Optimal Performance	18
INSTALL CLOUDERA MANAGER.....	18
Deploy a Cloudera Hadoop Cluster	18

ADDING A HADOOP USER	22
FUNCTIONAL TESTS.....	23
HDFS	23
YARN / MapReduce	23
Hive.....	24
Pig.....	26
HBase	26
Impala	26
SEARCHING WIKIPEDIA	27
WHERE TO GO FROM HERE	29
KNOWN LIMITATIONS.....	29
REFERENCES	29

Introduction

IDC published an update to their Digital Universe study in December 2012 and found that the rate of digital data creation is not only continuing to grow, but the rate is actually accelerating. By the end of this decade we will create 40 Zettabytes of new digital information yearly or the equivalent of 1.7MB of digital information for every man, woman, and child every second of every day.

This information explosion is creating new opportunities for our businesses to leverage digital information to serve their customers better, faster, and most cost effectively through Big Data Analytics applications. Hadoop technologies can be cost effective solutions and can manage structured, semi-structured and unstructured data unlike traditional solutions such as RDBMS. The need to track and analyze consumer behavior, maintain inventory and space, target marketing offers on the basis of consumer preferences and attract and retain consumers, are some of the factors pushing the demand for Big Data Analytics solutions using Hadoop technologies. According to a new market report published by Transparency Market Research (<http://www.transparencymarketresearch.com>) "Hadoop Market - Global Industry Analysis, Size, Share, Growth, Trends, and Forecast, 2012- 2018," the global Hadoop market was worth USD 1.5 billion in 2012 and is expected to reach USD 20.9 billion in 2018, growing at a CAGR of 54.7% from 2012 to 2018.

Hadoop like any new technology can be time consuming, and expensive for our customers to get deployed and operational. When we surveyed a number of our customers, two main challenges were identified to getting started: confusion over which Hadoop distribution to use and how to deploy using existing IT assets and knowledge. Hadoop software is distributed by several vendors including Pivotal, Hortonworks, and Cloudera with proprietary extensions. In addition to these distributions, Apache distributes a free open source version. From an infrastructure perspective many Hadoop deployments start outside the IT data center and do not leverage the existing IT automation, storage efficiency, and protection capabilities. Many customers cited the time it took IT to deploy Hadoop as the primary reason to start with a deployment outside of IT.

This guide is intended to simplify Hadoop deployments by creating a shared storage model with EMC Isilon scale out NAS and using an industry leader in Enterprise Hadoop, Cloudera, reduce the time to deployment, and the cost of deployment leveraging tools that can automate Hadoop cluster deployments.

Audience

This document is intended for IT program managers, IT architects, Developers, and IT management to easily deploy Cloudera Hadoop with automation tools and leverage EMC Isilon for HDFS shared storage. It can be used by somebody who does not yet have an EMC Isilon cluster by downloading the free EMC Isilon OneFS Simulator which can be installed as a virtual machine for testing and training. However, this document can also be used by somebody who will be installing in a production environment as best-practices are followed whenever possible.

Apache Hadoop Projects

Apache Hadoop is an open source, batch data processing system for enormous amounts of data. Hadoop runs as a platform that provides cost-effective, scalable infrastructure for building Big Data analytic applications. All Hadoop clusters contain a distributed file system called the Hadoop Distributed File System (HDFS), a computation layer called MapReduce.

The Apache Hadoop project contains the following subprojects:

Hadoop Distributed File System (HDFS) – A distributed file system that provides high-throughput access to application data.

Hadoop MapReduce – A software framework for writing applications to reliably process large amounts of data in parallel across a cluster.

Hadoop is supplemented by an ecosystem of Apache projects, such as Pig, Hive, Sqoop, Flume, Oozie, Whirr, HBase, and Zookeeper that extend the value of Hadoop and improves its usability.

Version 2 of Apache Hadoop introduces YARN, a sub-project of Hadoop that separates the resource management and processing components. YARN was born of a need to enable a broader array of interaction patterns for data stored in HDFS beyond MapReduce. The YARN-based architecture of Hadoop 2.0 provides a more general processing platform that is not constrained to MapReduce.

For full details of the Apache Hadoop project see <http://hadoop.apache.org/>.

Cloudera CDH and Cloudera Manager

CDH (Cloudera's Distribution Including Apache Hadoop) is the world's most complete, tested, and widely deployed distribution of Apache Hadoop. CDH is 100% open source and is the only Hadoop solution to offer batch processing, interactive SQL, and interactive search as well as enterprise-grade continuous availability. More enterprises have downloaded CDH than all other distributions combined.

CDH delivers the core elements of Hadoop – scalable storage and distributed computing – as well as all of the necessary enterprise capabilities such as security, high availability and integration with a broad range of hardware and software solutions

Cloudera Manager is the industry's first and most sophisticated management application for Apache Hadoop. Cloudera Manager sets the standard for enterprise deployment by delivering granular visibility into and control over every part of the Hadoop cluster — empowering operators to improve performance, enhance quality of service, increase compliance and reduce administrative costs.

Cloudera Manager is designed to make administration of Hadoop simple and straightforward, at any scale. With Cloudera Manager, you can easily deploy and centrally operate the complete Hadoop stack. The application automates the installation process, reducing deployment time from weeks to minutes; gives you a cluster-wide, real-time view of nodes and services running; provides a single, central console to enact configuration changes across your cluster; and incorporates a full range of reporting and diagnostic tools to help you optimize performance and utilization.

More information on Cloudera can be found on <http://www.cloudera.com/>.

Isilon Scale-Out NAS For HDFS

EMC Isilon is the only scale-out NAS platform natively integrated with the Hadoop Distributed File System (HDFS). Using HDFS as an over-the-wire protocol, you can deploy a powerful, efficient, and flexible data storage and analytics ecosystem.

In addition to native integration with HDFS, EMC Isilon storage easily scales to support massively large Hadoop analytics projects. Isilon scale-out NAS also offers unmatched simplicity, efficiency, flexibility, and reliability that you need to maximize the value of your Hadoop data storage and analytics workflow investment.

Overview of Isilon Scale-Out NAS for Big Data

The EMC Isilon scale-out platform combines modular hardware with unified software to provide the storage foundation for data analysis. Isilon scale-out NAS is a fully distributed system that consists of nodes of modular hardware arranged in a cluster. The distributed Isilon OneFS operating system combines the memory, I/O, CPUs, and disks of the nodes into a cohesive storage unit to present a global namespace as a single file system.

The nodes work together as peers in a shared-nothing hardware architecture with no single point of failure. Every node adds capacity, performance, and resiliency to the cluster, and each node acts as a Hadoop namenode and datanode. The namenode daemon is a distributed process that runs on all the nodes in the cluster. A compute client can connect to any node through HDFS.

As nodes are added, the file system expands dynamically and redistributes data, eliminating the work of partitioning disks and creating volumes. The result is a highly efficient and resilient storage architecture that brings all the advantages of an enterprise scale-out NAS system to storing data for analysis.

Unlike traditional storage, Hadoop's ratio of CPU, RAM, and disk space depends on the workload—factors that make it difficult to size a Hadoop cluster before you have had a chance to measure your MapReduce workload. Expanding data sets also makes sizing decisions upfront problematic. Isilon scale-out NAS lends itself perfectly to this scenario: Isilon scale-out NAS lets you increase CPUs, RAM, and disk space by adding nodes to dynamically match storage capacity and performance with the demands of a dynamic Hadoop workload.

An Isilon cluster optimizes data protection. OneFS more efficiently and reliably protects data than HDFS. The HDFS protocol, by default, replicates a block of data three times. In contrast, OneFS stripes the data across the cluster and protects the data with forward error correction codes, which consume less space than replication with better protection.

Cloudera and EMC Joint Support Statement

EMC Isilon and Cloudera are pleased to communicate a business collaboration and intention to enable joint support for EMC Isilon scale-out NAS and Cloudera Enterprise products to bring the value of Cloudera Enterprise to customers using EMC Isilon storage.

EMC Isilon Scale-Out NAS storage for Hadoop currently supports the Apache Hadoop Distributed File System (HDFS) protocol. CDH is 100% open source and is the only Hadoop solution to offer batch processing, interactive SQL and interactive search, as well as enterprise-grade continuous availability.

EMC Isilon allows a customer to start using Hadoop now by using already existing data thus eliminating extra copies and reducing associated CAPEX costs for additional storage capacity. In addition, EMC Isilon is the only Hadoop storage solution that allows you access to the data via NAS (i.e. SMB) or HDFS protocols as well as providing a POSIX-Compliant file system for regulated environments.

EMC Isilon and Cloudera are now jointly working to support the following two scenarios:

Customers running Cloudera Enterprise on EMC Isilon NAS products.

Customers will be able to leverage the full Cloudera Enterprise offering on their existing data sets stored in EMC Isilon. Cloudera Enterprise is a subscription offering that combines CDH with Cloudera Manager for system management, Cloudera Navigator for data management, technical support, indemnity, and open source advocacy. Customers will be able to simplify storage management and reduce overall costs by managing storage and compute independently, with new server hardware purchases required only for additional compute in the case of an existing EMC Isilon installation.

Customers wishing to integrate existing Cloudera Enterprise and EMC Isilon clusters.

Customers will be able to integrate existing EMC Isilon clusters with existing HDFS-storage based Cloudera Enterprise clusters by using existing Hadoop tools built for data movement. This scenario will allow customers to more easily ingest data into both systems, as well as enable use cases such as online or remote backup and disaster recovery.

EMC Isilon and Cloudera are currently working on product development and support models, and intend to have a supported joint offering in the market in the first half of 2014.

In addition, EMC Isilon and Cloudera are working together to advance the ongoing joint roadmap for Cloudera Enterprise on EMC Isilon Scale-Out NAS for subsequent releases of CDH, Cloudera manager, Cloudera Navigator and OneFS software.

This guide illustrates the first scenario. Using an existing Isilon cluster to integrate with Cloudera manager to allow CDH access to existing data sets located in the Isilon cluster.

Environment

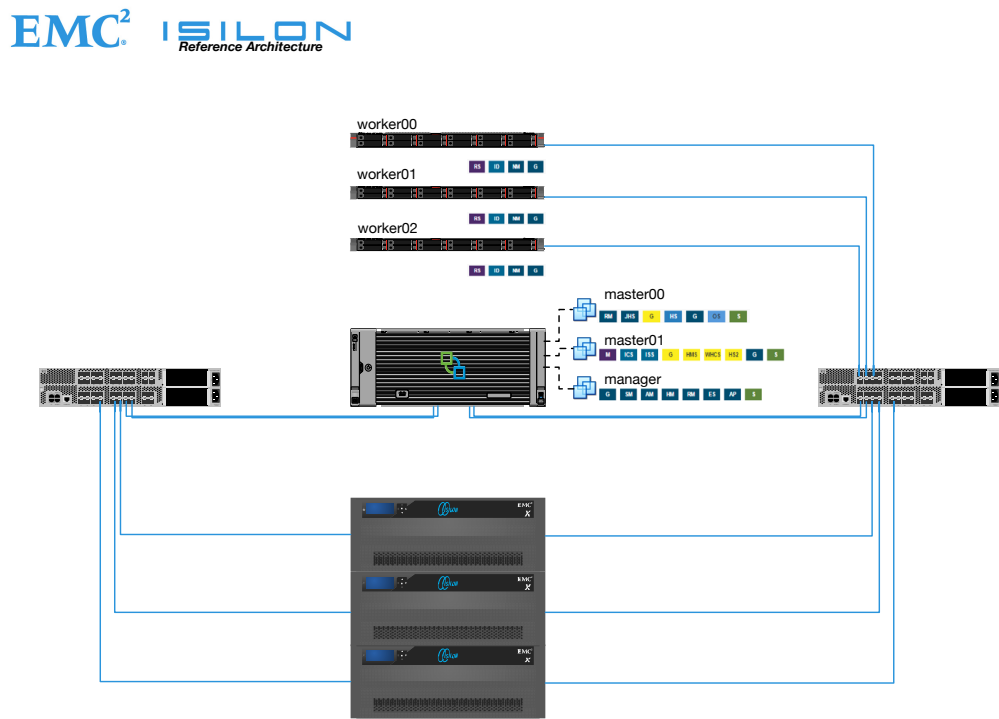
Versions

The test environments used for this document consist of the following software versions:

- Cloudera Manager 5.4
- Cloudera CDH 5.4.7
- Isilon OneFS 7.2.0.4

Hosts

A typical Hadoop environment composed of many types of hosts. Below is a description of these hosts.



cloudera

List of Hosts in Hadoop Environments

Host (Host Name)	Description
DNS	It is recommended to have a DNS configured for the environment
Linux workstation	You should have a Linux workstation with a GUI that you can use to control everything with.
Isilon nodes	You will need one or more Isilon Scale-Out NAS nodes. For functional testing, you can use a single Isilon OneFS Simulator node instead of the Isilon appliance. The nodes will be clustered together into an <i>Isilon Cluster</i> . Isilon nodes run the OneFS operating system.
Isilon InsightIQ	This is optional licensed software from Isilon and can be used to monitor the health and performance of your Isilon cluster. It is recommend for any performance testing.
Cloudera Manager	This host will manage your Hadoop cluster. This will run Cloudera Manager which will deploy and monitor the Hadoop components. In smaller environments, you may choose to deploy Cloudera Manager to your Hadoop Master host.
Hadoop Master	This host will the YARN Resource Manager, Job History Server, Hive Metastore Server, etc.. In general, it will run all "master" services except for the HDFS Name Node.
Hadoop Worker Node	There will be any number of worker nodes, depending on the compute requirements. Each node will run the YARN Node Manager, HBase Region Server, etc.. During the installation process, these nodes will run the HDFS Data Node but these will become idle like the HDFS Name Node.

Due to the many hosts involved, all commands that must be typed are prefixed with the standard system prompt identifying the user and host name. For example:

```
[user@workstation ~]$ ping myhost.lab.example.com
```


Installation Overview

Below is the overview of the installation process that this document will describe. If you have an existing Cloudera environment managed by Cloudera Manager and you wish to integrate EMC Isilon for HDFS then you can skip to the section titled "Connect Cloudera Hadoop to Isilon" (step 8 below).

1. Confirm prerequisites.
2. Prepare your network infrastructure including DNS.
3. Prepare your Isilon cluster.
4. Install Cloudera Manager.
5. Use Cloudera Manager to deploy Cloudera CDH to the virtual machines.
6. Perform key functional tests.

Prerequisites

Isilon

- For low-capacity, non-performance testing of Isilon, the EMC Isilon OneFS Simulator can be used instead of a cluster of physical Isilon appliances. This can be downloaded for free from <http://www.emc.com/getisilon>. Refer to the *EMC Isilon OneFS Simulator Install Guide* for details. Be sure to follow the section for running the virtual nodes in VMware ESX. Only a single virtual node is required but adding additional nodes will allow you to explore other features such as data protection, SmartPools (tiering), and SmartConnect (network load balancing).
- For physical Isilon nodes, you should have already completed the console-based installation process for your first Isilon node and added at least two other nodes for a minimum of 3 Isilon nodes.
- You must have OneFS version 7.1.1.0 with patch-130611 or version 7.2.0.0 and higher.
- You must obtain a OneFS HDFS license code and install it on your Isilon cluster. You can get your free OneFS HDFS license from <http://www.emc.com/campaign/isilon-hadoop/index.htm>.
- It is recommended, but not required, to have a SmartConnect Advanced license for your Isilon cluster.
- To allow for scripts and other small files to be easily shared between all nodes in your environment, it is highly recommended to enable NFS (Unix Sharing) on your Isilon cluster. By default, the entire /ifs directory is already exported and this can remain unchanged. This document assumes that a single Isilon cluster is used for this NFS export as well as for HDFS. However, there is no requirement that the NFS export be on the same Isilon cluster that you are using for HDFS.

Networking

- For the best performance, a single 10 Gigabit Ethernet switch should connect to at least one 10 Gigabit port on each host running a worker. Additionally, the same switch should connect to at least one 10 Gigabit port on each Isilon node.
- A single dedicated layer-2 network can be used to connect all hosts and Isilon nodes. Although multiple networks can be used for increased security, monitoring, and robustness, it adds complications that should be avoided when possible.
- At least one IP per hadoop node
- At a minimum, you will need to allocate to your Isilon cluster one IP address per Access Zone per Isilon node. In general, you will need one Access Zone for each separate Hadoop cluster that will use Isilon for HDFS storage. For the best possible load balancing during an Isilon node failure scenario, the recommended number of IP addresses is given by the formula below. Of course, this is in addition to any IP addresses used for non-HDFS pools. $\# \text{ of IP addresses} = 2 * (\# \text{ of Isilon Nodes}) * (\# \text{ of Access Zones})$ For example, 20 IP addresses are recommended for 5 Isilon nodes and 2 Access Zones.

- This document will assume that Internet access is available to all servers to download various components from Internet repositories.

DNS

- A DNS server is required and you must have the ability to create DNS records and zone delegations.
- It is recommended that your DNS server delegate a subdomain to your Isilon cluster. For instance, DNS requests for subnet0-pool0.isiloncluster1.lab.example.com or isiloncluster1.lab.example.com should be delegated to the Service IP defined on your Isilon cluster.
- To allow for a convenient way of changing the HDFS Name Node used by all Hadoop applications and services, it is recommended to create a DNS record for your Isilon cluster's HDFS Name Node service. This should be a CNAME alias to your Isilon SmartConnect zone. Specify a TTL of 1 minute to allow for quick changes while you are experimenting. For example, create a CNAME record for mycluster1-hdfs.lab.example.com that targets subnet0-pool0.isiloncluster1.lab.example.com. If you later want to redirect all HDFS I/O to another cluster or a different pool on the same Isilon cluster, you simply need to change the DNS record and restart all Hadoop services.

Other

- You will need one Linux workstation which you will use to perform most configuration tasks. No services will run on this workstation.
 - This should have a GUI and a web browser.
 - This must have Python 2.6.6 or higher 2.x version.
- CentOS 6.7 has been used for testing but most other systems should also work. Be aware that Centos 6.4 must be upgraded to support Python 2.6.6.
- sshpass should be installed to make managing the hadoop cluster elements easier.
- Several useful scripts and file templates are provided in the archive file isilon-hadoop-tools-x.x.tar.gz. Download the latest version from <https://github.com/claudiofahey/isilon-hadoop-tools/releases>.
- Time synchronization is critical for Hadoop. It is highly recommended to configure all hosts and Isilon nodes to use NTP. In general, you do not need to run NTP clients if using VMs.

Prepare Isilon

Assumptions

This document makes the assumptions listed below. These are not necessarily requirements but they are usually valid and simplify the process.

- It is assumed that you are not using a directory service such as Active Directory for Hadoop users and groups.
- It is assumed that you are not using Kerberos authentication for Hadoop.

SmartConnect For HDFS

The best practice for HDFS on Isilon is to utilize one SmartConnect IP address pool for each hadoop access zone. One IP address pool should be used by Hadoop clients to connect to the HDFS Name Node service on Isilon and it should use the dynamic IP allocation method to minimize connection interruptions in the event that an Isilon node fails. **Note that dynamic IP allocation requires a SmartConnect Advanced license.** A Hadoop client uses a specific SmartConnect IP address pool simply by using its zone name (DNS name) in the HDFS URI (e.g. hdfs://subnet0-pool1.isiloncluster1.lab.example.com:8020).

The same pool used for HDFS Name Node connections should also be used for HDFS data node connections. To assign specific SmartConnect IP address pools for data node connections, you will use the "isi hdfs racks modify" command.

If you do not have a SmartConnect Advanced license, you may choose to use a single static pool for name node and data node connections. This may result in some failed HDFS connections immediately after Isilon node failures.

For more information, see [EMC Isilon Best Practices for Hadoop Data Storage](#).

OneFS Access Zones

Access zones on OneFS are a way to select a distinct configuration for the OneFS cluster based on the IP address that the client connects to. For HDFS, this configuration includes authentication methods, HDFS root path, and authentication providers (AD, LDAP, local, etc.). By default, OneFS includes a single access zone called System.

If you will only have a single Hadoop cluster connecting to your Isilon cluster, then you can use the System access zone with no additional configuration. However, to have more than one Hadoop cluster connect to your Isilon cluster, it is best to have each Hadoop cluster connect to a separate OneFS access zone. This will allow OneFS to present each Hadoop cluster with its own HDFS namespace and an independent set of users.

For more information, see [Security and Compliance for Scale-out Hadoop Data Lakes](#).

To view your current list of access zones and the IP pools associated with them:

```
isiloncluster1-1# isi zone zones list
```

```
Name    Path
-----
System  /ifs
-----
Total: 1
```

```
isiloncluster1-1# isi networks list pools -v
```

```
subnet0:pool0
    In Subnet: subnet0
    Allocation: Static
    Ranges: 1
        10.111.129.115-10.111.129.126
    Pool Membership: 4
        1:10gige-1 (up)
        2:10gige-1 (up)
        3:10gige-1 (up)
        4:10gige-1 (up)
    Aggregation Mode: Link Aggregation Control Protocol (LACP)
    Access Zone: System (1)
    SmartConnect:
        Suspended Nodes : None
        Auto Unsuspend ... 0
        Zone             : subnet0-pool0.isiloncluster1.lab.example.com
        Time to Live      : 0
        Service Subnet    : subnet0
        Connection Policy : Round Robin
        Failover Policy   : Round Robin
        Rebalance Policy  : Automatic Failback
```

To create a new access zone and an associated IP address pool:

```
isiloncluster1-1# mkdir -p /ifs/isiloncluster1/zone1
isiloncluster1-1# isi zone zones create --name zone1 \
--path /ifs/isiloncluster1/zone1
isiloncluster1-1# isi networks create pool --name subnet0:pool1 \
--ranges 10.111.129.127-10.111.129.138 --ifaces 1-4:10gige-1 \
--access-zone zone1 --zone subnet0-pool1.isiloncluster1.lab.example.com \
--sc-subnet subnet0 --dynamic
```

Creating pool

```
'subnet0:pool1': OK
```

Saving:

OK

If you do not have a SmartConnect Advanced license, you will need to omit the `--dynamic` option.

To allow the new IP address pool to be used by data node connections:

```
isiloncluster1-1# isi hdfs racks create /rack0 --client-ip-ranges \
0.0.0.0-255.255.255.255
isiloncluster1-1# isi hdfs racks modify /rack0 --add-ip-pools subnet0:pool1
isiloncluster1-1# isi hdfs racks list
```

Name	Client IP Ranges	IP Pools
/rack0	0.0.0.0-255.255.255.255	subnet0:pool1

Total: 1

Sharing Data Between Access Zones

Access zones in OneFS provide a measure of multi-tenancy in that data within one access zone cannot be accessed by another access zone. In certain use cases, however, you may actually want to make the same dataset available to more than one Hadoop cluster. This can be done by using fully-qualified paths to refer to data in other access zones.

To use this approach, you will configure your Hadoop jobs to simply access the datasets from a common shared HDFS namespace. For instance, you would start with two independent Hadoop clusters, each with its own access zone on Isilon. Then you can add a 3rd access zone on Isilon, with its own IP addresses and HDFS root, and containing a dataset that is shared with other Hadoop clusters.

User and Group IDs

Isilon clusters and Hadoop servers each have their own mapping of user IDs (uid) to user names and group IDs (gid) to group names. When Isilon is used only for HDFS storage by the Hadoop servers, the IDs do not need to match. This is due to the fact that the HDFS wire protocol only refers to users and groups by their *names*, and never their numeric IDs.

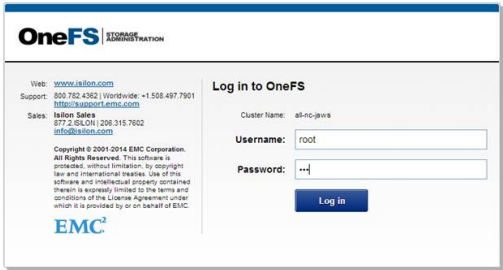
In contrast, the NFS wire protocol refers to users and groups by their numeric IDs. Although NFS is rarely used in traditional Hadoop environments, the high-performance, enterprise-class, and POSIX-compatible NFS functionality of Isilon makes NFS a compelling protocol for certain workflows. If you expect to use both NFS and HDFS on your Isilon cluster (or simply want to be open to the possibility in the future), it is highly recommended to maintain consistent names and numeric IDs for all users and groups on Isilon and your Hadoop servers. In a multi-tenant environment with multiple Hadoop clusters, numeric IDs for users in different clusters should be distinct.

For instance, the user sqoop in Hadoop cluster A will have ID 610 and this same ID will be used in the Isilon access zone for Hadoop cluster A as well as every server in Hadoop cluster A. The user sqoop in Hadoop cluster B will have ID 710 and this ID will be used in the Isilon access zone for Hadoop cluster B as well as every server in Hadoop cluster B.

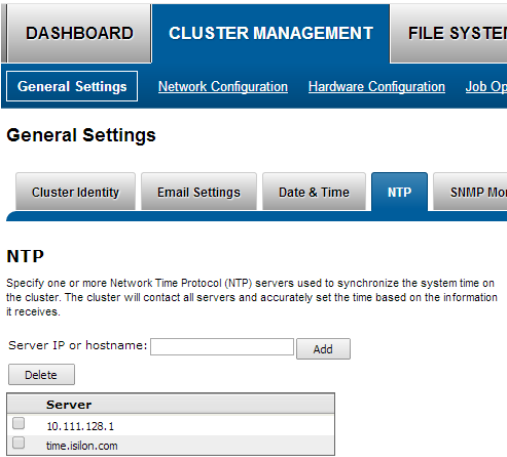
Configure Isilon for HDFS

In the steps below, replace *zone1* with **System** to use the default System access zone or you may specify the name of a new access zone that you previously created.

1. Open a web browser to the your Isilon cluster's web administration page. If you don't know the URL, simply point your browser to `https://isilon_node_ip_address:8080`, where *isilon_node_ip_address* is any IP address on any Isilon node that is in the System access zone. This usually corresponds to the ext-1 interface of any Isilon node.
2. Login with your root account. You specified the root password when you configured your first node using the console.



3. Check, and edit as necessary, your NTP settings. Click Cluster Management -> General Settings -> NTP.



4. SSH into any node in your Isilon cluster as root.
5. Confirm that your Isilon cluster is at OneFS version 7.1.1.0 or higher.

```
isiloncluster1-l# isi version
Isilon OneFS v7.1.1.0 ...
```

6. Verify your HDFS license.

```
isiloncluster1-l# isi license
```

Module	License Status	Configuration	Expiration Date
-----	-----	-----	-----

7. Create the HDFS root directory. This is usually called *hadoop* and must be within the access zone directory.

```
isiloncluster1-l# mkdir -p /ifs/isiloncluster1/zone1/hadoop
```

8. Set the HDFS root directory for the access zone.

```
isiloncluster1-l# isi zone zones modify zone1 \
--hdfs-root-directory /ifs/isiloncluster1/zone1/hadoop
```

9. Increase the HDFS daemon thread count.

```
isiloncluster1-l# isi hdfs settings modify --server-threads 256
```

10. Set the HDFS block size used for reading from Isilon.

```
isiloncluster1-l# isi hdfs settings modify --default-block-size 128M
```

11. Create an indicator file so that we can easily determine when we are looking your Isilon cluster via HDFS.

```
isiloncluster1-l# touch \
/ifs/isiloncluster1/zone1/hadoop/THIS_IS_ISILON_isiloncluster1_zone1
```

12. Extract the Isilon Hadoop Tools to your Isilon cluster. This can be placed in any directory under /ifs. It is recommended to use /ifs/isiloncluster1/scripts where *isiloncluster1* is the name of your Isilon cluster.

```
[user@workstation ~]$ scp isilon-hadoop-tools-x.x.tar.gz \
root@isilon_node_ip_address:/ifs/isiloncluster1/scripts
isiloncluster1-l# tar -xzf \
/ifs/isiloncluster1/isilon-hadoop-tools-x.x.tar.gz \
-C /ifs/isiloncluster1/scripts
isiloncluster1-l# mv /ifs/isiloncluster1/scripts/isilon-hadoop-tools-x.x \
/ifs/isiloncluster1/scripts/isilon-hadoop-tools
```

13. Execute the script `isilon_create_users.sh`. This script will create all required users and groups for the Hadoop services and applications. The script `isilon_create_users.sh` will create local user and group accounts on your Isilon cluster for Hadoop services. If you are using a directory service such as Active Directory, and you want these users and groups to be defined in your directory service, then DO NOT run this script. Instead, refer to the OneFS documentation and [EMC Isilon Best Practices for Hadoop Data Storage](#). Script Usage: `isilon_create_users.sh --dist <DIST> [--startgid <GID>] [--startuid <UID>] [--zone <ZONE>]` **dist** This will correspond to your Hadoop distribution - `cdh` **startgid** Group IDs will begin with this value. For example: 501 **startuid** User IDs will begin with this value. This is generally the same as `gid_base`. For example: 501 **zone** Access Zone name. For example: `System`

```
isiloncluster1-l# bash \
/ifs/isiloncluster1/scripts/isilon-hadoop-tools/onefs/isilon_create_users.sh \
--dist cdh --startgid 501 --startuid 501 --zone zone1
```

14. Execute the script `isilon_create_directories.sh`. This script will create all required directories with the appropriate ownership and permissions. Script Usage: `isilon_create_directories.sh --dist <DIST> [--fixperm] [--zone <ZONE>] dist` This will correspond to your Hadoop distribution - `cdh` **fixperm** If specified, ownership and permissions will be set on existing directories. **zone** Access Zone name. For example: System

```
isiloncluster1-l# bash \  
/ifs/isiloncluster1/scripts/isilon-hadoop-tools/onefs/isilon_create_directories.sh \  
--dist cdh --fixperm --zone zone1
```

15. Map the `hdfs` user to the Isilon superuser. This will allow the `hdfs` user to `chown` (change ownership of) all files. The command below will restart the HDFS service on Isilon to ensure that any cached user mapping rules are flushed. This will temporarily interrupt any HDFS connections coming from other Hadoop clusters.

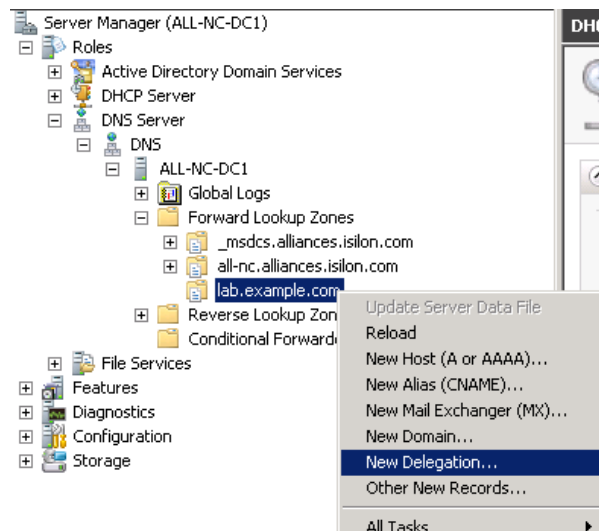
```
isiloncluster1-l# isi zone zones modify --user-mapping-rules="hdfs=>root" \  
--zone zone1  
  
isiloncluster1-l# isi services isi_hdfs_d disable ; \  
isi services isi_hdfs_d enable  
The service 'isi_hdfs_d' has been disabled.  
The service 'isi_hdfs_d' has been enabled.
```

Create DNS Records For Isilon

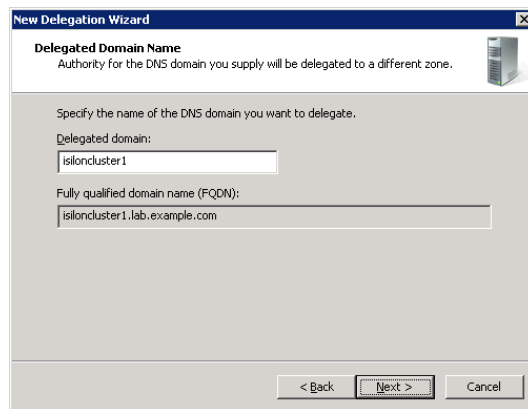
You will now create the required DNS records that will be used to access your Isilon cluster.

Create a delegation record so that DNS requests for the zone `isiloncluster1.lab.example.com` are delegated to the Service IP that will be defined on your Isilon cluster. The Service IP can be any unused static IP address in your lab subnet.

16. If using a Windows Server 2008 R2 server, see the screenshot below to create the delegation.

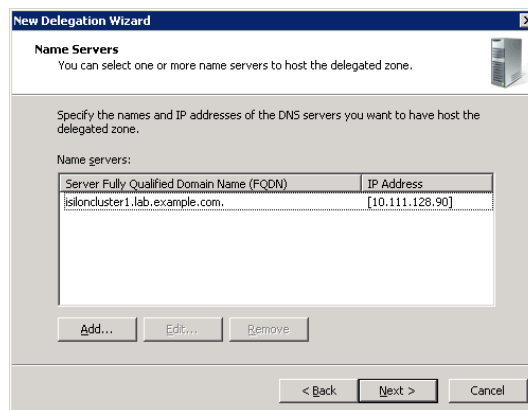


17. Enter the DNS domain that will be delegated. This should identify your Isilon cluster. Click Next.



The 'New Delegation Wizard' window shows the 'Delegated Domain Name' step. It instructs the user to specify the name of the DNS domain to be delegated. The 'Delegated domain:' field contains 'isiloncluster1'. The 'Fully qualified domain name (FQDN):' field contains 'isiloncluster1.lab.example.com'. Navigation buttons at the bottom include '< Back', 'Next >', and 'Cancel'.

18. Enter the DNS domain again.

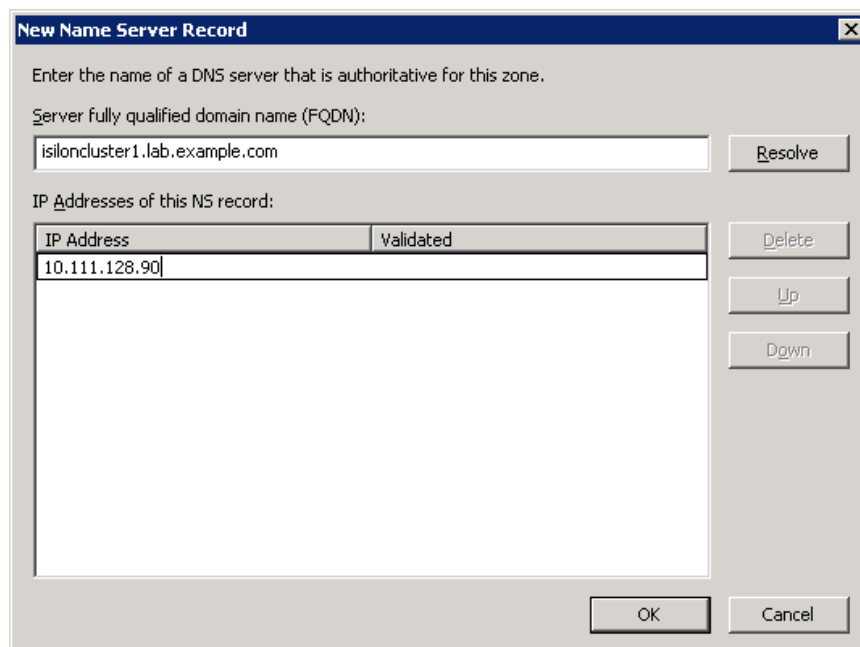


The 'New Delegation Wizard' window shows the 'Name Servers' step. It instructs the user to specify the names and IP addresses of the DNS servers to host the delegated zone. A table lists the name servers:

Server Fully Qualified Domain Name (FQDN)	IP Address
isiloncluster1.lab.example.com.	[10.111.128.90]

Buttons for 'Add...', 'Edit...', and 'Remove' are below the table. Navigation buttons at the bottom include '< Back', 'Next >', and 'Cancel'.

19. Then enter the Isilon SmartConnect Zone Service IP address. At this time, you can ignore any warning messages about not being able to validate the server. Click OK.



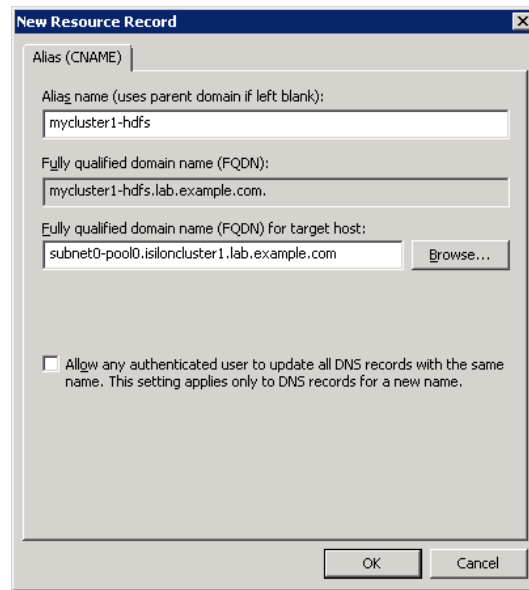
The 'New Name Server Record' window prompts the user to enter the name of a DNS server authoritative for the zone. The 'Server fully qualified domain name (FQDN):' field contains 'isiloncluster1.lab.example.com'. A 'Resolve' button is to the right. Below, the 'IP Addresses of this NS record:' section contains a table:

IP Address	Validated
10.111.128.90	

To the right of the table are 'Delete', 'Up', and 'Down' buttons. At the bottom are 'OK' and 'Cancel' buttons.

20. Click Next and then Finish.

21. If you are using your own lab DNS server, you should create a CNAME alias for your Isilon SmartConnect zone. For example, create a CNAME record for mycluster1-hdfs.lab.example.com that targets subnet0-pool0.isiloncluster1.lab.example.com.



22. Test name resolution.

```
[user@workstation ~]$ ping mycluster1-hdfs.lab.example.com
PING subnet0-pool0.isiloncluster1.lab.example.com (10.111.129.115) 56(84)
bytes of data.
64 bytes from 10.111.129.115: icmp_seq=1 ttl=64 time=1.15 ms
```

Prepare NFS Clients

To allow for scripts and other small files to be easily shared between all servers in your environment, it is highly recommended to enable NFS (Unix Sharing) on your Isilon cluster. By default, the entire /ifs directory is already exported but could be changed as it represents a potential security hole. However, Isilon best-practices suggest creating an NFS mount for /ifs/isiloncluster1/scripts.

23. On Isilon, create an NFS export for /ifs/isiloncluster1/scripts. Enable read/write and root access from all hosts in your lab subnet.

24. Mount your NFS export on your workstation

```
[root@workstation ~]$ yum install nfs-utils

[root@workstation ~]$ mkdir /mnt/scripts

[root@workstation ~]$ echo \

subnet0-pool0.isiloncluster1.lab.example.com:/ifs/isiloncluster1/scripts \

/mnt/scripts nfs \

nolock,nfsvers=3,tcp,rw,hard,intr,timeo=600,retrans=2,rsz=131072,wsz=524288 \

>> /etc/fstab

[root@workstation ~]$ mount -a
```

Tune Isilon for Optimal Performance

Isilon has a wealth of tuning options available for workload optimization. In general if Isilon is dedicated for Hadoop it is best to turn off L3 caching and use Metadata Read Acceleration. Metadata Read Acceleration requires Nodes that have SSD.

In addition, Isilon can tune the access pattern. If Isilon will be primarily used for MR jobs then the I/O Optimization pattern should be set for Streaming on the HDFS Root. MR jobs are benefitted by the streaming access pattern by as much as 10%. Else the HDFS Root should be set for Concurrency. Based on testing Concurrency offers an increase of roughly 35% for Impala Queries.

Isilon HDFS threads are tunable. The recommendation is to leave this to Auto.

There are a number of other sysctl parameters for Isilon 10g network tuning. Specifically for hadoop workloads they do not advantage the workload and may actually decrease performance.

Install Cloudera Manager

25. Download Cloudera Manager 5.2.0 from http://www.cloudera.com/content/cloudera/en/documentation/core/latest/topics/cm_qs_quick_start.html, section *Download and Run the Cloudera Manager Server Installer*.

26. Launch the installer.

```
[root@c5manager-server-0 ~]# ./cloudera-manager-installer.bin
```

27. Accept all defaults and complete the installation process.

28. Browse to <http://hadoopmanager-server-0.lab.example.com:7180/>.

29. Login using the following account: Username: admin Password: admin

Welcome to Cloudera Manager. Which edition do you want to deploy?

Upgrading to Cloudera Enterprise Data Hub Edition provides important features that help you manage and monitor your Hadoop clusters in mission-critical environments.

	Cloudera Express	Cloudera Enterprise Data Hub Edition Trial	Cloudera Enterprise
License	Free	60 Days After the trial period, the product will continue to function as Cloudera Express. Your cluster and your data will remain unaffected.	Annual Subscription Upload License
Node Limit	Unlimited	Unlimited	Unlimited
CDH	✓	✓	✓
Core Cloudera Manager Features	✓	✓	✓
Advanced Cloudera Manager Features		✓	✓
Cloudera Navigator		✓	✓
Cloudera Support			✓

For full list of features available in Cloudera Express and Cloudera Enterprise, [click here](#). ²

[Continue](#)

Deploy a Cloudera Hadoop Cluster

Only some of the steps are documented below. Refer to the Cloudera Manager and CDH Quick Start Guide (http://www.cloudera.com/content/cloudera/en/documentation/core/latest/topics/cm_qs_quick_start.html) for complete details.

30. Login to Cloudera Manager.



The login form is titled "Login". It contains a "Username:" label with a person icon, a text input field containing "admin", a "Password:" label with a lock icon, a password input field containing "*****", a checkbox labeled "Remember me on this computer.", and a blue "Login" button.

31. Specify hosts for your CDH cluster installation.

Specify hosts for your CDH cluster installation.

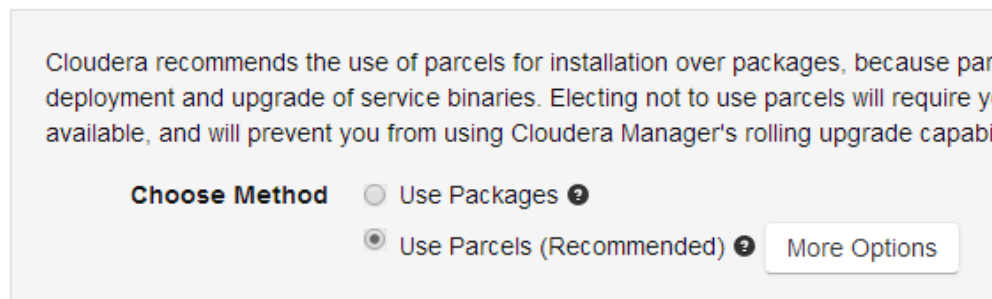


The form contains instructions: "Hosts should be specified using the same hostname (FQDN) that they will identify themselves with. Cloudera recommends including Cloudera Manager Server's host. This will also enable health monitoring for that host." A "Hint" says: "Search for hostnames and/or IP addresses using [patterns](#)." Below is a text area with the following content:
`c5manager-server-0.all-nc.alliances.isilon.com`
`mycluster1-namenode-0.all-nc.alliances.isilon.com`
`mycluster1-worker-0.all-nc.alliances.isilon.com`
`mycluster1-worker-1.all-nc.alliances.isilon.com`

You will likely want to install a specific version of CDH. When prompted to Select Repository, click More Options.

Cluster Installation

Select Repository



The form contains a paragraph: "Cloudera recommends the use of parcels for installation over packages, because parcel deployment and upgrade of service binaries. Electing not to use parcels will require you to manually manage the binaries, and will prevent you from using Cloudera Manager's rolling upgrade capabilities." Below is a "Choose Method" section with two radio buttons: "Use Packages" (unselected) and "Use Parcels (Recommended)" (selected). A "More Options" button is next to the "Use Parcels" option.

For example, To make CDH 5.1.3 available for selection, add the following to your Remote Parcel Repository URLs: `http://archive.cloudera.com/cdh5/parcels/5.1.3/`



The form shows a table with two rows. The first row has the URL `http://archive.cloudera.com/cdh5/parcels/latest/` and a button with a plus sign. The second row has the URL `http://archive.cloudera.com/cdh5/parcels/5.0.3/` and a button with a plus sign.

32. Now you can select the desired version of CDH. Then click Continue.

Cluster Installation

Select Repository

Cloudera recommends the use of parcels for installation over package deployment and upgrade of service binaries. Electing not to use parcels is not available, and will prevent you from using Cloudera Manager's role-based access control.

Choose Method

☐ Use Packages ?

☒ Use Parcels (Recommended) ?

Select the version of CDH

☐ CDH-5.1.0-1.cdh5.1.0.p0.53

☒ CDH-5.0.3-1.cdh5.0.3.p0.35

☐ CDH-4.7.0-1.cdh4.7.0.p0.40

33. Click Continue to select defaults and answer the prompts appropriately. When prompted to provide SSH login credentials.

34. When prompted to choose the CDH services that you want to install, select *Custom Services*. Then check *Isilon* and any other services that you would like to install. Do *not* select *HDFS*.

Cluster Setup

Choose the CDH 5 services that you want to install on your cluster.

Choose a combination of services to install.

- ☐ **Core Hadoop**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, and Sqoop
- ☐ **Core with HBase**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, Sqoop, and HBase
- ☐ **Core with Impala**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, Sqoop, and Impala
- ☐ **Core with Search**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, Sqoop, and Solr
- ☐ **Core with Spark**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, Sqoop, and Spark
- ☐ **All Services**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, Sqoop, HBase, Impala, Solr, Spark, and Key-Value Store Indexer
- ☒ **Custom Services**
Choose your own services. Services required by chosen services will automatically be included. Flume can be added after your initial cluster has been set up.

Service Type	Description
<input checked="" type="checkbox"/> HBase	Apache HBase provides random, real-time, read/write access to large data sets (requires HDFS and ZooKeeper).
<input type="checkbox"/> HDFS	Apache Hadoop Distributed File System (HDFS) is the primary storage system used by Hadoop applications. HDFS creates multiple replicas of data blocks and distributes them on compute hosts throughout a cluster to enable reliable, extremely rapid computations.
<input checked="" type="checkbox"/> Hive	Hive is a data warehouse system that offers a SQL-like language called HiveQL.
<input checked="" type="checkbox"/> Hue	Hue is a graphical user interface to work with Cloudera's Distribution Including Apache Hadoop (requires HDFS, MapReduce, and Hive).
<input checked="" type="checkbox"/> Impala	Impala provides a real-time SQL query interface for data stored in HDFS and HBase. Impala requires Hive service and shares Hive Metastore with Hue.
<input checked="" type="checkbox"/> Isilon	EMC Isilon is a distributed filesystem.
<input checked="" type="checkbox"/> Key-Value Store Indexer	Key-Value Store Indexer listens for changes in data inside tables contained in HBase and indexes them using Solr.
<input type="checkbox"/> MapReduce	Apache Hadoop MapReduce supports distributed computing on large data sets across your cluster (requires HDFS). YARN (MapReduce 2 Included) is recommended instead. MapReduce is included for backward compatibility.
<input checked="" type="checkbox"/> Oozie	Oozie is a workflow coordination service to manage data processing jobs on your cluster.
<input checked="" type="checkbox"/> Solr	Solr is a distributed service for indexing and searching data stored in HDFS.
<input checked="" type="checkbox"/> Spark	Apache Spark is an open source cluster computing system. This service runs Spark as an application on YARN.
<input checked="" type="checkbox"/> Sqoop 2	Sqoop is a tool designed for efficiently transferring bulk data between Apache Hadoop and structured datastores such as relational databases. The version supported by Cloudera Manager is Sqoop 2 .
<input checked="" type="checkbox"/> YARN (MR2 Included)	Apache Hadoop MapReduce 2.0 (MRV2), or YARN, is a data computation framework that supports MapReduce applications (requires HDFS).
<input checked="" type="checkbox"/> ZooKeeper	Apache ZooKeeper is a centralized service for maintaining and synchronizing configuration data.

[Back](#) 1 2 3 4 5 6 [Continue](#)

35. Customize role assignments. For test clusters, the following role assignment can be used:

View By Host

This table is grouped by hosts having the same roles assigned to them.

Hosts	Count	Existing Roles	Added Roles
cdhmini5-master-0 all-nc.alliances.isilon.com	1		
cdhmini5-worker-[0-2] all-nc.alliances.isilon.com	3		

hadoopmanager-server-0 for all Cloudera Management Service roles

mycluster1-worker-* for YARN Node Manager, HBase Region Server, Impala Daemon

mycluster1-master-0 for all other roles (e.g. YARN Resource Manager, Isilon Gateway)

For most small clusters you will need at least 3 “master” nodes, one manager node (Cloudera Manager), and your compute (worker) nodes. Master nodes function the majority of hadoop services (Impalad, Hbase Region Server, Hive Metastore, zookeeper, etc..). It is best to minimize role assignment of workers to the bear essentials. For clusters under 10 nodes, 3 masters is typical. As node count approaches 50 or more it is best to have 5 master nodes. This increases the fault tolerance to several of the core hadoop services such as Zookeeper.

36. When prompted to review changes, here you will specify your Isilon cluster address. **Default File System URI** `hdfs://mycluster1-hdfs.lab.example.com:8020` **WebHDFS URL** `http://mycluster1-hdfs.lab.example.com:8082/webhdfs/v1`

Cluster Setup

Review Changes

Default File System URI default_fs_name	Service-Wide (Isilon) C hdfs://mycluster1-hdfs.lab.example.com:8020 <small>Missing required value: Default File System URI</small>	The full file system URI, to be emitted as '%s default name'
WebHDFS URL webhdfs_url	Service-Wide (Isilon) C http://mycluster1-hdfs.lab.example.com:8082/webhdfs/v1	Full URL for the Web Interface of Isilon service.

37. Complete the installation process.
38. Review Cloudera Manager for any warnings or errors.

Cluster Setup

Congratulations!

The services are installed, configured, and running on your cluster.

Adding a Hadoop User

You must add a user account for each Linux user that will submit MapReduce jobs. The procedure below can be used to add a user named `hduser1`.

The steps below will create local user and group accounts on your Isilon cluster. If you are using a directory service such as Active Directory, and you want these users and groups to be defined in your directory service, then DO NOT run these steps. Instead, refer to the OneFS documentation and [EMC Isilon Best Practices for Hadoop Data Storage](#).

1. Add user to Isilon.

```
isiloncluster1-l# isi auth groups create hduser1 --zone zone1 \
--provider local
isiloncluster1-l# isi auth users create hduser1 --primary-group hduser1 \
--zone zone1 --provider local \
--home-directory /ifs/isiloncluster1/zone1/hadoop/user/hduser1
```

2. Add user to Hadoop nodes. Usually, this only needs to be performed on the master-0 node.

```
[root@mycluster1-master-0 ~]# adduser hduser1
```

3. Create the user's home directory on HDFS. `[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -mkdir -p /user/hduser1`

```
[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -chown hduser1:hduser1 \
/user/hduser1
```

```
[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -chmod 755 /user/hduser1
```

Functional Tests

The tests below should be performed to ensure a proper installation. Perform the tests in the order shown.

You must create the Hadoop user *hduser1* before proceeding.

HDFS

```
[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -ls /
Found 5 items
-rw-r--r--    1 root  hadoop                0 2014-08-05 05:59 /THIS_IS_ISILON
drwxr-xr-x    - hbase hbase              148 2014-08-05 06:06 /hbase
drwxrwxr-x    - solr  solr                0 2014-08-05 06:07 /solr
drwxrwxrwt    - hdfs supergroup          107 2014-08-05 06:07 /tmp
drwxr-xr-x    - hdfs supergroup          184 2014-08-05 06:07 /user
[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -put -f /etc/hosts /tmp
[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -cat /tmp/hosts
127.0.0.1 localhost
[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -rm -skipTrash /tmp/hosts
[root@mycluster1-master-0 ~]# su - hduser1
[hduser1@mycluster1-master-0 ~]$ hdfs dfs -ls /
Found 5 items
-rw-r--r--    1 root  hadoop                0 2014-08-05 05:59 /THIS_IS_ISILON
drwxr-xr-x    - hbase hbase              148 2014-08-05 06:28 /hbase
drwxrwxr-x    - solr  solr                0 2014-08-05 06:07 /solr
drwxrwxrwt    - hdfs supergroup          107 2014-08-05 06:07 /tmp
drwxr-xr-x    - hdfs supergroup          209 2014-08-05 06:39 /user
[hduser1@mycluster1-master-0 ~]$ hdfs dfs -ls
...

```

YARN / MapReduce

```
[hduser1@mycluster1-master-0 ~]$ hadoop jar \
/opt/cloudera/parcels/CDH/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar \
pi 10 1000
...
Estimated value of Pi is 3.140000000000000000000000
[hduser1@mycluster1-master-0 ~]$ hadoop fs -mkdir in
You can put any file into the in directory. It will be used the datasource for subsequent tests.
[hduser1@mycluster1-master-0 ~]$ hadoop fs -put -f /etc/hosts in
[hduser1@mycluster1-master-0 ~]$ hadoop fs -ls in
...
[hduser1@mycluster1-master-0 ~]$ hadoop fs -rm -r out
[hduser1@mycluster1-master-0 ~]$ hadoop jar \
/opt/cloudera/parcels/CDH/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar \
wordcount in out
...

```

```
[hduser1@mycluster1-master-0 ~]$ hadoop fs -ls out

Found 4 items

-rw-r--r--    1 hduser1 hduser1            0 2014-08-05 06:44 out/_SUCCESS
-rw-r--r--    1 hduser1 hduser1          24 2014-08-05 06:44 out/part-r-00000
-rw-r--r--    1 hduser1 hduser1            0 2014-08-05 06:44 out/part-r-00001
-rw-r--r--    1 hduser1 hduser1            0 2014-08-05 06:44 out/part-r-00002
```

```
[hduser1@mycluster1-master-0 ~]$ hadoop fs -cat out/part*
```

```
localhost      1
127.0.0.1      1
```

Browse to the YARN Resource Manager GUI <http://mycluster1-master-0.lab.example.com:8088/>.

Browse to the MapReduce History Server GUI <http://mycluster1-master-0.lab.example.com:19888/>. In particular, confirm that you can view the complete logs for task attempts.

Hive

```
[hduser1@mycluster1-master-0 ~]$ hadoop fs -mkdir -p sample_data/tab1
[hduser1@mycluster1-master-0 ~]$ cat - > tab1.csv
1,true,123.123,2012-10-24 08:55:00
2,false,1243.5,2012-10-25 13:40:00
3,false,24453.325,2008-08-22 09:33:21.123
4,false,243423.325,2007-05-12 22:32:21.33454
5,true,243.325,1953-04-22 09:11:33
Type <Control+D>.
[hduser1@mycluster1-master-0 ~]$ hadoop fs -put -f tab1.csv sample_data/tab1
[hduser1@mycluster1-master-0 ~]$ hive
hive>
DROP TABLE IF EXISTS tab1;
CREATE EXTERNAL TABLE tab1
(
    id INT,
    col_1 BOOLEAN,
    col_2 DOUBLE,
    col_3 TIMESTAMP
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
LOCATION '/user/hduser1/sample_data/tab1';

DROP TABLE IF EXISTS tab2;

CREATE TABLE tab2
(
    id INT,
    col_1 BOOLEAN,
    col_2 DOUBLE,
```



```

    month INT,
    day INT
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';

INSERT OVERWRITE TABLE tab2
SELECT id, col_1, col_2, MONTH(col_3), DAYOFMONTH(col_3)
FROM tab1 WHERE YEAR(col_3) = 2012;
...
OK
Time taken: 28.256 seconds

```

```

hive> show tables;
OK
tab1
tab2
Time taken: 0.889 seconds, Fetched: 2 row(s)

```

```

hive> select * from tab1;
OK
1      true   123.123      2012-10-24 08:55:00
2      false  1243.5        2012-10-25 13:40:00
3      false  24453.325      2008-08-22 09:33:21.123
4      false  243423.325     2007-05-12 22:32:21.33454
5      true   243.325        1953-04-22 09:11:33
Time taken: 1.083 seconds, Fetched: 5 row(s)

```

```

hive> select * from tab2;
OK
1      true   123.123      10      24
2      false  1243.5        10      25
Time taken: 0.094 seconds, Fetched: 2 row(s)

```

```

hive> select * from tab1 where id=1;
OK
1      true   123.123      2012-10-24 08:55:00
Time taken: 15.083 seconds, Fetched: 1 row(s)

```

```

hive> select * from tab2 where id=1;
OK
1      true   123.123      10      24
Time taken: 13.094 seconds, Fetched: 1 row(s)

```

```
hive> exit;
```

Pig

```
[hduser1@mycluster1-master-0 ~]$ pig
```

```
grunt> a = load 'in';
```

```
grunt> dump a;
```

```
...
```

```
Success!
```

```
...
```

```
grunt> quit;
```

HBase

```
[hduser1@mycluster1-master-0 ~]$ hbase shell
```

```
hbase(main):001:0> create 'test', 'cf'
```

```
0 row(s) in 3.3680 seconds
```

```
=> Hbase::Table - test
```

```
hbase(main):002:0> list 'test'
```

```
TABLE
```

```
test
```

```
1 row(s) in 0.0210 seconds
```

```
=> ["test"]
```

```
hbase(main):003:0> put 'test', 'row1', 'cf:a', 'value1'
```

```
0 row(s) in 0.1320 seconds
```

```
hbase(main):004:0> put 'test', 'row2', 'cf:b', 'value2'
```

```
0 row(s) in 0.0120 seconds
```

```
hbase(main):005:0> scan 'test'
```

ROW	COLUMN+CELL
row1	column=cf:a,timestamp=1407542488028,value=value1
row2	column=cf:b,timestamp=1407542499562,value=value2

```
2 row(s) in 0.0510 seconds
```

```
hbase(main):006:0> get 'test', 'row1'
```

COLUMN	CELL
cf:a	timestamp=1407542488028,value=value1

```
1 row(s) in 0.0240 seconds
```

```
hbase(main):007:0> quit
```

Impala

```
[hduser1@mycluster1-master-0 ~]$ impala-shell -i mycluster1-worker-0
```

```
[mycluster1-worker-0:21000] > invalidate metadata;
```

```
Query: invalidate metadata
```

```
Returned 0 row(s) in 1.05s
```

```
[cdhdas2-worker-0:21000] > show tables;
```

```
Query: show tables
```

```
+-----+
```

```
| name |
```

```
+-----+
```

```
| tab1 |
```

```
| tab2 |
```

```
+-----+
```

```
Returned 2 row(s) in 0.01s
```

```
[mycluster1-worker-0:21000] > select * from tab1;
```

```
Query: select * from tab1
```

```
+-----+-----+-----+-----+-----+-----+-----+
```

```
| id | col_1 | col_2 | col_3 |
```

```
+-----+-----+-----+-----+-----+-----+-----+
```

```
| 1 | true | 123.123 | 2012-10-24 08:55:00 |
```

```
| 2 | false | 1243.5 | 2012-10-25 13:40:00 |
```

```
| 3 | false | 24453.325 | 2008-08-22 09:33:21.123000000 |
```

```
| 4 | false | 243423.325 | 2007-05-12 22:32:21.334540000 |
```

```
| 5 | true | 243.325 | 1953-04-22 09:11:33 |
```

```
+-----+-----+-----+-----+-----+-----+-----+
```

```
Returned 5 row(s) in 0.20s
```

```
WARNINGS: Backend 0:Unknown disk id. This will negatively affect  
performance. Check your hdfs settings to enable block location metadata.
```

```
[mycluster1-worker-0:21000] > quit;
```

For more details regarding Impala, refer to http://www.cloudera.com/content/cloudera-content/cloudera-docs/Impala/latest/Installing-and-Using-Impala/ciu_tutorial.html.

Searching Wikipedia

One of the many unique features of Isilon is its multi-protocol support. This allows you, for instance, to write a file using SMB (Windows) or NFS (Linux/Unix) and then read it using HDFS to perform Hadoop analytics on it.

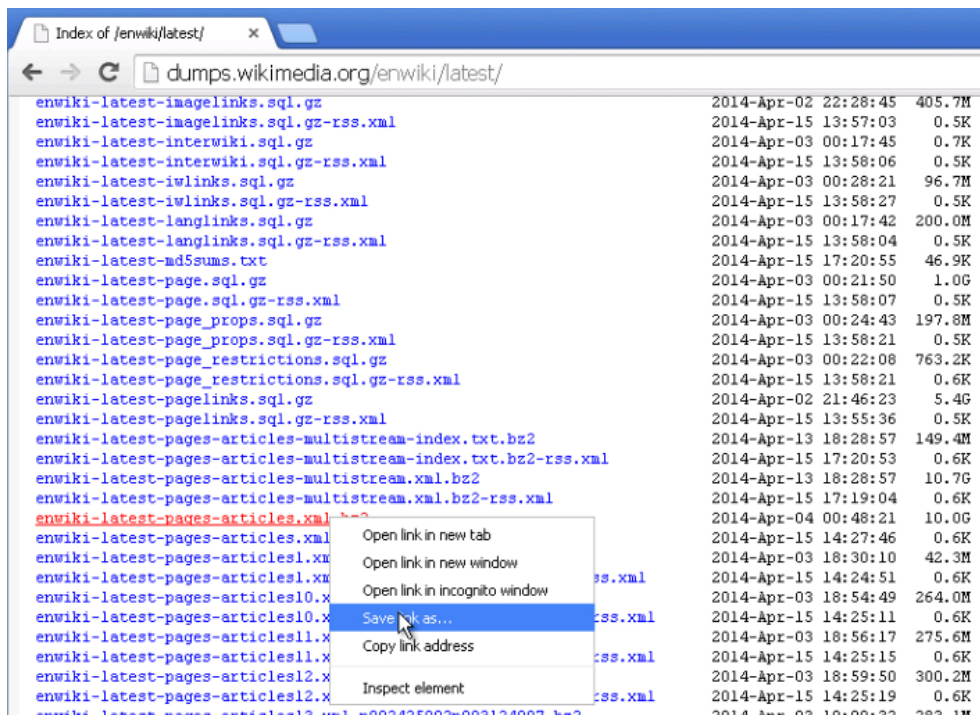
In this section, we exercise this capability to download the entire Wikipedia database (excluding media) using your favorite browser to Isilon. As soon as the download completes, we'll run a Hadoop grep to search the entire text of Wikipedia using our Hadoop cluster. As this search doesn't rely on a word index, your regular expression can be as complicated as you like.

1. First, let's connect your client (with your favorite web browser) to your Isilon cluster.
 - If you are using a Windows host or other SMB client:
 - Click Start -> Run.
 - Enter: `\\subnet0-pool0.isiloncluster1.lab.example.com\ifs`
 - You may authenticate as *root* with your Isilon root password.
 - Browse to `\\ifs\isiloncluster1\zone1\hadoop\tmp`.
 - Create a directory here called *wikidata*. This is where you will download the Wikipedia data to.

- If you are using a Linux host or other NFS client:

- Mount your NFS export. [root@workstation ~]\$ mkdir /mnt/isiloncluster1
- [root@workstation ~]\$ echo \
- subnet0-pool0.isiloncluster1.lab.example.com:/ifs \
- /mnt/isiloncluster1 nfs \
- noLOCK,nfsvers=3,tcp,rw,hard,intr,timeo=600,retrans=2,rsz=131072,wsz=524288 \
- >> /etc/fstab
- [root@workstation ~]\$ mount -a
- [root@workstation ~]\$ mkdir -p \
- /mnt/isiloncluster1/isiloncluster1/zone1/hadoop/tmp/wikidata

2. Open your favorite web browser and go to <http://dumps.wikimedia.org/enwiki/latest>.



3. Locate the file `enwiki-latest-pages-articles.xml.bz2` and download it directly to the `wikidata` folder on Isilon. Your web browser will be writing this file to the Isilon file system using SMB or NFS.

NOTE! This file is approximately 10 GB in size and contains the entire text of the English version of Wikipedia. If this is too large, you may want to download one of the smaller files such as `enwiki-latest-all-titles.gz`.

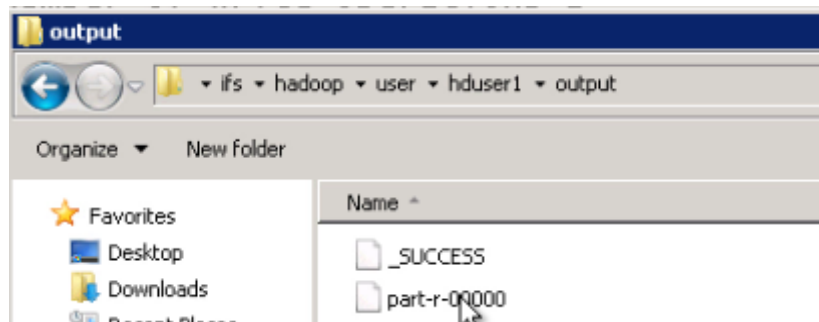
If you get an access-denied error, the quickest resolution is to SSH into any node in your Isilon cluster as root and run `chmod -R a+rwX /ifs/isiloncluster1/zone1/hadoop/tmp`.

4. Now let's run the Hadoop grep job. We'll search for all two-word phrases that begin with `EMC`.

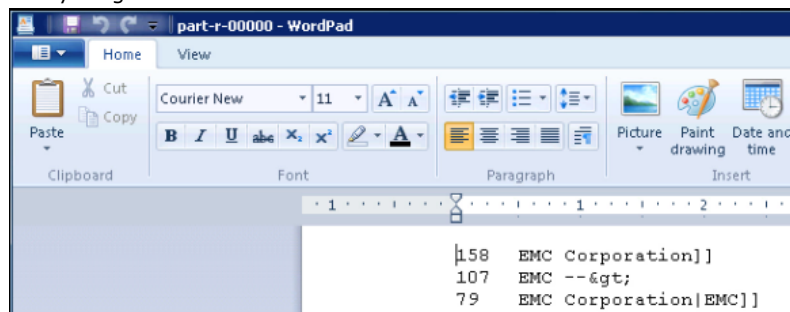
```
[hduser1@mycluster1-master-0 ~]$ hadoop fs -ls /tmp/wikidata
[hduser1@mycluster1-master-0 ~]$ hadoop fs -rm -r /tmp/wikigrep
[hduser1@mycluster1-master-0 ~]$ hadoop jar \
```

```
/opt/cloudera/parcels/CDH/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar \  
grep /tmp/wikidata /tmp/wikigrep "EMC [^ ]*"
```

When the the job completes, use your favorite text file viewer to view the output file `/tmp/wikigrep/part-r-00000`. If you are using Windows, you may open the file in Wordpad.



That's it! Can you think of anything else to search for?



Where To Go From Here

You are now ready to fine-tune the configuration and performance of your Isilon Hadoop environment. You should consider the following areas for fine-tuning.

YARN NodeManager container memory

Known Limitations

Although Kerberos is supported by OneFS, this document does not address a Hadoop environment secured with Kerberos. Instead, refer to [EMC Isilon Best Practices for Hadoop Data Storage](#).

References

[EMC Community - Isilon Hadoop Starter Kit Home Page](#)

[EMC Community - Isilon Hadoop Info Hub](#)

https://community.emc.com/community/connect/everything_big_data

<http://bigdatablog.emc.com/>

<http://www.emc.com/collateral/white-paper/h12877-wp-emc-isilon-hadoop-best-practices.pdf>

<http://www.emc.com/collateral/white-paper/h13354-wp-security-compliance-scale-out-hadoop-data-lakes.pdf>

<https://github.com/claudiofahey/isilon-hadoop-tools>

<http://www.cloudera.com>