



Cloudera Manager Installation Guide

Cloudera, Inc.
220 Portage Avenue
Palo Alto, CA 94306
info@cloudera.com
US: 1-888-789-1488
Intl: 1-650-362-0488
www.cloudera.com

Important Notice

© 2010-2012 Cloudera, Inc. All rights reserved.

Cloudera, the Cloudera logo, and any other product or service names or slogans contained in this document, except as otherwise disclaimed, are trademarks of Cloudera and its suppliers or licensors, and may not be copied, imitated or used, in whole or in part, without the prior written permission of Cloudera or the applicable trademark holder.

Hadoop and the Hadoop elephant logo are trademarks of the Apache Software Foundation. All other trademarks, registered trademarks, product names and company names or logos mentioned in this document are the property of their respective owners. Reference to any products, services, processes or other information, by trade name, trademark, manufacturer, supplier or otherwise does not constitute or imply endorsement, sponsorship or recommendation thereof by us.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Cloudera.

Cloudera may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Cloudera, the furnishing of this document does not give you any license to these patents, trademarks copyrights, or other intellectual property.

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.

Version: 4.1

Date: October 23, 2012

Contents

ABOUT THIS GUIDE	1
OTHER CLUSTERA MANAGER GUIDES	1
INTRODUCTION TO CLUSTERA MANAGER INSTALLATION	1
ABOUT THE CLUSTERA MANAGER INSTALLATION PROGRAM	2
ABOUT THE CLUSTERA MANAGER WIZARD.....	2
INSTALLING CLUSTERA MANAGER FOR THE FIRST TIME	3
REQUIREMENTS FOR CLUSTERA MANAGER.....	4
SUPPORTED OPERATING SYSTEMS FOR CLUSTERA MANAGER.....	4
SUPPORTED BROWSERS FOR CLUSTERA MANAGER ADMIN CONSOLE	5
OTHER REQUIREMENTS	5
<i>Version Support.....</i>	<i>5</i>
<i>Resources</i>	<i>5</i>
<i>Networking and Security.....</i>	<i>6</i>
SUPPORTED DATABASES FOR CLUSTERA MANAGER.....	8
INSTALLING AND CONFIGURING DATABASES	8
WHAT DATABASES MUST BE INSTALLED	9
ABOUT THE CLUSTERA MANAGER SERVER DATABASE	10
USING AN EXTERNAL DATABASE FOR STORING INFORMATION ABOUT MONITORING AND REPORTS	11
<i>Configuring an External Database for Oozie.....</i>	<i>11</i>
<i>Configuring an External Database for Hue</i>	<i>11</i>
<i>Backing up the Databases.....</i>	<i>11</i>
INSTALLING AND CONFIGURING A MYSQL DATABASE	12
<i>Configuring MySQL for the Service Monitoring and Activity Monitoring Databases.....</i>	<i>13</i>
<i>Installing the JDBC Connector to MySQL.....</i>	<i>15</i>
<i>Configuring MySQL</i>	<i>16</i>
INSTALLING AND CONFIGURING AN EXTERNAL POSTGRESQL DATABASE.....	20
<i>Configuring Your Systems to Support PostgreSQL</i>	<i>20</i>
<i>Creating the PostgreSQL Databases for Clustera Manager</i>	<i>22</i>
<i>Backing up the Databases.....</i>	<i>25</i>
INSTALLING AN EMBEDDED POSTGRESQL DATABASE	26
USING AN ORACLE DATABASE.....	27

<i>Collect Oracle Database Information</i>	27
<i>Install the JDBC Connector to Oracle</i>	28
<i>Adjust Oracle Settings to Accommodate Larger Clusters</i>	28
<i>Adjust Oracle System Settings for Sufficient Database Connectivity</i>	29
<i>Ensure your Oracle Database Supports UTF8</i>	29
INSTALLING CDH AND CLOUDERA MANAGER	30
INSTALLATION PATH A: AUTOMATED INSTALLATION BY CLOUDERA MANAGER	30
INSTALLATION PATH B: INSTALLATION USING YOUR OWN METHOD	30
UPGRADING TO CLOUDERA MANAGER 4.1	30
INSTALLATION PATH A - AUTOMATED INSTALLATION BY CLOUDERA MANAGER	31
<i>Step 1: Download and Run the Cloudera Manager Installer</i>	31
<i>Step 2: Start the Cloudera Manager Admin Console</i>	33
<i>Step 3: Use Cloudera Manager for Automated CDH Installation and Configuration</i>	34
<i>Step 4: Change the Default Administrator Password</i>	40
<i>Step 5: Test the Installation</i>	40
INSTALLATION PATH B - INSTALLATION USING YOUR OWN METHOD.....	40
<i>Before You Begin</i>	41
<i>Step 1: Install CDH</i>	42
<i>Step 2: Install the Cloudera Manager Server</i>	44
<i>Step 3: Configure a Database for the Cloudera Manager Server</i>	45
<i>Step 4: Install the Cloudera Manager Agents</i>	49
<i>Step 5: Start the Cloudera Manager Server</i>	51
<i>Step 6: Start the Cloudera Manager Agents</i>	51
<i>Step 7: Start the Cloudera Manager Admin Console</i>	52
<i>Step 8: Configure Services</i>	53
<i>Step 9: Change the Default Administrator Password</i>	55
<i>Step 10: Test the Installation</i>	55
INSTALLING IMPALA WITH CLOUDERA MANAGER	56
STEP 1: INSTALL CDH AND HIVE	56
STEP 2: INSTALL A DATABASE FOR THE HIVE METASTORE	56
STEP 3: CONFIGURE THE REMOTE DATABASE AS THE HIVE METASTORE	57
<i>Configuring Remote MySQL Database</i>	57

STEP 4: ADD THE IMPALA SERVICE.....	58
UPGRADING TO CLUSTERA MANAGER 4.1.....	59
UNDERSTANDING UPGRADES.....	59
<i>Before Upgrading.....</i>	<i>59</i>
<i>During the Upgrade</i>	<i>60</i>
<i>After Upgrading</i>	<i>60</i>
<i>Upgrade Paths</i>	<i>60</i>
DATABASE CONSIDERATIONS FOR CLUSTERA MANAGER UPGRADES	61
<i>Back up Databases.....</i>	<i>61</i>
<i>Modify Databases to Support UTF-8.....</i>	<i>61</i>
<i>Modify MySQL Databases to Support Larger Thread Stacks.....</i>	<i>62</i>
<i>Modify Databases to Support Appropriate Maximum Connections</i>	<i>62</i>
<i>Next Steps</i>	<i>63</i>
UPGRADE FROM CLUSTERA MANAGER 3.7.X TO CLUSTERA MANAGER 4.1	64
<i>Summary: What You are Going to Do.....</i>	<i>64</i>
<i>Upgrade Clusera Manager Server</i>	<i>64</i>
<i>Upgrade the Cluster Hosts</i>	<i>66</i>
<i>Deploy Updated Client Configurations.....</i>	<i>69</i>
<i>Verify the Upgrade.....</i>	<i>69</i>
<i>Upgrade CDH</i>	<i>70</i>
UPGRADE FROM CLUSTERA MANAGER 4 TO THE LATEST CLUSTERA MANAGER 4.....	70
<i>Summary: What You are Going to Do.....</i>	<i>70</i>
<i>Step 1. Stop the Clusera Management Service.....</i>	<i>70</i>
<i>Step 2. Upgrade the Clusera Manager Server and Agent Packages.....</i>	<i>71</i>
<i>Step 3. Start the Server</i>	<i>72</i>
<i>Step 4. Upgrade the Cluster Hosts</i>	<i>73</i>
<i>Step 5. Verify the Upgrade Succeeded</i>	<i>75</i>
<i>Testing the Installation</i>	<i>76</i>
UPGRADE FROM CLUSTERA MANAGER FREE EDITION 4 TO CLUSTERA MANAGER 4.....	76
<i>Step 1. Install the Clusera Manager license.....</i>	<i>76</i>
<i>Step 2. Run the upgrade wizard</i>	<i>77</i>
<i>Upgrading CDH</i>	<i>79</i>

UPGRADING CDH IN A CLouDERA MANAGED DEPLOYMENT	80
UPGRADING CDH3 TO CDH4 IN A CLouDERA MANAGED DEPLOYMENT	80
<i>Before You Begin</i>	<i>80</i>
<i>Upgrading to CDH4</i>	<i>80</i>
UPGRADING TO THE LATEST VERSION OF CDH4 IN A CLouDERA MANAGED DEPLOYMENT	89
<i>Before You Begin</i>	<i>89</i>
<i>Step 1. Stop all the CDH Services on All Hosts</i>	<i>90</i>
<i>Step 2. Back up the HDFS Metadata on the NameNode</i>	<i>91</i>
<i>Step 3. Upgrade Managed Components</i>	<i>91</i>
<i>Step 4. Start the Services you Stopped</i>	<i>95</i>
UPGRADING TO THE LATEST VERSION OF CDH3 IN A CLouDERA MANAGED DEPLOYMENT	96
<i>Before You Begin</i>	<i>96</i>
<i>Step 1. Stop all the CDH Services on All Hosts</i>	<i>97</i>
<i>Step 2. Back up the HDFS Metadata on the NameNode</i>	<i>97</i>
<i>Step 3. Upgrade Managed Components</i>	<i>98</i>
<i>Step 4. Start the Services you Stopped</i>	<i>102</i>
SPECIFYING THE RACKS FOR HOSTS	102
TESTING THE INSTALLATION	103
ENABLING THE OOZIE WEB CONSOLE	104
USING AN EXTERNAL DATABASE FOR OOZIE	104
USING AN EXTERNAL DATABASE FOR HUE	105
USING CUSTOM JAVA HOME LOCATIONS	106
MODIFYING CMF_AGENT_JAVA_HOME	106
MODIFYING SERVICE SETTINGS	107
DEPLOYING CLIENTS	108
UNINSTALLING CLouDERA MANAGER	108
UNINSTALLING CLouDERA MANAGER SERVER AND AGENTS	108
<i>Step 1: Stop all services</i>	<i>108</i>
<i>Step 2: Uninstall the Cloudera Manager Server.</i>	<i>109</i>
<i>Step 3: On all Agent hosts, uninstall CDH and the Cloudera Manager Agents.</i>	<i>110</i>
<i>Step 4: On all Agent hosts, remove all Cloudera Manager data.</i>	<i>111</i>
<i>Step 5: On all Agent hosts, kill any running Cloudera Manager and Hadoop processes.</i>	<i>111</i>

<i>Step 6: Remove the Cloudera Manager lock file.</i>	112
TROUBLESHOOTING INSTALLATION AND UPGRADE PROBLEMS	112
CHECKING DATABASE HOSTNAMES	115
RECOVERING FROM CLOUDERA MANAGER HOST FAILURES	116
CHANGING EMBEDDED POSTGRESQL DATABASE PASSWORDS	118
GETTING HELP AND SUPPORT	119
CLOUDERA SUPPORT	119
COMMUNITY SUPPORT	119
REPORT ISSUES	119
GET ANNOUNCEMENTS ABOUT NEW CDH AND CLOUDERA MANAGER RELEASES	120
APPENDIX A - UNDERSTANDING CUSTOM INSTALLATION SOLUTIONS	120
UNDERSTANDING HOW PACKAGE MANAGEMENT TOOLS WORK	120
<i>How Do Packaging and Package Management Tools Interact?</i>	120
<i>How Do Package Management Tools Find all Available Packages?</i>	121
<i>How Do I Use Package Management Tools To Install Older Versions of Cloudera Manager?</i>	122
CREATING AND USING YOUR OWN REPOSITORY	123
<i>Step 1: Download Installation Files</i>	123
<i>Step 2: Prepare the RPM or DEB Files</i>	123
<i>Step 3: Create a Repository</i>	124
<i>Step 4: Install a Web Server</i>	126
<i>Step 5: Publish Repository Files</i>	127

About this Guide

This guide explains how to install CDH and Cloudera Manager. This guide also explains how to use Cloudera Manager to install, configure, manage, and monitor CDH on your cluster. Cloudera Manager 4 supports managing CDH3 and CDH4.

Other Cloudera Manager Guides

Guide	Available Here
<i>Cloudera Manager 4.1.x Release Notes</i>	https://ccp.cloudera.com/display/ENT4DOC/Cloudera+Manager+4.1.x+Release+Notes
<i>Cloudera Manager User Guide</i>	https://ccp.cloudera.com/display/ENT4DOC/Cloudera+Manager+User+Guide
<i>Configuring Hadoop Security with Cloudera Manager</i>	https://ccp.cloudera.com/display/ENT4DOC/Configuring+Hadoop+Security+with+Cloudera+Manager
<i>Configuring TLS Security for Cloudera Manager</i>	https://ccp.cloudera.com/display/ENT4DOC/Configuring+TLS+Security+for+Cloudera+Manager
<i>Configuring Ports for Cloudera Manager</i>	https://ccp.cloudera.com/display/ENT4DOC/Configuring+Ports+for+Cloudera+Manager

Introduction to Cloudera Manager Installation

Cloudera Manager automates the installation and configuration of CDH on an entire cluster, requiring only that you have root SSH access to your cluster's machines, and access to the internet or a local repository with installation files for all these machines. Cloudera Manager consists of:

- A small self-executing Cloudera Manager installation program to install the Cloudera Manager Server and other packages in preparation for cluster host installation
- Cloudera Manager wizard for automating CDH installation and configuration on the cluster hosts
- Cloudera Manager features for monitoring and configuring the cluster after installation is completed

About the Cloudera Manager Installation Program

The Cloudera Manager installation program, which you will install on the host where you want to the Cloudera Manager Server to run, automatically:

- Installs the package repositories for Cloudera Manager and the Oracle Java Development Kit (JDK)
- Installs the Oracle JDK if it's not already installed
- Installs the Cloudera Manager Server
- Installs and configures an embedded PostgreSQL database

About the Cloudera Manager Wizard

After you have installed the Cloudera Manager Server and when you run it for the first time, you can use the Cloudera Manager wizard to do the following on the cluster hosts automatically.

- Using SSH, discover the cluster hosts you specify via IP address ranges or hostnames
- Configure the package repositories for Cloudera Manager, CDH, and the Oracle JDK
- Install the Cloudera Manager Agent and CDH (including Hue) on the cluster hosts
- Install the Oracle JDK if it's not already installed on the cluster hosts
- Determine mapping of services to host
- Suggest a Hadoop configuration and start the Hadoop services

Note:

- If you will use external databases, you must install and configure those databases before you start the wizard. These are the databases that will be used by Cloudera Manager, Service Monitor, Activity Monitor, Host Monitor, and Report Manager. See [Installing and Configuring Databases](#) for more information. If you will use the embedded PostgreSQL database, you do not have to prepare databases in advance.
- When you use the Cloudera Manager wizard to install or upgrade Cloudera Manager and/or CDH on your cluster hosts, all of those hosts need access to installation files. Installation files are available on the Internet at archive.cloudera.com or you can download installation files and create a local repository. For more information, see the individual installation and upgrade procedures, and the [Cloudera Manager FAQ](#).

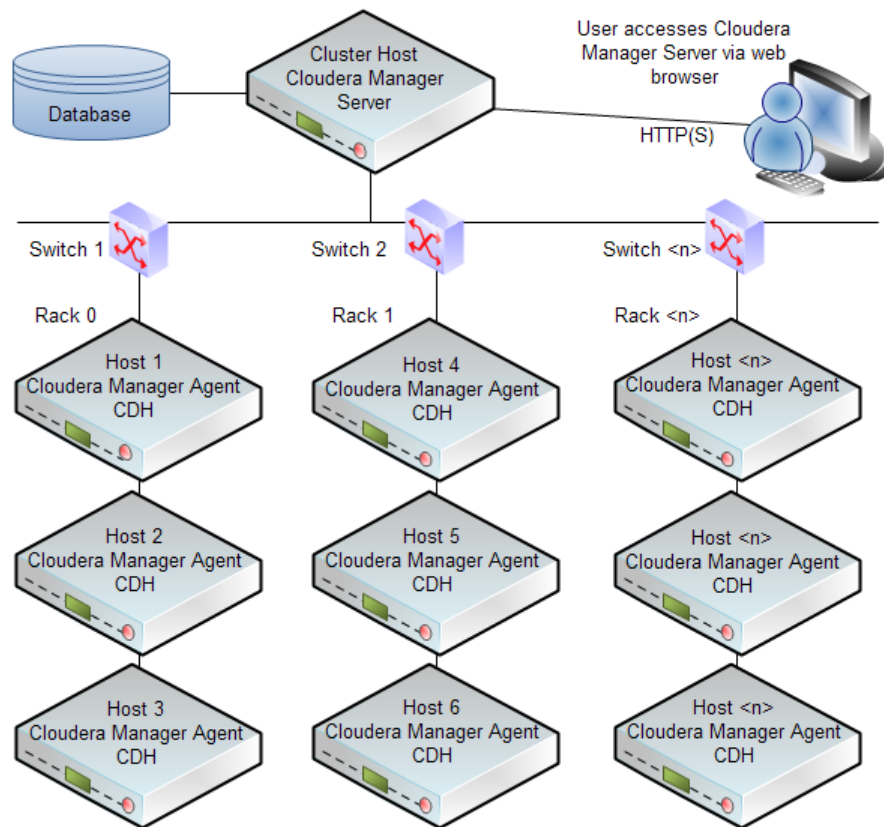
You can choose to abort the Cloudera Manager Agent and CDH installation process and Cloudera Manager wizard will automatically revert and completely rollback the installation process for any uninstalled components. (Installation that has completed successfully on a given host is not rolled back on that host.)

Installing Cloudera Manager for the First Time

To install Cloudera Manager, you will:

- Install a database application on the Cloudera Manager Server host machine or on a host machine that the Cloudera Manager Server can access, and (depending on the configuration you decide on) on other hosts as well.
- Install the Cloudera Manager Server on one cluster host machine.
- Install CDH and the Cloudera Manager Agents on the other cluster host machines.

The following diagram illustrates a simple example of where each component is installed.



Note

If you want to configure security for Cloudera Manager, see the following sections:

- [\(Optional\) Configuring Hadoop Security with Cloudera Manager](#)
- [\(Optional\) Configuring TLS Security for Cloudera Manager](#)

For overview and usage information, see the [Cloudera Manager User Guide](#).

Requirements for Cloudera Manager

Cloudera Manager interacts with a diversity of entities such as operating systems, databases, and browsers. Cloudera provides information about which major release version and minor release version is supported, where available. In some cases, such as some browsers, a minor version is not provided, but this information is provided, where available. After installing each element, upgrade to the latest patch version and apply any other appropriate updates. Note that the available updates may be specific to the operating system on which it is installed.

For example, you might be using CentOS in your environment. You could choose 6 as the major version and 2 as the minor version. These choices would mean you would be using CentOS 6.2. After installing this operating system, you would then apply any and all relevant CentOS 6.2 upgrades and patches.

For information on CDH requirements, see [Supported Operating Systems for CDH4](#).

Supported Operating Systems for Cloudera Manager

Cloudera Manager supports a range of operating systems including:

- Red Hat-compatible systems
 - Red Hat Enterprise Linux 5.7 and CentOS 5.7, 64-bit
 - Red Hat Enterprise Linux 6.2 and CentOS 6.2, 64-bit
 - Oracle Enterprise Linux 5.6 with Unbreakable Enterprise Kernel, 64-bit
- SLES systems
 - SUSE Linux Enterprise Server 11, 64-bit. Service Pack 1 or later is required. Also, the SUSE Linux Enterprise Software Development Kit 11 SP1 is required on cluster hosts running the Cloudera Manager Agents (not required on the Cloudera Manager Server host); you can download the SDK [here](#).
- Debian systems
 - Debian 6.0 (Squeeze), 64-bit
- Ubuntu systems
 - Ubuntu 10.04 (Lucid Lynx), 64-bit
 - Ubuntu 12.04 (Precise Pangolin), 64-bit

Note

For Hadoop to work properly, using the same version of the same operating system on all cluster hosts is strongly recommended.

Supported Browsers for Cloudera Manager Admin console

The Cloudera Manager Admin console, which you use to configure, manage, and monitor CDH, supports the following browsers:

- Firefox 3.6
- Firefox 11 or later
- Google Chrome
- Internet Explorer 8
- Internet Explorer 9
- Safari 5 or later

Other Requirements

Cloudera Manager supports a variety of services and depends on resources being available.

Version Support

- Cloudera Manager 4.1 supports CDH3 Update 1 (cdh3u1) or later and CDH4.0 or later. CDH3 Update 2 or later is strongly recommended.

Warning

Cloudera Manager 4.1 works with CDH3 and CDH4.0 or later, but does not work with CDH4.0 beta. You must upgrade any installations of CDH4.0 beta.

- If you want to use Cloudera Manager to manage Oozie, CDH3 Update 2 or later is required.
- Cloudera Manager uses Python. Python is part of the default installation for all operating systems that Cloudera Manager supports, so there is no need to complete any installation tasks to make Python available. Cloudera Manager is tested with the default installation. Modifying the Python installation available on systems on which you install Cloudera Manager is not supported.
- Impala 0.1.

Resources

Cloudera Manager requires sufficient:

- Disk space. For example, `/var` should be allocated a minimum of 5 GB of space.
- RAM. 4 GB is appropriate in most cases. 2 GB may be sufficient for smaller deployments. Cloudera Manager may need more than 4 GB for more demanding configurations, such as those involving significant caching.

Networking and Security

- Cluster hosts must have a working network name resolution system. Properly configuring DNS and reverse DNS meets this requirement.

If you use `/etc/hosts` instead of DNS, all hosts files must contain consistent information about host names and addresses across all nodes. For example, `/etc/hosts` might contain something of the form:

```
127.0.0.1 localhost.localdomain localhost
192.168.1.1 cluster-01.domain cluster-01
192.168.1.2 cluster-02.domain cluster-02
192.168.1.3 cluster-03.domain cluster-03
```

- In most cases, the Cloudera Manager Server must have SSH access to the cluster hosts when you run the installation or upgrade wizard. This does not apply if you install Cloudera Manager using [Path B](#).

Note

You must log in using a root account or an account that has password-less sudo permission. For authentication during the installation and upgrade procedures, you will need to either enter the password or upload a public and private key pair for the root or sudo user account. If you want to use a public and private key pair, the public key must be installed on the cluster hosts before you use Cloudera Manager. Authentication is not supported for accounts that have password-protected sudo permission.

Cloudera Manager uses SSH only during the initial install or upgrade. Once your cluster is set up, you can safely disable root SSH access or change the root password. Cloudera Manager does not save SSH credentials and all credential information is discarded once the installation is complete.

- No blocking by iptables or firewalls; make sure port 7180 is open because it is the port used to access Cloudera Manager after installation. Cloudera Manager communicates using specific ports, which must be open. For additional port information, see [Configuring Ports for Cloudera Manager Free Edition](#).
- No blocking by Security-Enhanced Linux (SELinux).

- Cloudera Manager and CDH use several user accounts and groups to complete their tasks. The set of user accounts and groups varies according to which components you choose to install. Do not delete these accounts or groups and do not modify their permissions and rights. Ensure no existing systems obstruct the functioning of these accounts and groups. For example, if you have scripts that delete user accounts not in a white-list, add these accounts to the list of permitted accounts. Cloudera Manager and CDH create and use the following accounts and groups:

Account	Type	Product	Comment
cloudera-scm	User and group	Cloudera Manager	
mapred	User and group	CDH3 and CDH4	MapReduce
hdfs	User and group	CDH3 and CDH4	Distributed file system
zookeeper	User and group	CDH3 and CDH4	Distributed system coordination service
yarn	User and group	CDH4	MapReduce2.0 or MRv2
httpfs	User and group	CDH3 and CDH4	HTTP gateway to HDFS
hbase	User and group	CDH3 and CDH4	Hadoop database
hive	User and group	CDH3 and CDH4	Hadoop data warehouse
hue	User and group	CDH3 and CDH4	Web interface to hadoop
oozie	User and group	CDH3 and CDH4	Workflow coordination system
flume	User and group	CDH3 and CDH4	Log collection system
hadoop	Group	CDH3 and CDH4	
impala	User and group	CDH4.1	Interactive query tool

Installing and Configuring Databases

- The Cloudera Manager Agent runs as root so that it can make sure the required directories are created and that processes and files are owned by the appropriate user (for example, the `hdfs` user and `mapred` user).

For additional port information, see [Configuring Ports for Cloudera Manager](#).

Supported Databases for Cloudera Manager

Cloudera Manager requires several databases. The Cloudera Manager server stores information about configured services, role assignments, configuration history, commands, users, and running processes in a database of its own. The Activity Monitor, Service Monitor, Report Manager, and Host Monitor also each use a database to store information.

The database you choose to use must be configured to support UTF8 character set encoding. The embedded PostgreSQL database that is installed using [Path A](#) automatically provides UTF8 encoding. If you install a custom database, you may need to enable UTF8 encoding. The commands for enabling UTF8 encoding are described in each database's section under [Installing and Configuring Databases](#).

After installing a database, upgrade to the latest patch version and apply any other appropriate updates. Note that the available updates may be specific to the operating system on which it is installed.

Cloudera Manager and its supporting services can use the following database systems and releases:

- MySQL:
 - 5.0
 - 5.1
 - 5.5
- Oracle
 - 10g Release 2
 - 11g Release 2
- PostgreSQL
 - 8.1
 - 8.3
 - 8.4

Installing and Configuring Databases

Cloudera Manager uses databases to store information about the Cloudera Manager configuration, as well as information such as the health of the system or task progress. Cloudera Manager supports using a variety of databases to store required information. To facilitate rapid completion of simple installations, the Cloudera Manager can install and configure a PostgreSQL database as part of the

broader Cloudera Manager installation process. This automatically installed database is sometimes referred to as an embedded PostgreSQL database. While the embedded database is a useful option for getting started quickly, Cloudera Manager also allows you to use other databases. You can opt to use your own PostgreSQL database or MySQL or Oracle databases.

If you plan to use the embedded database provided during the Cloudera Manager installation for all databases, you can skip ahead to [Installation Path A - Automated Installation by Cloudera Manager](#). To learn more about database options or if you are unsure whether or not using the embedded database is right for your environment, continue reading.

What Databases Must Be Installed

The Cloudera Manager Server and the server's Activity Monitor, Service Monitor, Report Manager, and Host Monitor all require databases. Cloudera Manager does support deploying different types of databases in a single environment, but doing so may create unexpected complications. Cloudera recommends choosing one of the three database providers to use for all five of the Cloudera Manager databases.

Cloudera provides two install paths:

- Path A automatically installs embedded PostgreSQL databases to meet the requirements of the services. This path reduces the number of installation tasks you must complete, as well as the number of choices to make.
- Path B requires you have databases in your environment for use by Cloudera Manager and the monitoring services. This path requires more input and intervention as you either install databases or gather information about existing databases. This path also provides greater flexibility in choosing database types and configurations.

A service works with a database. In most cases, you should install databases and services on the same host. For example, if you create the database for Activity Monitor on `myhost1`, then you should typically assign the Activity Monitor role to `myhost1`. You will assign the Service Monitor, Activity Monitor, Report Manager, and Host Monitor roles in the Cloudera Manager wizard during the install or upgrade process. After completing the install or upgrade process, you can also modify role assignments in the Management services pages of Cloudera Manager. While it is true that database location is changeable, before beginning an installation or upgrade, you should decide which hosts you will use. Note that the JDBC connector for your database **must** be installed on the hosts where you assign the Service Monitor, Activity Monitor, Report Manager, and Host Monitor roles. Installing JDBC connectors is described later in this guide.

It is possible to install the database and services on different hosts. Separating databases from services is more likely to occur in larger deployments and in cases where more sophisticated database administrators actively choose to establish such a configuration. For example, databases and services might be separated if your environment includes Oracle databases that will be separately managed by Oracle database administrators (DBAs).

The table that follows provides a summary; details are in the sections that follow.

Install or Upgrade Path	Install Supported Database For	Typically Install Databases on Systems That Will Host
Installation Path A - Automated Installation by Cloudera Manager	No installations required. Automated installation automatically creates embedded PostgreSQL databases for all Cloudera Manager and all services.	Cloudera Manager, Activity Monitor, Service Monitor, Report Manager, and Host Monitor roles. Placement can be adjusted using the configuration wizard as part of the installation process.
Installation Path B - Installation Using Your Own Method	The Cloudera Manager Server configuration and for Activity Monitor, Service Monitor, Report Manager, and Host Monitor.	The Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, and Host Monitor roles. Alternately, you may install these databases on other systems, assuming those systems are accessible to the Cloudera Manager Server.
Upgrade from Cloudera Manager Free Edition 3.7.x, Cloudera Manager Full Edition 3.7.x, or SCM Express 3.6 to Cloudera Manager 4.1 Full Edition	Activity Monitor, Service Monitor, Report Manager, and Host Monitor.	Activity Monitor, Service Monitor, Report Manager, and Host Monitor roles.

About the Cloudera Manager Server Database

This database, which is used for storing information about services' configurations, is independent of the databases used by the Activity Monitor, Service Monitor, Report Manager, and Host Monitor.

Installation type	Process
Automatic installation: Installation Path A	The wizard automatically installs, configures, and uses embedded PostgreSQL databases to store information about service configuration, as well as the Activity Monitor, Service Monitor, Report Manager, and Host Monitor. This functionality is provided by the <code>cloudera-scm-server-db</code> package, and you can start and stop these databases using the <code>service cloudera-scm-server-db [start stop]</code> command. If you are using Installation Path A, you can proceed directly to Installation Path A .
Manual installation:	You must install a supported database. This database can be installed on the

Installation type	Process
Installation Path B	machine where you install the Cloudera Manager Server or on a machine accessible to the Cloudera Manager Server. You will need to configure the connection between Cloudera Manager and the database, as is described in the documentation for Path B.

The installation instructions for the database containing the Cloudera Manager Server's configuration are also included under [Installation Path B - Installation Using Your Own Method](#).

Important

If you use a MySQL database to store information about service configuration, make sure that the `InnoDB` engine is configured, not the `MyISAM` engine. Cloudera Manager will not start if its tables are configured with the `MyISAM` engine. (Typically, tables revert to `MyISAM` if the `InnoDB` engine is misconfigured.) To check what engine your tables are using, run the following command from the MySQL shell:

```
mysql> show table status;
```

Using an External Database for Storing Information about Monitoring and Reports

Configuring an External Database for Oozie

By default, Cloudera Manager uses Derby for Oozie's database. If you want to use an external database for Oozie, you would do the configuration after Cloudera Manager is installed. For more information, see [Using an External Database for Oozie](#).

Configuring an External Database for Hue

By default, Cloudera Manager uses SQLite for Hue's database. If necessary, you can configure Cloudera Manager to use an external database such as MySQL or PostgreSQL as the database for Hue - do this after Cloudera Manager is installed. For more information, see [Using an External Database for Hue](#).

Backing up the Databases

It's important that you periodically back up the databases that Cloudera Manager uses to store configuration, monitoring, and reporting data. Be sure to back up all of the databases you are using with Cloudera Manager:

- **Cloudera Manager database:** This is the most important database to back up. This database contains all the information about what services you have configured, their role assignments, all configuration history, commands, users, and running processes. This is a relatively small database, typically smaller than 100MB.

Installing and Configuring Databases

- Activity Monitor database: Contains information about past activities. In large clusters, this database can become very large.
- Service Monitor database: contains monitoring information about daemons. In large clusters, this database can become very large.
- Report Manager database: Keeps track of disk utilization over time. This database is typically medium-sized.
- Host Manager database: Contains information about host status. The number of hosts in the cluster affects this database's size, so the database size varies, but the database is typically large in deployments with many hosts.

Installing and Configuring a MySQL Database

You can use MySQL databases to store information for Cloudera Manager. Cloudera Manager requires databases to store information for Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, and Host Monitor, so you may need to create databases for each of those entities. See your MySQL documentation for more information about installation and configuration.

To install MySQL on a Red Hat system:

```
$ sudo yum install mysql-server
```

To install MySQL on a SLES system:

```
$ sudo zypper install mysql
$ sudo zypper install libmysqlclient_r15
```

To install MySQL on an Debian/Ubuntu system:

```
$ sudo apt-get install mysql-server
```

After issuing the command to install MySQL, you may need to respond to prompts to confirm that you do want to complete the installation. After installation completes, start the mysql daemon.

On Red Hat systems

```
$ sudo service mysqld start
```

On SLES and Debian/Ubuntu systems

```
$ sudo service mysql start
```

Configuring MySQL for the Service Monitoring and Activity Monitoring Databases

The default settings in the MySQL installations in most distributions are very conservative with regards to buffer sizes and memory usage. For the Service Monitoring and Activity Monitoring databases, Cloudera recommends that you update `/etc/my.cnf` to at least the values shown below. It is important that the `datadir` directory, which, by default, is `/var/lib/mysql`, is on a partition that has plentiful free space.

Recommended Settings

Important

- For a fresh MySQL installation, apply the settings below before you start MySQL for the first time.
- For an existing installation, you need to take some additional steps when changing InnoDB settings; follow the instructions in the next section, [Reconfiguring InnoDB Settings for an Existing MySQL Installation](#).

```
[mysqld]
# Disabling symbolic-links is recommended to prevent assorted security
risks;
# to do so, uncomment this line:
# symbolic-links=0

key_buffer                = 16M
key_buffer_size            = 32M
max_allowed_packet        = 16M
thread_stack               = 256K
thread_cache_size          = 64
query_cache_limit          = 8M
query_cache_size           = 64M
query_cache_type           = 1
# Important: see Configuring the Databases and Setting max_connections
max_connections            = 550

# log-bin should be on a disk with enough free space
log-bin=/x/home/mysql/logs/binary/mysql_binary_log

# For MySQL version 5.1.8 or later
binlog_format               = mixed

read_buffer_size           = 2M
read_rnd_buffer_size       = 16M
sort_buffer_size            = 8M
join_buffer_size           = 8M
```

```
# InnoDB settings
innodb_file_per_table = 1
innodb_flush_log_at_trx_commit = 2
innodb_log_buffer_size = 64M
innodb_buffer_pool_size = 2G
innodb_thread_concurrency = 8
innodb_flush_method = O_DIRECT
innodb_log_file_size = 512M

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```

Configuring the Databases and Setting max_connections

The definition of a small or large cluster is not absolute, so this information is intended as general guidance. For the purposes of this discussion, clusters with fewer than 50 nodes can be considered small clusters and clusters with more than 50 nodes can be considered large clusters.

Follow these guidelines:

- In a small cluster, you can store more than one database (for example, both the Activity Monitor and Service Monitor) on the same host. If you do this, you should:
 - Put each database on its own storage volume.
 - Allow 100 maximum connections for each database and then add 50 extra connections. For example, for two databases set the maximum connections to 250. If you store all five databases on one host (the database for Activity Monitor, Service Monitor, Report Manager, Host Monitor databases, the Cloudera Manager Server database), set the maximum connections to 550.
- In a large cluster, do not store more than one database on the same host. In such a case, use a separate host for each database/host pair. The hosts need not be reserved exclusively for databases, but each database should be on a separate host.

Reconfiguring InnoDB Settings for an Existing MySQL Installation

To update InnoDB settings on all hosts that are using an existing MySQL installation, proceed as follows.

1. Stop MySQL.

On Red Hat systems

```
$ sudo service mysqld stop
```

On SLES and Debian/Ubuntu systems

```
$ sudo service mysql stop
```

2. Edit the InnoDB entries in `/etc/my.cnf` as shown in the previous section.
3. Move the old InnoDB log files to a backup location. The two files to move are `/var/lib/mysql/ib_logfile0` and `/var/lib/mysql/ib_logfile1`. Make sure you move these files out of the `/var/lib/mysql/` directory (don't copy them and leave the originals in place).
4. Start MySQL.

On Red Hat systems

```
$ sudo service mysqld start
```

On SLES and Debian/Ubuntu systems

```
$ sudo service mysql start
```

Installing the JDBC Connector to MySQL

You must install the JDBC connector to MySQL on any host that connects from Cloudera Manager Server to database applications. If you use external MySQL databases, this means you must install the connector on the Cloudera Manager Server host, as well as hosts to which you assign the Activity Monitor, Service Monitor, Report Manager, and Host Monitor roles.

Cloudera recommends that you assign roles and their corresponding databases to the same host. While putting roles and databases on the same host is recommended, it is not required. You could install a service, such as Activity Monitor on one host and install the corresponding database, such as the Activity Monitor database on a separate host. In such a case you would install the JDBC connector on the host running the Activity Monitor, not on the host with the Activity Monitor database.

On Red Hat 6 systems, run these commands on the relevant host to install the connector:

```
$ sudo yum install mysql-connector-java
```

On Red Hat 5 systems, run these commands on the relevant host to install the connector:

1. Download the MySQL JDBC connector from <http://www.mysql.com/downloads/connector/j/5.1.html>.
2. Extract the JDBC driver JAR file from the downloaded file; for example:

```
tar zxvf mysql-connector-java-5.1.18.tar.gz
```

Installing and Configuring Databases

3. Add the JDBC driver to the relevant server; for example:

```
$ sudo cp mysql-connector-java-5.1.18/mysql-connector-java-5.1.18-  
bin.jar /usr/share/cmfd/lib/
```

If the target directory does not yet exist on this host, you can create it before copying the .jar file; for example:

```
$ sudo mkdir -p /usr/share/cmfd/lib/  
$ sudo cp mysql-connector-java-5.1.18/mysql-connector-java-5.1.18-  
bin.jar /usr/share/cmfd/lib/
```

On SLES systems, run this command on the relevant host to install the connector:

```
$ sudo zypper install mysql-connector-java
```

On Debian/Ubuntu systems, run this command on the relevant host to install the connector:

```
$ sudo apt-get install libmysql-java
```

Configuring MySQL

Configure MySQL to use a strong password and to start at boot. Note that in the following procedure, your current root password is blank. Press the Enter key when you're prompted for the root password.

To set the MySQL root password:

```
$ sudo /usr/bin/mysql_secure_installation  
[...]  
Enter current password for root (enter for none):  
OK, successfully used password, moving on...  
[...]  
Set root password? [Y/n] y  
New password:  
Re-enter new password:  
Remove anonymous users? [Y/n] Y  
[...]  
Disallow root login remotely? [Y/n] N  
[...]  
Remove test database and access to it [Y/n] Y  
[...]  
Reload privilege tables now? [Y/n] Y  
All done!
```


To make sure the MySQL server starts at boot:

- On Red Hat systems:

```
$ sudo /sbin/chkconfig mysqld on
$ sudo /sbin/chkconfig --list mysqld
mysqld          0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

- On SLES systems:

```
$ sudo chkconfig --add mysql
```

- On Debian/Ubuntu systems:

```
$ sudo chkconfig mysql on
```

Creating the MySQL Databases for Cloudera Manager

The next step involves creating databases and user accounts for all database-backed services in Cloudera Manager.

You must create databases for each of the following features that are part of the Management Services:

- Activity Monitor
- Service Monitor
- Report Manager
- Host Monitor

You can create these databases on the host where the Cloudera Manager Server will run, or on any other nodes in the cluster. For performance reasons, you should typically install each database on the host on which the service runs, as determined by the roles you will assign during installation or upgrade. In larger deployments or in cases where database administrators (DBAs) are managing the databases the services will use, databases may be separated from services, but do not undertake such an implementation lightly.

The examples that follow allow access only to a specific user ('amon_user' on 'myhost1'), ('smon_user' on 'myhost2'), ('repman_user' on 'myhost3'), or ('hmon_user' on 'myhost4') respectively) where 'myhost1', 'myhost2', 'myhost3', and 'myhost4' refer to the name of the host on which you are creating the database. To restrict access in this way, you must use the hostname if this host will also have the corresponding role (Activity Monitor, Service Monitor, Report Manager, or Host Monitor respectively) as Cloudera recommends. But if instead another host will have the corresponding role, and you want to allow access to the database only from that host, you must specify the fully-qualified domain name of that host.

Installing and Configuring Databases

If you later decide to move the role (for example, Activity Monitor) to another machine, you must grant access to the corresponding user (for example 'amon_user') on the new host as well.

Note the values you enter for database names, user names, and passwords. The Cloudera Manager installation wizard requires this information to correctly connect to these databases.

The database must be configured to support UTF-8 character set encoding. The sample commands below include the required options to enable UTF-8 support.

To create the MySQL Databases for Cloudera Manager:

1. Log into MySQL as the root user:

```
$ mysql -u root -p
Enter password:
```

2. Create a database for the Activity Monitor. The database name, user name, and password can be anything you want. For example:

```
mysql> create database activity_monitor DEFAULT CHARACTER SET utf8;
Query OK, 1 row affected (0.00 sec)

mysql> grant all on activity_monitor.* TO 'amon_user'@'myhost1'
IDENTIFIED BY 'amon_password';
Query OK, 0 rows affected (0.00 sec)
```

3. Create a database for the Service Monitor. The database name, user name, and password can be anything you want. For example:

```
mysql> create database service_monitor DEFAULT CHARACTER SET utf8;
Query OK, 1 row affected (0.00 sec)

mysql> grant all on service_monitor.* TO 'smon_user'@'myhost2'
IDENTIFIED BY 'smon_password';
Query OK, 0 rows affected (0.00 sec)
```

4. Create a database for the Report Manager. The database name, user name, and password can be anything you want. For example:

```
mysql> create database reports_manager DEFAULT CHARACTER SET utf8;
Query OK, 1 row affected (0.00 sec)

mysql> grant all on reports_manager.* TO 'repman_user'@'myhost3'
IDENTIFIED BY 'repman_password';
Query OK, 0 rows affected (0.00 sec)
```

5. Create a database for the Host Monitor. The database name, user name, and password can be anything you want. For example:

```
mysql> create database host_monitor DEFAULT CHARACTER SET utf8;
Query OK, 1 row affected (0.00 sec)

mysql> grant all on host_monitor.* TO 'hmon_user'@'myhost4'
IDENTIFIED BY 'hmon_password';
Query OK, 0 rows affected (0.00 sec)
```

Backing Up the MySQL Databases

To back up the MySQL database, run the `mysqldump` command on the MySQL host, as follows:

```
$ mysqldump -h<hostname> -u<username> -p<password> <database> >
/tmp/<database-backup>.sql
```

For example, to back up database `scm_database` on the local host as the root user, with the password `mypasswd`:

```
$ mysqldump -pmypasswd scm_database > /tmp/scm_database-backup.sql
```

To back up database `scm_database` on remote host `myhost.example.com` as the root user, with the password `mypasswd`:

```
$ mysqldump -hmyhost.example.com -uroot -pcloudera scm_database >
/tmp/scm_database-backup.sql
```

Retrieving the Database Host, User Name, or Password

After you are finished installing Cloudera Manager, you can retrieve the database host, user name or password, if necessary, by reading the `/etc/cloudera-scm-server/db.properties` file:

```
# cat /etc/cloudera-scm-server/db.properties

Auto-generated by scm_prepare_database.sh
#
Sat Oct 1 12:19:15 PDT 201
#
com.cloudera.cmf.db.type=mysql
com.cloudera.cmf.db.host=localhost:7432
com.cloudera.cmf.db.name=scm
com.cloudera.cmf.db.user=scm
com.cloudera.cmf.db.password=TXqEESuhj5
```

Installing and Configuring Databases

Having installed MySQL, customized MySQL settings before starting MySQL, installed the JDBC connector, configured MySQL settings after starting the service, created the required databases, and backed up any desired database contents or information, proceed to [Installing CDH and Cloudera Manager](#).

Installing and Configuring an External PostgreSQL Database

Use the following instructions to install PostgreSQL and set up a database on the appropriate hosts. It's useful to set a password for the root user of PostgreSQL. Note the host name and port number where you install PostgreSQL because you will need to specify them when you install the JDBC connector to PostgreSQL in a later step. Note that PostgreSQL does not have an accepted default port. You must determine the port used in your environment. You will also need to create a PostgreSQL database and user account for Cloudera Manager to use to store data. See your PostgreSQL documentation for more information about installation and configuration.

To install PostgreSQL on a Red Hat system:

```
$ sudo yum install postgresql-server
```

To install PostgreSQL on a SLES system:

```
$ sudo zypper install postgresql-server
```

To install PostgreSQL on an Ubuntu system:

```
$ sudo apt-get install postgresql
```

Configuring Your Systems to Support PostgreSQL

You must configure the PostgreSQL database to run as expected. This involves:

- Configuring PostgreSQL to accept network connections.
- Initializing the database to work with Cloudera Manager.
- Configuring the operating system to start PostgreSQL.

Configuring PostgreSQL to accept network connections

By default, PostgreSQL only accepts connections on the loopback interface. Remember to reconfigure PostgreSQL to accept connections from the Fully Qualified Domain Name (FQDN) of the machines hosting the management roles. If you do not make these changes, the management processes will not be able to connect to and use the database on which they depend.

Initializing and configuring the external PostgreSQL database

1. Prepare the external PostgreSQL database for use with the Cloudera Manager Server.

- On Red Hat and SLES systems:

```
$ sudo service postgresql initdb
```

- On Debian/Ubuntu systems:

```
$ sudo /etc/init.d/postgresql initdb
```

2. Enable MD5 authentication. Edit `pg_hba.conf`, which is usually found in `/var/lib/pgsql/data` or `/etc/postgresql/8.4/main`. Add the following line:

```
host all all 127.0.0.1/32 md5
```

Add this line before another line in the configuration file that references ident authentication.

You can modify the contents of line to support different configurations. For example, if you want to access PostgreSQL from a different host, replace 127.0.0.1 with your IP address and update `postgresql.conf`, which is typically found in the same place as `pg_hba.conf` to include:

```
listen_addresses = '*'
```

3. Start the PostgreSQL database.

- On Red Hat and SLES systems:

```
$ sudo service postgresql start
```

- On Debian/Ubuntu systems:

```
$ sudo /etc/init.d/postgresql start
```

4. Configure the PostgreSQL server to start at boot.

- On Red Hat systems:

```
$ sudo /sbin/chkconfig postgresql on
$ sudo /sbin/chkconfig --list postgresql
postgresql          0:off  1:off  2:on   3:on   4:on
```

Installing and Configuring Databases

```
5:on      6:off
```

- On SLES systems:

```
$ sudo chkconfig --add postgresql
```

- On Debian/Ubuntu systems:

```
$ sudo chkconfig postgresql on
```

Creating the PostgreSQL Databases for Cloudera Manager

The next step involves creating databases and user accounts for all database-backed services in Cloudera Manager.

You must create databases for each of the following features that are part of the Management Services:

- Activity Monitor
- Service Monitor
- Report Manager
- Host Monitor

You can create these databases on the host where the Cloudera Manager Server will run, or on any other nodes in the cluster. For performance reasons, you should typically install each database on the host on which the service runs, as determined by the roles you will assign during installation or upgrade. In larger deployments or in cases where database administrators (DBAs) are managing the databases the services will use, databases may be separated from services, but do not undertake such an implementation lightly.

The examples that follow allow access only to a specific user ('amon_user' on 'myhost1'), ('smon_user' on 'myhost2'), ('repman_user' on 'myhost3'), or ('hmon_user' on 'myhost4') respectively) where 'myhost1', 'myhost2', 'myhost3', and 'myhost4' refer to the name of the host on which you are creating the database. To restrict access in this way, you must use the hostname if this host will also have the corresponding role (Activity Monitor, Service Monitor, Report Manager, or Host Monitor respectively) as Cloudera recommends. But if instead another host will have the corresponding role, and you want to allow access to the database only from that host, you must specify the fully-qualified domain name of that host.

If you later decide to move the role (for example, Activity Monitor) to another machine, you must grant access to the corresponding user (for example 'amon_user') on the new host as well.

Note the values you enter for database names, user names, and passwords. The Cloudera Manager installation wizard requires this information to correctly connect to these databases.

The database must be configured to support UTF-8 character set encoding. The sample commands below include the required options to enable UTF-8 support.

To create the PostgreSQL Databases for Cloudera Manager:

1. Connect to PostgreSQL.

```
$ sudo -u postgres psql
```

2. Create a database for the Activity Monitor feature and assign permissions to a database user. The database name, user name, and password can be anything you want.

```
postgres=# CREATE ROLE amon_user LOGIN PASSWORD 'amon_password';
postgres=# CREATE DATABASE activity_monitor OWNER amon_user ENCODING
'UTF8';
```

3. Create a database for the Service Monitor feature and assign permissions to a database user. The database name, user name, and password can be anything you want.

```
postgres=# CREATE ROLE smon_user LOGIN PASSWORD 'smon_password';
postgres=# CREATE DATABASE service_monitor OWNER smon_user ENCODING
'UTF8';
```

4. Create a database for the Report Manager feature and assign permissions to a database user. The database name, user name, and password can be anything you want.

```
postgres=# CREATE ROLE rman_user LOGIN PASSWORD 'rman_password';
postgres=# CREATE DATABASE reports_manager OWNER rman_user ENCODING
'UTF8';
```

5. Create a database for the Host Monitor feature and assign permissions to a database user. The database name, user name, and password can be anything you want.

```
postgres=# CREATE ROLE hmon_user LOGIN PASSWORD 'hmon_password';
postgres=# CREATE DATABASE host_monitor OWNER hmon_user ENCODING
'UTF8';
```

Configuring PostgreSQL Settings

There are several settings you should update to ensure your system performs as expected. Update these settings in the `/etc/postgresql.conf` file. Settings vary based on cluster size and resources.

Large Clusters

Large clusters may contain up to 1000 hosts. For large clusters consider the following suggestions as a starting point for settings.

- `max_connection`: For large clusters, each database is typically hosted on a different machine. The general rule is to allow each database on a host 100 maximum connections and then add 50 extra connections. As a result, in the normal case for large clusters, configure each of the five machines that hosts a single database for 150 connections. You may have to increase the system resources available to PostgreSQL, as described at <http://www.postgresql.org/docs/9.1/static/kernel-resources.html>.
- `shared_buffers`: 1024MB. Note that this requires that the operating system can allocate sufficient shared memory. See Postgres information on [Managing Kernel Resources](#) for more information on setting kernel resources.
- `wal_buffers`: 16MB. This value is derived from the `shared_buffers` value. Setting `wal_buffers` to be approximately 3% of `shared_buffers` up to a maximum of approximately 16MB works well in most case.
- `checkpoint_segments`: 128. The [PostgreSQL Tuning Guide](#) recommends values between 32 and 256 for write-intensive systems, such as this one.
- `checkpoint_completion_target`: 0.9. This setting is only available in PostgreSQL 8.3 and later. These versions are highly recommended.

Small to Mid-sized Clusters

For small to mid-sized clusters, consider the following suggestions as a starting point for settings. If resources are especially limited, consider reducing the buffer sizes and checkpoint segments further. Ongoing tuning may be required based on each machine's resource utilization. For example, if Cloudera Manager is running on the same machine as other roles, the following values may be acceptable:

- `shared_buffers`: 256MB
- `wal_buffers`: 8MB
- `checkpoint_segments`: 16
- `checkpoint_completion_target`: 0.9

Configuration Settings for Postgres 8.1

Cloudera recommends using PostgreSQL 8.4 or later. While more recent versions provide better results, earlier versions may be used. For example, Cloudera supports PostgreSQL 8.1, which is bundled with some older Linux distributions. If you use PostgreSQL 8.1, settings such as `checkpoint_completion_target` are not available. Consequently, consider using the following recommended settings:

- `shared_buffers: 131072`
- `wal_buffers: 4096`
- `checkpoint_segments: 256`

Note that because PostgreSQL 8.1 does not support entering parameters in MB, the preceding values are provided in buffers or segments. For example, each buffer is 8KB, so 131072 is equivalent to 1024 MB.

After updating database settings, you must restart PostgreSQL for the new settings to take effect.

Restarting PostgreSQL

After making database configuration changes, you must restart the database for the changes to be applied.

To restart PostgreSQL:

```
$ pg_ctl restart
```

Backing up the Databases

Cloudera recommends that you periodically back up the databases that Cloudera Manager uses to store configuration, monitoring, and reporting data. Be sure to back all of the databases you are using with Cloudera Manager:

- Cloudera Manager database: Contains all the information about what services you have configured, their role assignments, all configuration history, commands, users, and running processes. This is a relatively small database (<100MB), and is the most important to back up.
- Activity Monitor database: Contains information about past activities. In large clusters, this database can grow large.
- Service Monitor database: Contains monitoring information about daemons. In large clusters, this database can grow large.
- Report Manager database: Keeps track of disk utilization over time. Medium-sized.
- Host Manager database: Contains information about host status. Relatively small.

Installing and Configuring Databases

Backing Up the PostgreSQL Database

It's important that you periodically back up the external PostgreSQL database that Cloudera Manager uses to store configuration information.

To back up the PostgreSQL database, you can simply backup the `/var/lib/cloudera-scm-server-db` directory.

You can also use the `pg_dump` utility to back up the external PostgreSQL database.

To use the `pg_dump` utility:

1. Log in to the host where the Cloudera Manager Server is installed.
2. Run the following command as root:

```
cat /etc/cloudera-scm-server/db.properties.  
The db.properties file contains:  
# Auto-generated by scm_prepare_database.sh  
# Mon Jul 27 22:36:36 PDT 2011  
com.cloudera.cmf.db.type=postgresql  
com.cloudera.cmf.db.host=localhost:7432  
com.cloudera.cmf.db.name=scm  
com.cloudera.cmf.db.user=scm  
com.cloudera.cmf.db.password=NnYfWIjlbk
```

3. Run the following command as root:

```
# pg_dump -h localhost -p 7432 -U scm >  
/tmp/scm_server_db_backup.$(date +%Y%m%d)
```

4. Enter the password specified for the `com.cloudera.cmf.db.password` property on the last line of the `db.properties` file. Cloudera Manager generated the password for you during installation.

For more information about using the `pg_dump` utility, see this [page](#).

Installing an Embedded PostgreSQL Database

You can complete a manual installation that uses an embedded PostgreSQL server. If you are using [Installation Path A - Automated Installation by Cloudera Manager](#), an embedded PostgreSQL database is automatically installed, so you do not need to complete this procedure. If, however, you are using [Installation Path B - Installation Using Your Own Method](#), and you want to use an embedded PostgreSQL database, use this procedure.

Use this procedure to manually install PostgreSQL and set up an embedded PostgreSQL database on the appropriate hosts. Note the host name and port number where you install PostgreSQL because you will need to specify them when you install the JDBC connector. Note that PostgreSQL does not have an accepted default port. You must determine the port used in your environment. You will also need to

create a PostgreSQL database and user account for Cloudera Manager to use to store data. See your PostgreSQL documentation for more information about installation and configuration.

To install the embedded PostgreSQL database package on the Cloudera Manager Server host:

On a Red Hat system if you have a yum repo configured:

```
$ sudo yum install cloudera-manager-server-db
```

On a Red Hat system if you're transferring RPMs manually:

```
$ sudo yum --nogpgcheck localinstall cloudera-manager-server-db.noarch.rpm
```

On a SUSE system:

```
$ sudo zypper install cloudera-manager-server-db
```

On a Debian/Ubuntu system:

```
$ sudo apt-get install cloudera-manager-server-db
```

Using an Oracle Database

To use an Oracle database, several conditions must be met.

- You should collect information about the Oracle database you will use.
- You should install the Oracle JDBC.
- For larger CDH clusters, adjust Oracle settings or ask your DBA to do this for you.
- Ensure your Oracle database supports UTF8 character set encoding.

Collect Oracle Database Information

Installing, configuring, and maintaining an Oracle database should be completed by your organization's database administrator. In preparation for configuring Cloudera Manager to work with Oracle databases, gather the following information from your Oracle DBA:

- Host Name - The DNS name or the IP address of the host where the Oracle database is installed.
- SID - the name of the database that will store Cloudera Manager information. This database could contain schemas that would store information for the Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, and Host Monitor.
- User name - a user name for each schema that is storing information. This means you might have five unique usernames for the five schemas.

Installing and Configuring Databases

- Password - a password corresponding to each user name.

You will use the Oracle database information that you have gathered to configure the external database to work with the Cloudera Manager Server.

Install the JDBC Connector to Oracle

You must install the JDBC connector to Oracle on any host that connects from Cloudera Manager Server to database applications. This means you must install the connector on the Cloudera Manager Server host, as well as hosts to which you assign the Activity Monitor, Service Monitor, Report Manager, and Host Monitor roles.

Cloudera recommends that you assign roles and their corresponding databases to the same host. While putting roles and databases on the same host is recommended, it is not required. You could install a service, such as Activity Monitor on one host and install the corresponding database, such as the Activity Monitor database on a separate host. In such a case you would install the JDBC connector on the host running the Activity Monitor, not on the host with the Activity Monitor database.

You must download and install the `ojdbc6.jar` file, which contains the JDBC driver. There are different versions of the `ojdbc6.jar` file. You must download the version that is designed for:

- Java 6
- The Oracle database version used in your environment

For example, for an environment using Oracle 11g R2, the jar file can be downloaded from <http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-112010-090769.html>.

Copy the appropriate `ojdbc6.jar` file to `/usr/share/cm/lib` for all hosts running Cloudera Manager services and connecting to Oracle. This includes Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, and Host Monitor.

Adjust Oracle Settings to Accommodate Larger Clusters

Depending on the size of your deployments, your DBA may need to modify Oracle settings for monitoring services. Note that these guidelines are for larger clusters and do not apply to Cloudera Manager configuration database and to smaller clusters. Many factors contribute to whether to reconfigure your database settings, but in most cases, if your cluster has more than 100 hosts, you should consider making the following changes:

- Enable direct and asynchronous I/O by setting the `FILESYSTEMIO_OPTIONS` parameter to `SETALL`.
- Increase the RAM available to Oracle by changing the `MEMORY_TARGET` parameter. The amount of memory to assign depends on the size of Hadoop cluster.
- Create more redo log groups and spread the redo log members across separate disks/LUNs.
- Increase the size of redo log members to be at least 1 gigabyte.

Adjust Oracle System Settings for Sufficient Database Connectivity

Work with your Oracle database administrator to ensure appropriate values are applied for your Oracle database settings. You must determine the number of connections, transactions, and sessions to be allowed. Allow 100 maximum connections for each database and then add 50 extra connections. For example, for two databases set the maximum connections to 250. If you store all five databases on one host (the database for Activity Monitor, Service Monitor, Report Manager, Host Monitor databases, the Cloudera Manager Server database), set the maximum connections to 550.

From the maximum number of connections, you can determine the number of anticipated sessions using the following formula:

```
sessions = (1.1 * maximum_connections) + 5
```

For example, if a host has two databases, you anticipate 250 maximum connections. If you anticipate a maximum of 250 connections, plan for 280 sessions.

Once you know the number of sessions, you can determine the number of anticipated transactions using the following formula:

```
transactions = 1.1 * sessions
```

Continuing with the previous example, if you anticipate 280 sessions, you can plan for 308 transactions.

Work with your Oracle database administrator to apply these derived values to your system.

Using the sample values above, Oracle attributes would be set as follows:

```
alter system set processes=250;  
alter system set transactions=308;  
alter system set sessions=280;
```

Ensure your Oracle Database Supports UTF8

The database you use must be configured to support UTF8 character set encoding. One way your DBA might implement UTF8 character set encoding in Oracle databases is using the `dbca` utility. In such a case, when creating a database, the `characterSet AL32UTF8` option might be used to specify proper encoding. Consult with your DBA to ensure UTF8 encoding is properly configured.

Having collected information about your Oracle database, installed the Oracle JDBC, considered having database settings adjusted, and ensured UTF-8 encoding is enabled, proceed to [Install CDH and Cloudera Manager](#).

Installing CDH and Cloudera Manager

There are three paths to installing this version of CDH and Cloudera Manager:

- Installation Path A
- Installation Path B
- Upgrading

Important

Follow the instructions for only one of the installation paths.

Any database type may be used with any of the paths.

Installation Path A: Automated Installation by Cloudera Manager

In Installation Path A, Cloudera Manager can automate the installation of databases for Cloudera Manager and its services, the packages for CDH, Cloudera Manager, and the Oracle JDK. This option is available if your cluster deployment meets the following requirements:

- Uniform SSH access to cluster hosts on the same port from Cloudera Manager Server host.
- All hosts must have access to standard package repositories.
- All hosts must have access to the either `archive.cloudera.com` on the internet or to a local repository with the necessary installation files.

For Installation Path A instructions, click [here](#).

Installation Path B: Installation Using Your Own Method

Follow Installation Path B if your cluster does not meet the requirements for Installation Path A, or if you want or need to manage the installation of the CDH, Cloudera Manager, and the Oracle JDK packages yourself using whatever method you currently use to install software and configuration files on your cluster hosts.

For Installation Path B instructions, click [here](#).

Upgrading to Cloudera Manager 4.1

Follow one of the Upgrade Paths if your cluster already has an installation of a previous version of the Cloudera Management Suite or SCM Express Edition.

For upgrading instructions, see [Upgrading to Cloudera Manager 4.1](#).

Installation Path A - Automated Installation by Cloudera Manager

If your cluster meets the requirements for Installation Path A, follow the instructions in this section for automated installation by Cloudera Manager. The requirements for Path A are:

- Uniform SSH access to cluster hosts on the same port from Cloudera Manager Server host.
- All hosts must have access to standard package repositories.
- All hosts must have access to the either `archive.cloudera.com` on the internet or to a local repository with the necessary installation files.

The Cloudera Manager configuration, as well as the other monitoring and management information is stored in databases. As part of the process of Installation Path A, Cloudera Manager installs embedded PostgreSQL databases. It is simplest to use these automatically installed and configured databases. During the installation, you are provided with the option to select databases other than the automatically installed databases. If you intended to customize the installation to use other databases, install and configure them before beginning to use Installation Path A.

Using custom databases is a more advanced process, which is more often a part of an [Installation Using Your Own Method](#). For more information on installing custom databases, see [Installing and Configuring Databases](#). Otherwise, use the embedded PostgreSQL database, which the installer creates.

The general steps in this procedure for Installation Path A are:

- [Step 1: Download and Run the Cloudera Manager Installer](#)
- [Step 2: Start the Cloudera Manager Admin Console](#)
- [Step 3: Use Cloudera Manager for Automated CDH Installation and Configuration](#)
- [Step 4: Change the Default Administrator Password](#)
- [Step 5: Test the Installation](#)

Step 1: Download and Run the Cloudera Manager Installer

Important

For installation purposes, the Cloudera Manager Server must have SSH access to the cluster hosts and you must log in using a root account or an account that has password-less sudo permission. See [Requirements for Cloudera Manager](#) for more information.

Cloudera Manager accesses `archive.cloudera.com` by using yum on Red Hat systems, zypper on SUSE systems, or apt-get on Debian/Ubuntu systems. If your hosts access the Internet through an HTTP Proxy, you can configure yum, zypper, or apt-get, system-wide, to access `archive.cloudera.com` through a proxy. To do so, modify the system configuration on the Cloudera Manager Server host and on every cluster host where you want to install CDH. This is not required in all cases.

To configure your system to use a proxy

On Red Hat systems, add the following property to `/etc/yum.conf`:

```
http_proxy=http://server:port/
```

On SUSE systems, add the following property to `/root/.curlrc`:

```
--proxy=http://server:port/
```

On Debian/Ubuntu systems, add the following property to `/etc/apt/apt.conf`:

```
Acquire::http::Proxy "http://server:port";
```

To download and run the Cloudera Manager installer:

1. Download `cloudera-manager-installer.bin` from the [Cloudera Downloads page](#) to the host where you want to install the Cloudera Manager Server that is on your cluster or is accessible to your cluster over your network. Install Cloudera Manager on a single host.
2. After downloading `cloudera-manager-installer.bin`, change it to have executable permission.

```
$ chmod u+x cloudera-manager-installer.bin
```

3. Run `cloudera-manager-installer.bin`.

```
$ sudo ./cloudera-manager-installer.bin
```

4. Read the Cloudera Manager Readme and then press **Enter** to choose **Next**.
5. Read the Cloudera Manager License and then press **Enter** to choose **Next**. Use the arrow keys and press **Enter** to choose **Yes** to confirm you accept the license.
6. Read the Oracle Binary Code License Agreement and then press **Enter** to choose **Next**. Use the arrow keys and press **Enter** to choose **Yes** to confirm you accept the Oracle Binary Code License Agreement.

7. The Cloudera Manager installer begins installing the Oracle JDK and the Cloudera Manager repo files and then installs the packages. The installer also installs the Cloudera Manager Server.

Note

If an error message "Failed to start server" appears while running `cloudera-manager-installer.bin`, exit the installation program. If the Cloudera Manager Server log file `/var/log/cloudera-scm-server/cloudera-scm-server.log` contains the following message, then it's likely you have SELinux enabled:

```
Caused by: java.lang.ClassNotFoundException:
com.mysql.jdbc.Driver
    at java.net.URLClassLoader$1.run(Unknown Source)
    at java.security.AccessController.doPrivileged(Native
Method)
    at java.net.URLClassLoader.findClass(Unknown Source)
    at java.lang.ClassLoader.loadClass(Unknown Source)
    ...
```

You can disable SELinux by running the following command on the Cloudera Manager Server host:

```
$ sudo setenforce 0
```

To disable it permanently, edit `/etc/selinux/config`.

7. Note the complete URL provided for the Cloudera Manager Admin Console, including the port number, which is 7180 by default. Click **OK** to continue.
8. Click **OK** to exit the installer.

Note

If the installation is interrupted for some reason, you may need to clean up before you can re-run it. See [Uninstalling Cloudera Manager](#).

Step 2: Start the Cloudera Manager Admin Console

The Cloudera Manager Admin Console enables you to use Cloudera Manager to configure, manage, and monitor Hadoop on your cluster. Before using the Cloudera Manager Admin Console, gather information about the server's URL and port.

Installing CDH and Cloudera Manager

The server URL takes the following form:

```
http://<Server host>:<port>
```

<Server host> is the fully-qualified domain name or IP address of the host machine where the Cloudera Manager Server is installed.

<port> is the port configured for the Cloudera Manager Server. The default port is 7180.

For example, use a URL such as the following:

```
http://myhost.example.com:7180/
```

Cloudera Manager does not support changing the `admin` username for the installed account. You can change the password using Cloudera Manager after you run the wizard in the next section. While you cannot change the `admin` username, you can add a new user, assign administrative privileges to the new user, and then delete the default `admin` account.

To start the Cloudera Manager Admin Console:

1. In a web browser, enter the URL, including the port, for the Cloudera Server.
The login screen for Cloudera Manager appears.
2. Log into Cloudera Manager. The default credentials are:
Username: `admin`
Password: `admin`

Step 3: Use Cloudera Manager for Automated CDH Installation and Configuration

The following instructions show you how to use the Cloudera Manager wizard to do an initial installation and configuration. The wizard helps you to install and set up Cloudera packages across your cluster and will:

- Install and validate your Cloudera Manager License
- Find the cluster hosts you specify via hostname and IP-address ranges
- Connect to each host with SSH to install the Cloudera Manager Agent and CDH (including Hue)
- Install the Oracle JDK on the cluster hosts (if not already installed)
- Configure Hadoop automatically and start the Hadoop services

Important

- All hosts in the cluster must have access to either `archive.cloudera.com` on the internet or to a local repository with the necessary installation files.

To use Cloudera Manager:

1. The first time you start the Cloudera Manager Admin Console, the install wizard starts up.
2. Browse to your Cloudera Manager License file. If you don't install the license now, Cloudera Manager Free Edition will be installed.

Note

The instructions that follow assume you have installed a Cloudera Manager license. If you are not yet ready to install a Cloudera Manager license, and want to proceed with a Free Edition installation, stop here and use the [Cloudera Manager Free Edition Installation Guide](#) instead. If you install the Free Edition, and later need to upgrade to the full version of Cloudera Manager, follow the instructions under [Upgrading from Cloudera Manager Free Edition 4.1 to the Cloudera Manager Full Edition](#).

3. After you install the Cloudera Manager license, restart the Cloudera Manager server.

On Red Hat/CentOS/SUSE systems:

```
$ sudo service cloudera-scm-server restart
```

On Debian/Ubuntu systems:

```
$ sudo service cloudera-scm-server restart
```

4. After the Cloudera Manager server restarts, use your web browser to connect to the Cloudera Manager Admin Console URL again and log in, as described in [Step 2](#).

Note

After restarting the server, wait a few seconds for the server to finish initializing before you try to reconnect to the Admin Console.

5. Information is displayed indicating what the CDH installation includes. Click **Continue**.
6. To enable Cloudera Manager to automatically discover your cluster hosts where you want to install CDH, enter the cluster hostnames or IP addresses. You can also specify hostname and IP address ranges:

7. For example:

Use this Expansion Range	To Specify these Hosts
10.1.1.[1-4]	10.1.1.1, 10.1.1.2, 10.1.1.3, 10.1.1.4
host[1-3].company.com	host1.company.com, host2.company.com, host3.company.com
host[07-10].company.com	host07.company.com, host08.company.com, host09.company.com, host10.company.com

You can specify multiple addresses and address ranges by separating them by commas, semicolons, tabs, or blank spaces, or by placing them on separate lines. Use this technique to make more specific searches instead of searching overly wide ranges.

The scan results will include all addresses scanned, but only scans that reach hosts running SSH will be selected for inclusion in your cluster by default.

Note

If you don't know the IP addresses of all of the hosts, you can enter an address range that spans over unused addresses and then deselect the hosts that do not exist (and are not discovered) later in this procedure. However, keep in mind that wider ranges will require more time to scan.

7. Click **Search**.

Cloudera Manager identifies the hosts on your cluster to allow you to configure them for CDH. If there are a large number of hosts on your cluster, wait a few moments to allow them to be discovered and shown in the wizard. If the search is taking too long, you can stop the scan by clicking **Abort Scan**. To find additional hosts, add their host name or IP address and click **Search** again.

Note

Cloudera Manager scans hosts by checking for network connectivity. If there are some hosts where you want to install CDH that are not shown in the list, make sure you have network connectivity between the Cloudera Manager Server host and those hosts. Common causes of loss of connectivity are firewalls and interference from SELinux.

8. Verify that the number of hosts shown matches the number of hosts where you want to install CDH. Deselect host entries that do not exist and deselect the hosts where you do not want to install CDH. Click **Install CDH On Selected Hosts**.

9. Choose the CDH version to install.

- a. Select the major release of CDH to install. This is often CDH4.
- b. Select the specific release of CDH to install from within the major version you selected. You may choose a custom repository.
- c. Select the specific release of Impala to install on your hosts. You may choose either the latest version or use a custom repository.

Note

Impala works with CDH that is installed on RHEL/CentOS 6.2. You cannot install Impala on RHEL/CentOS 6.2 and then use that installation to query information stored on CDH installations that are not on RHEL/CentOS 6.2 systems. This means that if you have CDH deployed on operating systems such as Debian/Ubuntu, SuSE, or RHEL/CentOS 5.7, you cannot use Impala with that installation.

- d. Select the specific release of Cloudera Manager to install on your hosts. You may choose either the version that matches with the Cloudera Manager Server you are currently using or you can specify an installation at a custom repository.
- e. If you opted to use custom repositories for installation files, you must provide a GPG key URL that will apply for all repositories.
- f. Click **Continue**.

10. Provide credentials for authenticating with hosts.

- a. Select **root** or enter the user name for an account that has password-less sudo permissions.
- b. Select an authentication method.
 - If you choose to use password authentication, enter and confirm the password.
 - If you choose to use public-key authentication provide a passphrase and path to the required key files.
 - You can choose to specify an alternate ssh port. The default value is 22.
 - You can specify the maximum number of host installations to run at once. The default value is 10.

- c. Click **Start Installation** to begin installing CDH and the Cloudera Manager Agent on the cluster hosts.

The status of installation on each host is displayed in the following screen.

The Cloudera Manager wizard uses SSH to access the cluster hosts and follows a sequence of steps to download and install CDH and the Cloudera Manager Agent. The wizard configures package repositories, installs the Oracle JDK, CDH, and the Cloudera Manager Agent, and then starts the Cloudera Manager Agent. The wizard runs a maximum of 10 installations in parallel to avoid excessive network load. The status of installation on each host is displayed on the page that appears after you click **Start Installation**. You can also click the **Details** link for individual hosts to view detailed information about the installation and error messages if installation fails on any hosts.

Note

Clicking **Abort Installation** while installation is in progress halts any pending or in-progress installations and rolls back any in-progress installations to a clean state. Clicking **Abort Installation** does not affect completed or failed host installations.

If installation fails on a host, you can click the **Retry** link next to the failed host to try installation on that host again. To retry installation on all failed hosts, click **Retry Failed Hosts** at the bottom of the screen.

11. When the **Continue** button appears at the bottom of the screen, the installation process is completed.

If the installation has completed successfully on some hosts but failed on others, you can click **Continue** if you want to skip installation on the failed hosts and continue to the next screen to start configuring CDH on the successful hosts.

12. When you continue, the Host Inspector runs to validate the installation, and provides a summary of what it finds, including all the versions of the installed components. If the validation is successful, click **Continue**.
13. Choose the services you want to start on your cluster.
 - a. Choose which version of CDH to use.
 - b. Choose the combination of services to install: Core Hadoop, HBase Services, All Services, or Custom Services.

Note

Some services depend on others; for example, HBase requires HDFS and ZooKeeper.

Most of the combinations install MapReduce v1. Choose the custom option to install MapReduce v2 (YARN) or use the Add Service functionality to add YARN after installation completes.

- c. You can choose to enable the Cloudera Management Service.

Note

The Cloudera Management Services, which are added to each package, are Cloudera Manager processes that run to support monitoring and management features in Cloudera Manager.

- d. Click **Inspect Role Assignments** to see how the wizard will assign roles for the services you have chosen, and change them if you need to. These assignments are typically acceptable, but you can reassign services to nodes of your choosing, if desired.

The wizard evaluates the hardware configurations of the cluster hosts to determine the best machines for each role. For example, the wizard assigns the NameNode role to the machine that best meets the NameNode requirements. The wizard also configures other options, such as the number of map and reduce slots for TaskTracker, on the basis of the size of the cluster and the physical characteristics of each machines, such as the number of CPUs, amount of RAM, and disk space.

- e. Click **Continue** when you are satisfied with the assignments.

14. On the Database Setup page, configure settings for the Activity Monitor, Service Monitor, Report Manager, and Host Monitor databases.

- Establish database settings.
- Leave the default settings of **Use Embedded Database** to have Cloudera Manager create and configure all required databases.
- Select **Custom** to specify external databases.
- Click **Test Connection** to confirm that Cloudera Manager can communicate with the databases using the information you have supplied. This process takes two heartbeats to complete (about 30 seconds with the default heartbeat interval; much longer than you may expect).

If the test succeeds in all four cases, click **Continue**; otherwise check and correct the information you have provided for the databases and then try the test again.

15. Review Configuration Changes to be applied.

- a. Confirm the settings entered for file system paths. The file paths required vary based on the services to be installed. For example, you might confirm the NameNode Data Directory and the DataNode Data Directory for HDFS or confirm the TaskTracker Local Data Directory List or JobTracker Local Data Directory for MapReduce.
- b. Supply the name of the mail server (it can be `localhost`), the mail server user, and the mail recipients.

Note

The default configuration settings assume that you are using SMTPS (port 443) for your mail server. If that's not the case, you can change the configuration settings later in the `Alert Publisher` section of the `Management Services > Configuration` tab.

- c. Click **Continue**.

The wizard starts the services on your cluster.


16. When all of the services are started, click **Continue**.

17. Click **Continue**.

Step 4: Change the Default Administrator Password

As soon as possible after running the wizard and beginning to use Cloudera Manager, you should change the default administrator password.

To change the administrator password:

1. Click the gear icon  to display the **Administration** page.
2. Click the **Users** tab.
3. Click the **Change Password** button next to the **admin** account.
4. Enter a new password twice and then click **Submit**.

Step 5: Test the Installation

Now that you have finished with the CDH and Cloudera Manager installation, you are ready to test the installation. For testing instructions, see [Testing the Installation](#).

Note

If you change the hostname or port where the Cloudera Manager is running, or you enable TLS security, you must restart the Cloudera Management Services to update the URL to the Server. For instructions, see [Restarting a Service](#).

Installation Path B - Installation Using Your Own Method

To manage the installation of packages yourself, follow the instructions in this section. If you have already installed Cloudera Manager and CDH, skip this section and continue on to other installation tasks. For example, if you already have already completed a Path A installation, you might [Specify the Racks for Hosts](#) or [Test the Installation](#) next.

Note

If you use Puppet to install packages on your cluster, you can obtain the Puppet recipe for installing the packages for CDH and Cloudera Manager Server and Agent on this [site](#).

Before You Begin

Cloudera Manager and Cloudera Distribution of Hadoop (CDH) are comprised of a set of services. These services interact among each other and use databases to complete tasks. The parts that make up this system are very flexible, so you could deploy these services and resources in many different ways, though the process is greatly simplified by following Cloudera's installation and configuration guidelines.

Considering this, Cloudera recommends you begin by establishing a foundation of database resources that can be utilized as they become necessary throughout the installation process. Begin by deploying the necessary supporting services and then proceeding through the installation process.

Install the Oracle JDK

Install the Oracle Java Development Kit (JDK) on each of your cluster hosts where you want to run Hadoop before installing Cloudera's packages. Cloudera Manager can manage both CDH3 and CDH4 hosts, and the required JDK version varies accordingly.

- For installation instructions and recommendations for CDH3, see [Java Development Kit Installation for CDH3](#).
- For installation instructions and recommendations for CDH4, see [Java Development Kit Installation for CDH4](#).

Install Databases for the Cloudera Manager Services

Create and configure databases for the Cloudera Manager Activity Monitor, Service Monitor, Report Manager, and Host Monitor. Cloudera supports various database solutions including the PostgreSQL embedded database, PostgreSQL external databases, MySQL databases, or Oracle databases.

Information about how these databases are set up in your environment is required to complete the CDH and Cloudera Manager configuration. The details of what is required varies among database types. Gather this information either as you complete the installations or from database administrators who have the information required. A list of what information is required for each database type is provided in each database section.

Follow the instructions at [Installing and Configuring Databases](#) to complete this task.

Database choices	Notes and Instructions
Option A: Embedded PostgreSQL	This is the same PostgreSQL application and database that the Cloudera Manager wizard installs. For installation and configuration instructions, see

Database choices	Notes and Instructions
	Installing an Embedded PostgreSQL Database.
Option B: External PostgreSQL	After PostgreSQL is installed, you need to run a script to prepare a database for the Cloudera Manager Server as described in Installing and Configuring an External PostgreSQL Database.
Option C: External MySQL	You can use the same MySQL application that is used for the monitoring and reporting features, as described in Installing and Configuring a MySQL Database. After MySQL is installed, you need to run a script to prepare a database for the Cloudera Manager Server, as is described later in this topic.
Option D: External Oracle	You can use an external Oracle database for monitoring and reporting features, as described in Using an Oracle Database.

Establish Your Repository Strategy

Cloudera recommends installing products using package management tools such as `yum` for RedHat, `zypper` for SUSE, or `apt-get` for Debian/Ubuntu. These tools depend on access to repositories to install software. For example, Cloudera maintains Internet-accessible repositories for CDH and Cloudera Manager installation files. If you are installing CDH and Cloudera Manager to machines that do not have access to Cloudera repositories, consider creating your own internally hosted repository. For more information, see [Appendix A - Understanding Custom Installation Solutions.](#)

Step 1: Install CDH

This section describes how to install CDH on Red Hat, CentOS, SUSE, and Debian/Ubuntu systems, such as Ubuntu. This installation is done in preparation for using Cloudera Manager to configure and manage your cluster. For information about installing CDH, see the [CDH4 Installation Guide.](#)

Important

Cloudera Manager requires Hadoop to be installed on all hosts, but Hadoop must **not** be configured and must **not** be running.

Important

The Activity Monitor in Cloudera Manager 4.0 requires the `hue-plugins` package to be installed on the JobTracker host, regardless of whether you are using Hue. If you are using Hue, the `hue-plugins` package must be installed on all hosts.

1. Use **one** of the following commands to install packages on every host in your cluster:

For CDH4 and Impala On Red Hat/CentOS/Oracle systems:

```
$ sudo yum -y install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop
hadoop-hdfs hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client
hadoop-0.20-mapreduce hue-plugins hbase hive oozie oozie-client pig
zookeeper impala impala-shell
```

For CDH3 On Red Hat/CentOS/Oracle systems:

```
$ sudo yum -y install hadoop-0.20 hadoop-0.20-native.x86_64 hadoop-
0.20-sbin.x86_64 hue-plugins hadoop-zookeeper hadoop-hbase oozie
oozie-client
```

For CDH4 On SUSE systems:

```
$ sudo zypper install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop
hadoop-hdfs hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client
hadoop-0.20-mapreduce hue-plugins hbase hive oozie oozie-client pig
zookeeper
```

For CDH3 On SUSE systems:

```
$ sudo zypper install hadoop-0.20 hadoop-0.20-native.x86_64 hadoop-
0.20-sbin.x86_64 hue-plugins hadoop-zookeeper hadoop-hbase oozie
oozie-client
```

For CDH4 On Debian/Ubuntu systems:

```
$ sudo apt-get install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop
hadoop-hdfs hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client
hadoop-0.20-mapreduce hue-plugins hbase hive oozie oozie-client pig
zookeeper
```

For CDH3 On Debian/Ubuntu systems:

```
$ sudo apt-get install hadoop-0.20 hadoop-0.20-native.x86_64 hadoop-
0.20-sbin.x86_64 hue-plugins hadoop-zookeeper hadoop-hbase oozie
oozie-client
```

Installing CDH and Cloudera Manager

2. To install the `hue-common` package and all Hue applications on the Hue machine, install the `hue` meta-package:

On Red Hat/CentOS/Oracle systems:

```
$ sudo yum install hue
```

On SUSE systems:

```
$ sudo zypper install hue
```

On Debian/Ubuntu systems:

```
$ sudo apt-get install hue
```

3. Disable autostart for Hue on the Hue machine, and for Oozie on every machine on which it is installed.

For Redhat/CentOS/Oracle and SUSE systems:

```
$ sudo /sbin/chkconfig hue off  
$ sudo /sbin/chkconfig oozie off
```

For Debian/Ubuntu systems:

```
$ sudo update-rc.d -f hue remove  
$ sudo update-rc.d -f oozie remove
```

Step 2: Install the Cloudera Manager Server

Install the Cloudera Manager Server either on the machine where the database is installed, or on a machine that has access to the database. This machine need not be a host in the cluster that you want to manage with Cloudera Manager. The Cloudera Manager Server does not require CDH4 to be installed on the same machine.

On the Cloudera Manager Server machine, type the following commands to install the Cloudera Manager packages.

To install Cloudera Manager on a Red Hat system if you have a yum repo configured:

```
$ sudo yum install cloudera-manager-daemons  
$ sudo yum install cloudera-manager-server
```

To install Cloudera Manager on a Red Hat system if you're transferring RPMs manually:

```
$ sudo yum --nogpgcheck localinstall cloudera-manager-daemons-*.rpm
$ sudo yum --nogpgcheck localinstall cloudera-manager-server-*.rpm
```

To install Cloudera Manager Server on a SUSE system:

```
$ sudo zypper install cloudera-manager-daemons cloudera-manager-server
```

To install Cloudera Manager Server on a Debian/Ubuntu system:

```
$ sudo apt-get install cloudera-manager-daemons cloudera-manager-server
```

Step 3: Configure a Database for the Cloudera Manager Server

To manage the services, Cloudera Manager Agents, and configurations in your cluster, the Cloudera Manager Server stores data in a database. You can either use an existing database or install a new database. After installing the database, you must then run a script to prepare that database for use with the Cloudera Manager Server.

Note

The Cloudera Manager Server database is separate from the databases used by the Cloudera Manager Activity Monitor, Service Monitor, Report Manager, and Host Monitor. You installed these services' databases in [the prerequisites](#).

In this release, you can use any of the database options listed in the table below.

Important

Cloudera Manager uses only one database to store configuration data, so you do not need to complete all options listed below. After establishing one database for Cloudera Manager, move onto the next steps. Do not install all the database options.

Using the Embedded PostgreSQL Database for the Cloudera Manager Server

To use the embedded PostgreSQL database:

1. Prepare the embedded PostgreSQL database for use with the Cloudera Manager Server by running this command:

```
$ sudo service cloudera-scm-server-db initdb
```

2. Start the embedded PostgreSQL database by running this command:

```
$ sudo service cloudera-scm-server-db start
```

The Cloudera Manager Server can now use the embedded PostgreSQL database. You can skip to the next step, [Install the Cloudera Manager Agents](#).

Preparing the Database for the Cloudera Manager Server

Cloudera Manager configuration can be completed using the `scm_prepare_database.sh` script, which is installed in the `/usr/share/cmfschema` directory on the host where the Cloudera Manager Server package is installed. You must run the script on the Cloudera Manager Server host.

After you have installed your database application or collected information about an existing Oracle installation, use the `scm_prepare_database.sh` script to prepare the database for use with the Cloudera Manager Server. This script enables Cloudera Manager Server to connect to an external database in MySQL, PostgreSQL, or Oracle. The script prepares the database by:

- Creating the Cloudera Manager Server database configuration file.
- Creating a database for the Cloudera Manager Server to use. This is optional and is only completed if options are specified.
- Setting up a user account for the Cloudera Manager Server. This is optional and is only completed if options are specified.

Script syntax

```
scm_prepare_database.sh database-type [options] database-name username  
password
```

Required Parameter	Description
database-type	To connect to a MySQL database, specify <code>mysql</code> as the database type. To connect to an Oracle database, specify <code>oracle</code> . To connect to an external PostgreSQL database, specify <code>postgresql</code> .
database-name	The name of the Cloudera Manager Server database you want to create.
username	The username for the Cloudera Manager Server database you want to create.
password	The password for the Cloudera Manager Server database you want to create. If you don't specify the password on the command line, the script will prompt

Required Parameter	Description
	you to enter it.

Option	Description
-h or --host	The IP address or hostname of the host where MySQL or Oracle is installed. The default is to use the local host.
-P or --port	The port number to use to connect to MySQL or Oracle. The default port is 3306. This option is used for a remote connection only.
-u or --user	The username for the MySQL or Oracle application. The default is <code>root</code> .
-p or --password	The password for the MySQL or Oracle application. The default is no password.
--scm-host	The hostname where the Cloudera Manager Server is installed. Omit if the Cloudera Manager server and MySQL or Oracle are installed on the same host.
--config-path	The path to the Cloudera Manager Server configuration files. The default is <code>/etc/cloudera-scm-server</code> .
--schema-path	The path to the Cloudera Manager schema files. The default is <code>/usr/share/cmf/schema</code> (the location of the script).
-f	The script will not stop if an error is encountered.
-? or --help	Display help.

Note

You can also run `scm_prepare_database.sh` without options to see the syntax.

Example 1: Running the script when MySQL is installed on another host

This example explains how to run the script on the Cloudera Manager Server machine (myhost2) and create and use a temporary MySQL user account to connect to MySQL remotely on the MySQL machine (myhost1).

1. On myhost1's MySQL prompt, create a temporary user who can connect from myhost2:

```
mysql> grant all on *.* to 'temp'@'%' identified by 'temp' with grant option;
Query OK, 0 rows affected (0.00 sec)
```

2. On the Cloudera Manager Server host (myhost2), run the script:

```
$ sudo /usr/share/cmf/schema/scm_prepare_database.sh mysql -h
myhost1.sf.cloudera.com -u temp -ptemp --scm-host
myhost2.sf.cloudera.com scm scm scm
Looking for MySQL binary
Looking for schema files in /usr/share/cmf/schema
Verifying that we can write to /etc/cloudera-scm-server
Connecting to mysql at myhost1 as 'temp'
Creating Cloudera Manager database 'scm'
Setting up Cloudera Manager user 'scm'@'myhost2.sf.cloudera.com'
Installing Cloudera Manager schema from file
/usr/share/cmf/schema/cmf_schema_00001.ddl
Installing Cloudera Manager schema from file
/usr/share/cmf/schema/cmf_schema_00002.ddl
Creating Cloudera Manager configuration file in /etc/cloudera-scm-
server
All done, your Cloudera Manager database is ready to go!
```

3. On myhost1, delete the temporary user:

```
mysql> drop user 'temp'@'%;
Query OK, 0 rows affected (0.00 sec)
```

Example 2: Running the script to configure Oracle

This shows an example of running the script to configure an Oracle database.

```
[root@rhel55-6 ~]# /usr/share/cmf/schema/scm_prepare_database.sh -h cm-
oracle.example.com oracle orcl sample_user sample_pass
Verifying that we can write to /etc/cloudera-scm-server
Creating SCM configuration file in /etc/cloudera-scm-server
Executing: /usr/java/jdk1.6.0_31/bin/java -cp /usr/share/java/mysql-
connector-java.jar:/usr/share/cmf/schema/./lib/*
```



```
com.cloudera.enterprise.dbutil.DbCommandExecutor /etc/cloudera-scm-
server/db.properties com.cloudera.cmf.db.
[ main] DbCommandExecutor INFO Successfully connected to database.
All done, your SCM database is configured correctly!
```

Example 3: Running the script when PostgreSQL is collocated with the Cloudera Manager Server

This example explains how to run the script on the Cloudera Manager Server machine when you have installed PostgreSQL on the same machine.

```
$ /usr/share/cmf/schema/scm_prepare_database.sh postgresql -u temp -ptemp
scm scm scm
```

Step 4: Install the Cloudera Manager Agents

Important

It is recommended that you install CDH4 before installing the Cloudera Manager Agents.

In this step, you will install the Cloudera Manager Agents and the `/etc/cloudera-scm-agent/config.ini` configuration file on every machine in your cluster that you want to manage using Cloudera Manager. You can use whatever method you currently use to install software and configuration files on your cluster nodes.

1. On every Cloudera Manager Agent host machine (including those that will run one or more of the Cloudera Manager Services: Service Monitor, Activity Monitor, Event Server, Alert Publisher, Report Manager), use the following commands to install the Cloudera Manager packages:

To install the Cloudera Manager Agent and Services on a Red Hat system if you have a yum repo configured:

```
$ sudo yum install cloudera-manager-agent cloudera-manager-daemons
```

To install the Cloudera Manager Agent and Services on a Red Hat system if you're transferring RPMs manually:

```
$ sudo yum --nogpgcheck localinstall cloudera-manager-agent-
package.*.x86_64.rpm cloudera-manager-daemons
```

To install the Cloudera Manager Agent and Services on a SUSE system:

```
$ sudo zypper install cloudera-manager-agent cloudera-manager-daemons
```

To install the Cloudera Manager Agent and Services on a Debian/Ubuntu system:

```
$ sudo apt-get install cloudera-manager-agent cloudera-manager-daemons
```

Note

The `cloudera-manager-daemons` package is required on the systems that will run the Service Monitor, Activity Monitor, Event Server, Alert Publisher, and Report Manager services. It is optional on the other systems, but if you decide not to install it on those other systems, and you later decide to move the services to different systems, you will need to remember to install the `cloudera-manager-daemons` package on those systems then.

2. On every Cloudera Manager Agent host machine, configure the Cloudera Manager Agent to point to the Cloudera Manager Server by setting the following properties in the `/etc/cloudera-scm-agent/config.ini` configuration file:

Property	Description
<code>server_host</code>	Name of host machine where the Server is running
<code>server_port</code>	Port on host machine where the Server is running

Note

The Cloudera Manager Agent configures its hostname automatically. However, if your cluster machines are multi-homed (that is, they have more than one hostname), and you want to specify which hostname the Cloudera Manager Agent uses, you can update the `listening_hostname=` property in the `/etc/cloudera-scm-agent/config.ini` configuration file on the cluster machines. If you want to specify which IP address the Cloudera Manager Agent uses, you can update the `listening_ip=` property in the same file.

Step 5: Start the Cloudera Manager Server

Important

When you start the Cloudera Manager Server and Agents, Cloudera Manager assumes you are not already running HDFS and MapReduce. If you are, shut down HDFS and MapReduce (`service hadoop-0.20-<daemon> stop`), and configure the init scripts to not start on boot (for example, `chkconfig hadoop-0.20-<daemon> off`). Contact Cloudera Support for help converting your existing Hadoop configurations for use with Cloudera Manager.

To start the Cloudera Manager Server:

1. To start the Cloudera Manager Server, type this command on the Cloudera Manager Server machine:

```
$ sudo service cloudera-scm-server start
```

2. If you have problems starting the Server, such as database permissions problems, you can use the Server's log `/var/log/cloudera-scm-server/cloudera-scm-server.log` to troubleshoot the problem.

Note

If the Server fails to start, and you are using MySQL to store information about service configuration, check that the InnoDB engine is configured, not the MyISAM engine; the server will not start if its tables are configured with the MyISAM engine, and an error such as the following will appear in the log file:

```
Tables ... have unsupported engine type ... . InnoDB is
required.
```

For more information, see [Installing and Configuring a MySQL Database](#).

Step 6: Start the Cloudera Manager Agents

To start the Cloudera Manager Agents:

1. To start the Cloudera Manager Agent, run this command on each Agent machine:

```
$ sudo service cloudera-scm-agent start
```

When the Agent starts up, it contacts the Cloudera Manager Server. When the Agent machines reboot, `cloudera-scm-agent` will start automatically.

Troubleshooting Cloudera Manager Agent Connection Problems

If there is a communication failure between a Cloudera Manager Agent and Cloudera Manager Server, you can use the Cloudera Manager Server log file `/var/log/cloudera-scm-server/cloudera-scm-server.log` and the Cloudera Manager Agent log files `/var/log/cloudera-scm-agent/cloudera-scm-agent.log` to troubleshoot the problem. The following is a common error.

Error message	Description
<code>error: (113, 'No route to host')</code> in <code>cloudera-scm-agent.log</code> .	This indicates that the agent is unable to connect to the Cloudera Manager Server. Make sure that <code>iptables</code> and <code>SELinux</code> are both turned off.

Step 7: Start the Cloudera Manager Admin Console

The Cloudera Manager Admin Console enables you to use Cloudera Manager to configure, manage, and monitor Hadoop on your cluster. Before using the Cloudera Manager Admin Console, gather information about the server's URL and port.

The server URL takes the following form:

```
http://<Server host>:<port>
```

`<Server host>` is the fully-qualified domain name or IP address of the host machine where the Cloudera Manager Server is installed.

`<port>` is the port configured for the Cloudera Manager Server. The default port is 7180.

For example, use a URL such as the following:

```
http://myhost.example.com:7180/
```

Cloudera Manager does not support changing the `admin` username for the installed account. You can change the password using Cloudera Manager after you run the wizard in the next section. While you cannot change the `admin` username, you can add a new user, assign administrative privileges to the new user, and then delete the default `admin` account.

To start the Cloudera Manager Admin Console:

1. In a web browser, enter the URL, including the port, for the Cloudera Server.
The login screen for Cloudera Manager appears.
2. Log into Cloudera Manager. The default credentials are:
Username: `admin`
Password: `admin`

Step 8: Configure Services

The following instructions describe how to use the Cloudera Manager wizard to configure and start the Hadoop services.

To configure services:

1. When you start the Cloudera Manager Admin Console, the install wizard starts up. Click **Continue** to get started.
2. Browse to your Cloudera Manager License file. If you don't install it now, Cloudera Manager Free Edition will be installed.

Note

The instructions that follow assume you have installed a Cloudera Manager license. If you are not yet ready to install a Cloudera Manager license, and want to proceed with a Free Edition installation, stop here and use the [Cloudera Manager Free Edition Installation Guide](#) instead. If you install the Free Edition, and later need to upgrade to the full version of Cloudera Manager, follow the instructions under [Upgrade from Cloudera Manager Free Edition 4 to Cloudera Manager 4](#).

3. After you install the Cloudera Manager license, you must restart the Cloudera Manager server. From the command line, enter:

```
$ sudo service cloudera-scm-server restart
```

4. After the Cloudera Manager server restarts, log in again.

Note

After restarting the server, wait a few seconds for the server to finish initializing before you try to reconnect to the Admin Console.

5. Click **Continue** in the next screen.
6. On the "Specify hosts..." page, verify that the hosts you previously [configured](#) are reported as being managed. Then click **Skip Host Installation** (you have already installed CDH4 and Cloudera Manager components).
7. Choose the Hadoop services you want to start. You can choose one of the standard combinations: Core Hadoop, HBase Services, or All Services; these combinations take into account the dependencies between the Hadoop services.

Alternatively, you can choose Custom Services, and select the services individually.

Note

Some services depend on others; for example, HBase requires HDFS and ZooKeeper.

The Cloudera Management Services, which are added to each package, are Cloudera Manager processes that run to support monitoring and management features in Cloudera Manager.

8. On the Database Setup page, enter the information requested. If the installation you are upgrading includes existing roles, those roles will not require configuration information. At most you will need to provide information for up to the Activity Monitor, Service Monitor, Report Manager, and Host Monitor databases.

Important

The value you enter as the database hostname **must** match the value you entered for the hostname (if any) when you created the database (see [Installing and Configuring Databases](#)).

For example, if you entered the following for the Activity Monitor database

```
mysql> grant all on activity_monitor.* TO 'amon_user'@'localhost'
IDENTIFIED BY 'amon_password';
```

the value you enter here for the database hostname must be `localhost`.

On the other hand, if you had entered the following when you created the database

```
mysql> grant all on activity_monitor.* TO
'amon_user'@'myhost1.myco.com' IDENTIFIED BY 'amon_password';
```

the value you enter here for the database hostname must be `myhost1.myco.com`.

If you did not specify a host, or used a wildcard to allow access from any host, you can enter either the fully-qualified domain name (FQDN) here, or `localhost`. For example, if you entered

```
mysql> grant all on activity_monitor.* TO 'amon_user'@'%' IDENTIFIED
BY 'amon_password';
```

the value you enter here for the database hostname can be either the FQDN or `localhost`.

Similarly, if you entered

```
mysql> grant all on activity_monitor.* TO 'amon_user' IDENTIFIED BY
'amon_password';
```

the value you enter here for the database hostname can be either the FQDN or `localhost`.

9. Click **Test Connection** to confirm that Cloudera Manager can communicate with the databases using the information you have supplied. This atypical transaction takes two heartbeats to complete (about 30 seconds with the default heartbeat interval).


If the test succeeds in all cases, click **Continue**; otherwise check and correct the information you have provided for the databases and then try the test again.

10. Confirm the settings entered for file system paths, such as the NameNode Data Directory and the DataNode Data Directory.
11. Supply the name of the mail server (it can be `localhost`), the mail server user, and the mail recipients.
12. Click **Continue**.
The wizard starts the services on your cluster.
13. When all of the services are started, click **Continue**.
14. In the final screen, you can read instructions for generating a client configuration to allow users work with the HDFS, MapReduce, HBase, or other services you created. The procedure is also described in the [Generating a Client Configuration](#) section.
15. Click **Continue**.

Step 9: Change the Default Administrator Password

As soon as possible after running the wizard and beginning to use Cloudera Manager, you should change the default administrator password.

To change the administrator password:

1. Click the gear icon  to display the **Administration** page.
2. Click the **Users** tab.
3. Click the **Change Password** button next to the **admin** account.
4. Enter a new password twice and then click **Submit**.

Step 10: Test the Installation

Now that you have finished the CDH4 and Cloudera Manager installation, you are ready to test the installation. For testing instructions, see [Testing the Installation](#).

Note

If you change the hostname or port where the Cloudera Manager is running, or you enable TLS security, you must restart the Cloudera Management Services to update the URL to the Server. For instructions, see [Restarting a Service](#).

Installing Impala with Cloudera Manager

Step 1: Install CDH and Hive

To use Cloudera Impala, you must install the CDH, Hive, and Impala. Install CDH, Hive and Impala on the nodes that will run Impala. Use **only one** of the following ways to deploy CDH, Hive, and Impala:

- [Installation Path A - Automated Installation by Cloudera Manager](#): Installs Cloudera Manager, CDH, and Impala as part of the process of using the pre-packaged installer.
- [Installation Path B - Installation Using Your Own Method](#): Installs Cloudera Manager, CDH, and Impala, specifying each package individually using package management tools.

Step 2: Install a Database for the Hive Metastore

Install a MySQL database to use for the Hive metastore. Install this database on a single machine in your cluster. Impala does not support the embedded Derby database.

To install a MySQL database

1. Install the MySQL server.

```
$ sudo yum install mysql-server
```

After issuing the command to install MySQL, you may need to respond to prompts to confirm that you do want to complete the installation.

2. After installation completes, start the mysql daemon.

```
$ sudo service mysqld start
```

3. Configure the MySQL server to start at boot.

```
$ sudo /sbin/chkconfig mysqld on
$ sudo /sbin/chkconfig --list mysqld
mysqld          0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

To configure MySQL

In the following procedure, your current `root` password is blank. Press the Enter key when you're prompted for the root password.

1. Set the MySQL root password:

```
$ sudo /usr/bin/mysql_secure_installation
```



```
[...]
Enter current password for root (enter for none):
OK, successfully used password, moving on...
[...]
Set root password? [Y/n] y
New password:
Re-enter new password:
Remove anonymous users? [Y/n] Y
[...]
Disallow root login remotely? [Y/n] N
[...]
Remove test database and access to it [Y/n] Y
[...]
Reload privilege tables now? [Y/n] Y
All done!
```

2. Configure MySQL server to start at boot:

```
$ sudo /sbin/chkconfig mysqld on
$ sudo /sbin/chkconfig --list mysqld
mysqld          0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

Step 3: Configure the Remote Database as the Hive Metastore

The recommended production environment for Hive is to use a database on one or more remote servers as the metastore, and MySQL is the most popular database to use. To set this up:

- Configure the remote MySQL database to recognize Hive
- Configure Hive to connect to the remote MySQL database

Impala does not support Derby.

Configuring Remote MySQL Database

Before you can run the Hive metastore with a remote MySQL database, you must configure a connector to the remote MySQL database, set up the initial database schema, and configure the MySQL user account for the Hive user.

Install the [MySQL JDBC Connector](#) in the Hive lib directory:

```
$ curl -L 'http://www.mysql.com/get/Downloads/Connector-J/mysql-connector-
java-5.1.22.tar.gz/from/http://mysql.he.net/' | tar xz
$ sudo cp mysql-connector-java-5.1.22/mysql-connector-java-5.1.22-bin.jar
/usr/lib/hive/lib/
```

Installing Impala with Cloudera Manager

The MySQL administrator should create the initial database schema using the `hive-schema-0.9.0.mysql.sql` file located in the `/usr/lib/hive/scripts/metastore/upgrade/mysql` directory.

```
$ mysql -u root -p
mysql> CREATE DATABASE hivemetastoredb;
mysql> USE hivemetastoredb;
mysql> SOURCE /usr/lib/hive/scripts/metastore/upgrade/mysql/hive-schema-0.9.0.mysql.sql;
mysql> CREATE USER 'hive'@'%' IDENTIFIED BY 'hive';
mysql> GRANT ALL PRIVILEGES ON hivemetastoredb.* TO 'hive'@'%' WITH GRANT OPTION;
mysql> FLUSH PRIVILEGES;
mysql> quit;
```

Step 4: Add the Impala Service

Complete the process of configuring Impala so it uses the Hive metastore and any other settings issues are resolved. Take note of the settings you applied to the remote MySQL database. You will need these values, such as the database name and database user name as you configure the Cloudera Service you add.

As you configure Impala, you will need to modify HDFS and Impala settings. Configurations that are recommended for many environments are as follows:

HDFS Configurations

Property	Value
DataNode Local Path Access Users dfs.block.local-path-access.user	impala
DataNode Data Directory Permissions dfs.datanode.data.dir.perm	755
Enable HDFS Block Metadata API dfs.datanode.hdfs-blocks-metadata.enabled	true
Enable HDFS Short Circuit Read dfs.client.read.shortcircuit	true

Impala Configurations

Property	Value
Hive Metastore Database Type	mysql
Hive Metastore Database Name	metastore
Hive Metastore Database Host	<db hostname>
Hive Metastore Database Port	3306
Hive Metastore Database User javax.jdo.option.ConnectionUserName	hive
Hive Metastore Database Password javax.jdo.option.ConnectionPassword	Hive

Add the Impala service and update the configuration as described in [Adding the Cloudera Impala Service](#) in the Cloudera Manager User Guide.

Upgrading to Cloudera Manager 4.1

You can upgrade existing Cloudera Manager and Cloudera's Distribution Including Apache Hadoop (CDH) to this version. Upgrading preserves existing data and settings, while enabling the use of the new features provided with the latest product versions. To enable new features, some new settings are added, but nothing is removed.

Understanding Upgrades

The process for upgrading to Cloudera Manager varies based on the starting point. The categories of tasks to be completed include the following:

- Install any databases that are newly required for this release.
- Upgrade Cloudera Manager.
- Upgrade hosts in the cluster.

Before Upgrading

- The Cloudera Manager Server must have SSH access to the cluster hosts and you must log in using a root account or an account that has password-less sudo permission. See [Requirements for Cloudera Manager](#) for more information.

Upgrading to Cloudera Manager 4.1

- Ensure there are no running commands. Use the Admin Console's main navigation bar to check for any running commands. You can either wait for commands to complete or abort any running commands. For more information on viewing and aborting running commands, see [Viewing Running and Recent Commands](#).
- Ensure you have completed any required process for preparing databases, as described in [Database Considerations for Cloudera Manager Upgrades](#).

During the Upgrade

During the upgrade process, the following changes occur:

- The database schemas are modified for any databases storing information for Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, and Host Monitor.
- Configuration information is reorganized.

After Upgrading

After completing an upgrade to 4.1, the following is true:

- You should deploy client configurations to ensure client services have the most current configuration.
- Required databases are established to store information for Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, and Host Monitor.
- The database schemas reflect the current version.
- The Host Monitor service is added and active.
- The Cloudera Manager Server and all supporting services, such as the Activity Monitor, Service Monitor, Report Manager, and Host Monitor are updated.

Upgrade Paths

In some cases, completing an upgrade requires changes to your environment, and in other cases, elements are already in place. For example, if you are upgrading your environment from 3.7 to 4.1, you must add Host Monitor, but if you are upgrading from 4.0 beta or 4.0 GA to 4.1, this is not required, as Host Monitor is included in 4.0. The specific steps required vary based on the path taken.

To upgrade from an older version of Cloudera Manager, begin by upgrading to 3.7, and then proceed to upgrade from there.

Warning

Cloudera Manager 4.1 works with CDH3 and CDH4, but does not work with CDH4.0 beta. You must upgrade any installations of CDH4.0 beta.

Begin the upgrade process by evaluating [Database Considerations for Cloudera Manager Upgrades](#).

Database Considerations for Cloudera Manager Upgrades

Cloudera Manager uses databases to store information about system configurations and tasks. Before upgrading, complete the pre-upgrade database tasks that apply in your environment.

After you have completed these steps, the upgrade processes automatically complete any additional updates to database schemas and service data stored. You do not need to complete any data migration.

Back up Databases

Before beginning the upgrade process, Cloudera recommends you back up all databases. This is especially important if you are upgrading from 3.7.x and there is any possibility you may want to revert to using 3.7.x. For information on backing up databases:

- For MySQL, see [Backing up the MySQL Database](#).
- For PostgreSQL, see [Backing up the PostgreSQL Database](#).
- For Oracle, work with your database administrator to ensure databases are properly backed up.

If any additional database will be required as a result of the upgrade, complete any required preparatory work to install and configure those databases. For example, Cloudera Manager 4.0 offers a Host Monitoring service that requires a database. To enable the Host Monitoring service, you must install a database. The upgrade instructions assume all required databases have been prepared. For more information on using databases, see [Installing and Configuring Databases](#). Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, and Host Monitor all require databases.

Modify Databases to Support UTF-8

Cloudera Manager 4.0 adds support for UTF-8 character sets. Update any existing databases in your environment that are not configured to support UTF-8.

Modifying MySQL to Support UTF-8

To modify a MySQL database to support UTF-8, the default character set must be changed and then you must restart the mysql service. Use the following commands to complete these tasks:

```
mysql> alter database default character set utf8;
mysql> quit
$ sudo service mysql restart
```

Modifying PostgreSQL to Support UTF-8

There is no single command available to modify an existing PostgreSQL database to support UTF-8. As a result, you must complete the following process:

1. Use `pg_dump` to export the database to a file.
This creates a backup of the database that you will import into a new, empty database that supports UTF-8.

Upgrading to Cloudera Manager 4.1

2. Drop the existing database.
This deletes the existing database.
3. Create a new database that supports unicode encoding and that has the same name as the old database. Use a command of the following form, replacing the database name and user name with values that match your environment:

```
CREATE DATABASE scm_database WITH OWNER scm_user ENCODING 'UTF8'
```

4. Review the contents of the exported database for non-standard characters. If you find unexpected characters, modify these so the database backup file contains the expected data.
5. Import the database backup to the newly created database.

Modifying Oracle to Support UTF-8

Work with your Oracle database administrator to ensure any Oracle databases support UTF-8.

Modify MySQL Databases to Support Larger Thread Stacks

If you are upgrading a deployment that uses MySQL databases, you must set the `thread_stack` value in `my.cnf` to be 256KB. The `my.cnf` file is normally located in `/etc` or `/etc/mysql`.

After modifying the `thread_stack` value, you must restart the `mysql` service:

```
$ sudo service mysql restart
```

Modify Databases to Support Appropriate Maximum Connections

Check existing databases configurations to ensure the proper maximum number of connections is supported. Update the maximum configuration values, as required.

Modifying the Maximum number of MySQL Connections

Allow 100 maximum connections for each database and then add 50 extra connections. For example, for two databases set the maximum connections to 250. If you store all five databases on one host (the database for Activity Monitor, Service Monitor, Report Manager, Host Monitor databases, the Cloudera Manager Server database), set the maximum connections to 550.

Modifying the Maximum number of PostgreSQL Connections

Update the `max_connection` parameter in the `/etc/postgresql.conf` file.

You may have to increase the system resources available to PostgreSQL, as described at <http://www.postgresql.org/docs/9.1/static/kernel-resources.html>.

Modifying the Maximum number of Oracle Connections

Work with your Oracle database administrator to ensure appropriate values are applied for your Oracle database settings. You must determine the number of connections, transactions, and sessions to be allowed. Allow 100 maximum connections for each database and then add 50 extra connections. For example, for two databases set the maximum connections to 250. If you store all five databases on one host (the database for Activity Monitor, Service Monitor, Report Manager, Host Monitor databases, the Cloudera Manager Server database), set the maximum connections to 550.

From the maximum number of connections, you can determine the number of anticipated sessions using the following formula:

```
sessions = (1.1 * maximum_connections) + 5
```

For example, if a host has two databases, you anticipate 250 maximum connections. If you anticipate a maximum of 250 connections, plan for 280 sessions.

Once you know the number of sessions, you can determine the number of anticipated transactions using the following formula:

```
transactions = 1.1 * sessions
```

Continuing with the previous example, if you anticipate 280 sessions, you can plan for 308 transactions.

Work with your Oracle database administrator to apply these derived values to your system.

Using the sample values above, Oracle attributes would be set as follows:

```
alter system set processes=250;  
alter system set transactions=308;  
alter system set sessions=280;
```

Next Steps

After you have completed any required database preparatory tasks, continue to the upgrade path that is appropriate for your environment. Supported paths include:

- [Upgrade from Cloudera Manager 3.7.x to Cloudera Manager 4.1](#)
- [Upgrade from Cloudera Manager 4 to the Latest Cloudera Manager 4](#)
- [Upgrade from Cloudera Manager Free Edition 4 to Cloudera Manager 4](#)

Upgrading to Cloudera Manager 4.1

Upgrade from Cloudera Manager 3.7.x to Cloudera Manager 4.1

Summary: What You are Going to Do

Upgrading from Cloudera Manager 3.7.x to Cloudera Manager 4.1 involves the following broad steps:

[Upgrade Cloudera Manager Server](#) - Stop Cloudera Manager services, copy files to the Cloudera Manager server, upgrade the server, and restart services.

[Upgrade the Cluster Hosts](#) - Use the upgrade wizard to upgrade hosts in the cluster.

[Deploy Updated Client Configurations](#) - Update client configurations to ensure clients operate as expected with the upgraded systems.

[Verify the Upgrade](#) - You can choose to check the versions of installed components.

[Upgrade CDH](#) - You may choose to upgrade CDH installations. Cloudera Manager 4 can manage both CDH 3 and CDH 4.

Before beginning the upgrade, follow the guidelines described in [Database Considerations for Cloudera Manager Upgrades](#).

After completing the upgrade from Cloudera Manager 3.7.x to Cloudera Manager 4.1, as described in this topic, all required updates to database schemas and service data is completed automatically. You do not need to complete any additional database updates or data migration.

Upgrade Cloudera Manager Server

This process involves stopping running Cloudera Manager service, downloading and applying updates to Cloudera Manager, and restarting the Cloudera Manager service. Valid licenses from Cloudera Manager 3.7.x continue to work with Cloudera Manager 4.

The default name of the repository on your system is `cloudera-manager`. This name is usually in square brackets on the first line of the repo file.

For example, you could view the contents of the repo file, including the repo name in brackets. This file might be at `/etc/yum.repos.d/cloudera-manager.repo` and its contents could be viewed using the `more` command as follows:

```
[user@system yum.repos.d]$ more cloudera-manager.repo
[cloudera-manager]
...
```

The location of the repo files varies by operating system and package management solution.

- For yum the repo file is at `/etc/yum.repos.d/cloudera-manager.repo`.
- For zypper the repo file is at `/etc/zypp/repos.d/cloudera-manager.repo`.

To find the base URL for your distribution go to <http://archive.cloudera.com/cm4/> and navigate to the directory that matches your operating system. For example, for Red Hat 6, you would use http://archive.cloudera.com/cm4/redhat/6/x86_64/cm/. Alternately, you can create your

own repository, as described in [Appendix A - Understanding Custom Installation Solutions](#). Creating your own repository is necessary if you are upgrading machines that do not have access to the Internet.

To clean all `yum`'s cache directories, use the command `yum clean all`. Doing so ensures that you download and install the latest versions of the packages. If your system is not up to date, and any underlying system components need to be upgraded before this `yum update` can succeed, `yum` will tell you what those are.

To upgrade to the new server

1. Stop the server on the 3.7.x Server host:

```
$ sudo service cloudera-scm-server stop
```

2. If you are using the embedded PostgreSQL database, stop `cloudera-manager-server-db` on the host on which it is running:

```
$ sudo service cloudera-manager-server-db stop
```

3. Install the new version of the server.

- You can run commands on the Cloudera Manager Server host to update only the Cloudera Manager components. For example, under Red Hat, to upgrade from Cloudera's repository you can run commands such as the following on the Cloudera Manager Server host:

```
$ sudo yum clean all
$ sudo yum update --disablerepo='*' --enablerepo=cloudera-
manager
```

On a SLES system, use commands like this to clean cached repository information update only the Cloudera Manager components:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://myhost.example.com/path_to_cm_repo
```

For example:

```
$ sudo zypper clean --all
$ sudo zypper up -r
http://archive.cloudera.com/cm4/sles/11/x86_64/cm/
```

Upgrading to Cloudera Manager 4.1

- At the end of this process you should have the 4.1 versions of the following packages installed on the host that will become the Cloudera Manager Server host:

```
cloudera-manager-agent.x86_64
cloudera-manager-daemons.noarch
cloudera-manager-server.noarch
cloudera-manager-server-db.noarch
```

In the preceding example, <arch> is an architecture-specific suffix such as x86_64. You may also see additional packages for plugins, depending on what was previously installed on the Server host.

4. Start the server.

On the Cloudera Manager Server host (the system on which you installed the `cloudera-manager-server.noarch` package) do the following:

```
$ sudo service cloudera-scm-server-db start
$ sudo service cloudera-scm-server start
```

You should see the following:

```
Starting cloudera-scm-server: [ OK ]
```

Note

If you have problems starting the server, such as database permissions problems, you can use the server's log `/var/log/cloudera-scm-server/cloudera-scm-server.log` to troubleshoot the problem.

Upgrade the Cluster Hosts

After updating Cloudera Manager, connect to Cloudera Manager and use the wizard to continue the upgrade process. In this part of the process, the Cloudera Manager agents are updated and databases are updated. The Host Monitor role is a new addition for Cloudera Manager 4, so upgrading includes adding this role and its supporting database.

Important

All hosts in the cluster must have access to the Internet if you plan to use `archive.cloudera.com` as the source for installation files. If you do not have Internet access, create a custom repository.

1. Log in to the Cloudera Manager Admin Console. If you have just restarted the Cloudera Manager server, you may need to log in again.
2. On the Welcome screen, click **Continue** to proceed to the Upgrade cluster hosts screen.
3. On the Upgrade cluster hosts screen, verify that the hosts you want to upgrade appear. You can search for additional hosts, if you need to, by entering their hostnames or IP addresses and clicking **Find Hosts**.

When all the hosts are shown as being managed, click **Continue**.

4. Provide credentials for authenticating with hosts.
 - a. Select **root** or enter the user name for an account that has password-less sudo permissions.
 - b. Select an authentication method.
 - If you choose to use password authentication, enter and confirm the password.
 - If you choose to use public-key authentication provide a passphrase and path to the required key files.
 - You can choose to specify an alternate SSH port. The default value is 22.
 - You can specify the maximum number of host installations to run at once. The default value is 10.
5. Click **Start Installation** to install and start Cloudera Manager Agents.

The status of installation on each host is displayed on the page that appears after you click **Start Installation**. You can also click the **Details** link for individual hosts to view detailed information about the installation and error messages if installation fails on any hosts.

Note

If you click the **Abort Installation** button while installation is in progress, it will halt any pending or in-progress installations and roll back any in-progress installations to a clean state. The **Abort Installation** button does not affect host installations that have already completed successfully or already failed.

If installation fails on a host, you can click the **Retry** link next to the failed host to try installation on that host again. To retry installation on all failed hosts, click **Retry Failed Hosts** at the bottom of the screen.

6. When the **Continue** button appears at the bottom of the screen, the installation process is complete.

If the installation has completed successfully on some hosts but failed on others, you can click

Continue if you want to skip installation on the failed hosts and continue to the next screen to start installing the Cloudera Management services on the successful hosts.

7. On the next screen, click **Continue** to install the Cloudera Management services.
8. On the next screen, select any services to be installed. Provide information about the databases to be used with those services. Click **Continue** to install the Cloudera Management services.
9. The wizard evaluates the hardware configurations of the cluster hosts to determine the best machines for each role.

Important

For best performance, make sure the Host Monitor role is assigned to the host on which you installed the corresponding databases. For example, if you created the Host Monitor database on `myhost1`, then you should assign the Activity Monitor role to `myhost1`. The JDBC connector **must** be installed and configured on any machine to which any of these roles is assigned.

Select a host and click **Continue**.

10. On the Database Setup page, enter any required information for Host Monitor databases.

Important

The value you enter as the database hostname **must** match the value you entered for the hostname (if any) when you created the database (see [Installing and Configuring Databases](#)).

- a. Enter the fully-qualified domain name for the server that is hosting the database in **Database Host Name**.
- b. Select the proper database type from the choices provided in **Database Type**.
- c. Enter the name you specified when you created the database in **Database Name**.
- d. Enter the user name you specified when you created the database in **Username**.
- e. Enter the password you specified when you created the database in **Password**.

Note

Problems may occur if a database with a blank password is used.

11. Click **Test Connection** to confirm that Cloudera Manager can communicate with the databases using the information you have supplied. This transaction takes two heartbeats to complete (about 30 seconds with the default heartbeat interval).

If the test succeeds in all cases, click **Continue**; otherwise check and correct the information you have provided for the databases and then try the test again.

12. Review the configuration changes to be applied during the upgrade and click **Accept**.

The upgrade process executes commands associated with applying the configuration changes. For example, the upgrade process may create and configure new roles. At the end of this process, status information is displayed for your system.

13. Review the **Status and Health Summary** to click any service listed as **Started with Outdated Configurations** or **Stopped**.
14. On the summary page for the service, click **Actions** and choose the option to start or restart the service. For example, if the Host Monitor service was stopped, click **Start this Host Monitor...**
15. Confirm that you want to start or restart the service. For example for the Host Monitor service, click **Start this Host Monitor**.

The **Command Details** dialog box appears and displays command process. If the services start without errors, you have completed the upgrade to Cloudera Manager 4.1. To verify that the upgrade has completed as expected, check the server versions.

Deploy Updated Client Configurations

During upgrades between major versions, resource locations may change. To ensure clients have current information about resources, update client configuration as described in [Deploying Client Configuration Files](#).

Verify the Upgrade

You can use the host inspector to verify the upgrade completed.

To verify the upgrade has completed as expected

1. [Connect to the Cloudera Manager Admin Console](#).
2. Click the **Hosts** tab.
3. Click **Host Inspector**.
4. Click **Show Inspector Results**.

All results from the host inspector process are displayed including the currently installed versions. If this includes listings of current component versions, the installation completed as expected.

Upgrading to Cloudera Manager 4.1

Upgrade CDH

Cloudera Manager 4 can manage both CDH3 and CDH4, so upgrading existing CDH3 installations is not required, but to get the benefits of CDH4, you may want to upgrade CDH. See the following topics for more information on upgrading CDH:

- [Upgrading CDH3 to CDH4 in a Cloudera Managed Deployment](#)
- [Upgrading CDH](#)

Upgrade from Cloudera Manager 4 to the Latest Cloudera Manager 4

Upgrading from Cloudera Manager 4 to the latest version of Cloudera Manager is a relatively simple process, which centers on upgrading Cloudera Manager Server packages. This process applies to upgrading from Cloudera Manager 4 to a newer version of Cloudera Manager 4. For example, this process applies to upgrading Cloudera Manager 4.0 Beta or Cloudera Manager 4.0.2.

To complete the upgrade, you stop the Cloudera Management Service, upgrade the packages (and database tables, if necessary), and then start the Cloudera Management Service again.

It is possible to complete the following upgrade without shutting down the Hadoop services. Hadoop daemons can continue running, unaffected, while Cloudera Manager is upgraded.

Warning

Cloudera Manager 4.0 can manage CDH3 and CDH4, but cannot manage CDH4.0 beta. If you upgrade to Cloudera Manager 4.0, you must upgrade any existing installations of CDH4.0 beta, as well.

Summary: What You are Going to Do

Upgrading from Cloudera Manager 4.0 to the latest version of Cloudera Manager involves the following broad steps:

[Step 1. Stop the Cloudera Management Service](#)

[Step 2. Upgrade the Cloudera Manager Server and Agent Packages](#)

[Step 3. Start the Server](#)

[Step 4. Upgrade the Cluster Hosts](#)

[Step 5. Verify the Upgrade Succeeded](#)

Step 1. Stop the Cloudera Management Service

The Cloudera Manager Service must be stopped before upgrades can occur.

To stop the Cloudera Management Service

1. Click the **Services** tab in Cloudera Manager Admin Console.

2. Choose **Stop** on the **Actions** menu for the Cloudera Management Services.

Step 2. Upgrade the Cloudera Manager Server and Agent Packages

In this step, you upgrade the Cloudera Manager Server packages to the latest version.

1. Stop the server on the Cloudera Manager Server host using the following command:

```
$ sudo service cloudera-scm-server stop
```

2. Install the new version of the server.

To install the new version, you can upgrade from Cloudera's repository at

<http://archive.cloudera.com/cm4/>. Alternately, you can create your own repository, as described in [Appendix A - Understanding Custom Installation Solutions](#). Creating your own repository is necessary if you are upgrading a cluster that does not have access to the Internet.

For example, under Red Hat, to upgrade from Cloudera's repository you can run commands such as the following on the Cloudera Manager Server host to update only the Cloudera Manager components:

```
$ sudo yum clean all
$ sudo yum update --disablerepo='*' --enablerepo=cloudera-manager
```

Notes

- `cloudera-manager` is the name of the repository on your system; the name is usually in square brackets on the first line of the repo file, in this example `/etc/yum.repos.d/cloudera-manager.repo`:

```
[daly@c0129-5 yum.repos.d]$ more cloudera-manager.repo
[cloudera-manager]
...
```

- `yum clean all` cleans up yum's cache directories, ensuring that you download and install the latest versions of the packages.
- If your system is not up to date, and any underlying system components need to be upgraded before this `yum update` can succeed, yum will tell you what those are.

On a SLES system, use commands like this to clean cached repository information update only the Cloudera Manager components:

Upgrading to Cloudera Manager 4.1

```
$ sudo zypper clean --all
$ sudo zypper up -r http://myhost.example.com/path_to_cm_repo
```

For example:

```
$ sudo zypper clean --all
$ sudo zypper up -r
http://archive.cloudera.com/cm4/sles/11/x86_64/cm/4/
```

On a Debian/Ubuntu system, use commands like this to clean cached repository information and update only the Cloudera Manager components:

```
$ sudo apt-get clean
$ sudo apt-get upgrade -t squeeze-cm4
```

At the end of this process you should have the 4.0 versions of the following packages installed on the Cloudera Manager Server host:

```
cloudera-manager-daemons.noarch
cloudera-manager-server.noarch
cloudera-manager-server-db.noarch
```

You may also see additional packages for plugins, depending on what was previously installed on the Server host.

If the commands to update the server complete without errors, you can assume the upgrade has completed as desired. For additional assurance, you will have the option to check that the server versions have been updated after you start the server. The process of checking the server version is described in [Step 5. Verify the Upgrade Succeeded](#).

Step 3. Start the Server

To start the server

On the Cloudera Manager Server host (the system on which you installed the `cloudera-manager-server.noarch` package) do the following:

```
$ sudo service cloudera-scm-server-db start
$ sudo service cloudera-scm-server start
```


Note

The `sudo service cloudera-scm-server-db start` command is necessary if you are using the embedded PostgreSQL database.

You should see the following:

```
Starting cloudera-scm-server: [ OK ]
```

Note

If you have problems starting the server, such as database permissions problems, you can use the server's log `/var/log/cloudera-scm-server/cloudera-scm-server.log` to troubleshoot the problem.

Step 4. Upgrade the Cluster Hosts

Cloudera Manager can automatically upgrade existing agents. After you upgrade Cloudera Manager, when it is started for the first time, it checks for any older versions of agents. If older agents are detected, Cloudera Manager provides the opportunity to automatically update agents, which you should do, unless you have some reason not to do so.

To upgrade the agents**Important**

All hosts in the cluster must have access to the Internet if you plan to use `archive.cloudera.com` as the source for installation files. If you do not have Internet access, create a custom repository.

1. Log in to the Cloudera Manager Admin Console. If you have just restarted the Cloudera Manager server, you may need to log in again.
2. On the Welcome screen, click **Continue** to proceed to the Upgrade cluster hosts screen.
3. On the Upgrade cluster hosts screen, verify that the hosts you want to upgrade appear. You can search for additional hosts, if you need to, by entering their hostnames or IP addresses and clicking **Find Hosts**.

When all the hosts are shown as being managed, click **Continue**.

4. Provide credentials for authenticating with hosts.

- a. Select **root** or enter the user name for an account that has password-less sudo permissions.
 - b. Select an authentication method.
 - If you choose to use password authentication, enter and confirm the password.
 - If you choose to use public-key authentication provide a passphrase and path to the required key files.
 - You can choose to specify an alternate SSH port. The default value is 22.
 - You can specify the maximum number of host installations to run at once. The default value is 10.
5. Click **Start Installation** to install and start Cloudera Manager Agents.

The status of installation on each host is displayed on the page that appears after you click **Start Installation**. You can also click the **Details** link for individual hosts to view detailed information about the installation and error messages if installation fails on any hosts.

Note

If you click the **Abort Installation** button while installation is in progress, it will halt any pending or in-progress installations and roll back any in-progress installations to a clean state. The **Abort Installation** button does not affect host installations that have already completed successfully or already failed.

If installation fails on a host, you can click the **Retry** link next to the failed host to try installation on that host again. To retry installation on all failed hosts, click **Retry Failed Hosts** at the bottom of the screen.

6. When the **Continue** button appears at the bottom of the screen, the installation process is complete.

If the installation has completed successfully on some hosts but failed on others, you can click **Continue** if you want to skip installation on the failed hosts and continue to the next screen to start installing the Cloudera Management services on the successful hosts.

7. On the next screen, click **Continue** to install the Cloudera Management services.
8. On the next screen, select any services to be installed. Provide information about the databases to be used with those services. Click **Continue** to install the Cloudera Management services.
9. The wizard evaluates the hardware configurations of the cluster hosts to determine the best machines for each role.

Important

For best performance, make sure the Host Monitor role is assigned to the host on which you installed the corresponding databases. For example, if you created the Host Monitor database on `myhost1`, then you should assign the Activity Monitor role to `myhost1`. The JDBC connector **must** be installed and configured on any machine to which any of these roles is assigned.

Select a host and click **Continue**.

10. On the Database Setup page, enter any required information for Host Monitor databases.

Important

The value you enter as the database hostname **must** match the value you entered for the hostname (if any) when you created the database (see [Installing and Configuring Databases](#)).

- a. Enter the fully-qualified domain name for the server that is hosting the database in **Database Host Name**.
- b. Select the proper database type from the choices provided in **Database Type**.
- c. Enter the name you specified when you created the database in **Database Name**.
- d. Enter the user name you specified when you created the database in **Username**.
- e. Enter the password you specified when you created the database in **Password**.

Note

Problems may occur if a database with a blank password is used.

11. Click **Test Connection** to confirm that Cloudera Manager can communicate with the databases using the information you have supplied. This transaction takes two heartbeats to complete (about 30 seconds with the default heartbeat interval).

If the test succeeds in all cases, click **Continue**; otherwise check and correct the information you have provided for the databases and then try the test again.

Step 5. Verify the Upgrade Succeeded

If the commands to update and start the server complete without errors, you can assume the upgrade has completed as desired. For additional assurance, you can check that the server versions have been updated.

Upgrading to Cloudera Manager 4.1

To verify the server upgrade succeeded

1. [Connect to the Cloudera Manager Admin Console](#).
2. Click the **Hosts** tab.
3. Click **Host Inspector**.
On large clusters, the host inspector may take some time to finish running. You must wait for the process to complete before proceeding to the next step.
4. Click **Show Inspector Results**.

All results from the host inspector process are displayed including the currently installed versions. If this includes listings of current component versions, the installation completed as expected.

Testing the Installation

When you have finished the upgrade to Cloudera Manager, you can test the installation; follow instructions under [Testing the Installation](#).

Note

If you change the hostname or port where the Cloudera Manager is running, or you enable TLS security, you must restart the Cloudera Management Services to update the URL to the Server. For instructions, see [Restarting a Service](#).

Upgrade from Cloudera Manager Free Edition 4 to Cloudera Manager 4

Upgrading from Cloudera Manager Free Edition 4 to Cloudera Manager 4 involves the following broad steps:

[Step 1. Install the Cloudera Manager License](#)


[Step 2. Run the upgrade wizard](#)

Note

Cloudera Manager 4 can continue to use the databases that were established for Cloudera Manager 4 Free Edition. Cloudera Manager 4 also supports other database types. To learn about using additional databases for Cloudera Manager, review the guidelines provided under [Installing and Configuring Databases](#). Unless you want to change databases, no database installations are required.

Step 1. Install the Cloudera Manager license

To install the license:

1. Log in to the Admin Console of Cloudera Manager Free Edition 4.
2. Click the gear icon  to display the **Administration** page.

3. On the **License** tab, browse to your Cloudera Manager License file to upgrade to the full version of Cloudera Manager.

After you upload your license file, a message is displayed instructing you to restart Cloudera Manager to allow the new license to take effect.

4. Restart Cloudera Manager. This may be done by running `service cloudera-scm-server restart` from the command line.

Note

After restarting the server, wait a few seconds for the server to finish initializing before you try to reconnect to the Admin Console.

5. Click **Continue** after restarting the server.
6. Log in to the Cloudera Manager Admin Console again.

When you log back in, in the [next step](#), you will be in the upgrade wizard.

Step 2. Run the upgrade wizard

Having restarted the Cloudera Manager server, the wizard appears when you reconnect.

Important

- All hosts must have access to the either `archive.cloudera.com` on the internet or to a local repository with the necessary installation files.

1. On the Welcome page, click **Continue** to proceed to the Upgrade cluster hosts page.
2. On the Upgrade cluster hosts page, verify that the hosts you want to upgrade appear. You can search for additional hosts, if you need to, by entering their hostnames or IP addresses and clicking **Find Hosts**.

When all the hosts are shown as being managed, click **Continue**.

3. Provide SSH credentials.

To authenticate with the hosts, you must either use a root account that is on all of your cluster hosts, or use an account that has password-less sudo permissions. Select **root** or enter the user name for an account that has password-less sudo permissions.

4. You can either use a shared password for the account, or use a public and private key pair.

Note

In this release, Cloudera has tested OpenSSH-style key pairs. Other key pairs (such as

PuTTY-generated pairs) may not work.

To enter a password, click **All hosts accept same password** and enter the account password.

To use a public and private key pair, click **All hosts accept same public key**. Specify or browse for the location of the public and private keys. If your keys contain a passphrase, enter it.

5. Review the Choose Cloudera Manager Repository page. You can use the default repository location or specify an alternate location. If you are using a custom repository, specify that repository's location here.
6. Click **Start Installation** to install new CDH components.

The wizard runs a maximum of 10 installations in parallel to avoid excessive network load. The status of installation on each host is displayed on the page that appears after you click **Start Installation**. You can also click the **Details** link for individual hosts to view detailed information about the installation and error messages if installation fails on any hosts.

Note

If you click the **Abort Installation** button while installation is in progress, it will halt any pending or in-progress installations and roll back any in-progress installations to a clean state. The **Abort Installation** button does not affect host installations that have already completed successfully or already failed.

If installation fails on a host, you can click the **Retry** link next to the failed host to try installation on that host again. To retry installation on all failed hosts, click **Retry Failed Hosts** at the bottom of the page.

7. When the **Continue** button appears at the bottom of the page, the installation process is complete.
If the installation has completed successfully on some hosts but failed on others, you can click **Continue** if you want to skip installation on the failed hosts and continue to the next page to start installing the Cloudera Management services on the successful hosts.
8. On the next page, click **Continue** to install the Cloudera Management services.
9. On the next page, select the dependenc(ies) for your service(s) and click **Continue**.
10. The wizard evaluates the hardware configurations of the cluster hosts to determine the best machines for each role.

Important

For best performance, make sure the Activity Monitor, Service Monitor, Report Manager, and Host Monitor roles are assigned to the host(s) that host the databases for those services.

Click **Continue** when you are satisfied with the assignments.

11. On the Database Setup page, enter the information requested. This may be the database information for the database you are continuing to use from when you installed Cloudera Manager Free Edition. If you intend to use new databases, provide their information here. At most you will need to provide information for up to the Activity Monitor, Service Monitor, Report Manager, and Host Monitor databases.
12. Click **Test Connection** to confirm that Cloudera Manager can communicate with the databases using the information you have supplied. This atypical transaction takes two heartbeats to complete (about 30 seconds with the default heartbeat interval).

If the test succeeds in all cases, click **Continue**; otherwise check and correct the information you have provided for the databases and then try the test again.

13. Confirm the settings entered for file system paths, such as the NameNode Data Directory and the DataNode Data Directory.
14. Supply the name of the mail server (it can be `localhost`), the mail server user, and the mail recipients.
15. Click **Continue**.

The wizard starts the services on your cluster.

16. When all of the services are started, click **Continue**.
17. Click **Continue**, and click **Continue** on the next page, which confirms that the upgrade has succeeded.

You have now completed the upgrade to Cloudera Manager 4.

Upgrading CDH

Cloudera Manager 4 can manage both CDH3 and CDH4, so upgrading existing CDH3 installations is not required, but to get the benefits of CDH4, you may want to upgrade CDH. See the following topics for more information on upgrading CDH:

- [Upgrading CDH](#)
- [Upgrading CDH3 to CDH4](#)

Upgrading CDH in a Cloudera Managed Deployment

Cloudera Manager supports managing CDH4 and CDH3. To ensure the highest level of functionality and stability, consider upgrading to the most recent version of CDH4 or CDH3. The process of upgrading CDH to the most recent version is different when those CDH installations are managed by Cloudera Manager.

Upgrading CDH4 or CDH3 to the most recent version is described in the following topics:

- [Upgrading CDH3 to CDH4 in a Cloudera Managed Deployment](#)
- [Upgrading to the Latest Version of CDH4 in a Cloudera Managed Deployment](#)
- [Upgrading to the Latest Version of CDH3 in a Cloudera Managed Deployment](#)

Upgrading CDH3 to CDH4 in a Cloudera Managed Deployment

Important

The following instructions describe how to upgrade components managed by Cloudera Manager from a CDH3 release to the latest CDH4 release. This involves uninstalling the CDH3 packages and installing the CDH4 packages.

For instructions on upgrading components, such as [Flume](#), that Cloudera Manager does not manage, see the [CDH4 Installation Guide](#).

Before You Begin

Important

Before upgrading, be sure to read about the latest [Incompatible Changes](#) and [Known Issues and Work Arouns in CDH4](#) in the [CDH4 Release Notes](#).

If you are upgrading a cluster that is part of a production system, be sure to plan ahead. As with any operational work, be sure to reserve a maintenance window with enough extra time allotted in case of complications. The Hadoop upgrade process is well understood, but it is best to be cautious. For production clusters, Cloudera recommends allocating up to a full day maintenance window to perform the upgrade, depending on the number of hosts, the amount of experience you have with Hadoop and Linux, and the particular hardware you are using.

To upgrade CDH in multiple clusters, repeat this process for each cluster.

Upgrading to CDH4

Use the instructions that follow to upgrade to CDH4.

Step 1: Back Up Important Items

1. Back up the databases

For instructions, see [Database Considerations for Cloudera Manager Upgrades](#).

Do this step now if your cluster includes any Ubuntu or Debian systems running CDH3u3 or earlier.

Otherwise, you can perform this step later if you prefer – any time before you use Cloudera Manager to [upgrade the cluster](#).

2. If the directory `/usr/lib/oozie/libext` exists, move it to a temporary location before you proceed.

Step 2: Stop All CDH Components

To stop all services

1. In the Cloudera Manager Admin Console, click the **Services** tab.
2. Click the top **Actions** button that corresponds to the cluster and choose **Stop**. Click **Stop** in the confirmation screen.

The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Note

As necessary, repeat the process of stopping services for each cluster.

3. For each Cloudera Management Service entry, click **Actions** and click **Stop**. Click **Stop** in the confirmation screen.

The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Step 3: Back up the HDFS Metadata

Back up the HDFS metadata on the NameNode machine.

Important

Do the following when you are sure that all Hadoop services have been shut down. **It is particularly important that the NameNode service is not running so that you can make a consistent backup.**

Cloudera recommends backing up HDFS metadata on a regular basis, as well as before a major upgrade.

1. On the **Services** page of Cloudera Manager, click the link for the HDFS service, then on that page find the name of the NameNode Data Directory (under **NameNode Settings**).
2. From the command line on the NameNode machine, back up that directory; for example, if the data directory is `/mnt/hadoop/hdfs/name`, do the following as root:

```
# cd /mnt/hadoop/hdfs/name
# tar -cvf /root/nn_backup_data.tar .
```

You should see output like this:

```
./
./current/
./current/fsimage
./current/fstime
./current/VERSION
./current/edits
./image/
./image/fsimage
```

3. Check the output.

Warning

If you see a file containing the word *lock*, the NameNode is probably still running. Repeat the preceding steps, starting by shutting down the Hadoop services.

Step 4: Uninstall CDH3

Uninstall CDH3 on each host.

- On Red Hat-compatible systems:

```
$ sudo yum remove hadoop-0.20 hue-common hadoop-pig oozie-client hive
hadoop-hbase hadoop-zookeeper mahout
```

- On SUSE systems:

```
$ sudo zypper remove hadoop-0.20 hue-common hadoop-pig oozie-client
hive hadoop-hbase hadoop-zookeeper mahout
```

- On Ubuntu and Debian systems:

```
$ sudo apt-get purge hadoop-0.20 hue-common hadoop-pig oozie-client
hive hadoop-hbase hadoop-zookeeper mahout
```

Warning

If you are upgrading an Ubuntu or Debian system from CDH3u3 or earlier, you **must** use `apt-get purge` (rather than `apt-get remove`) to make sure the re-install succeeds, but be aware that `apt-get purge` removes all your configuration data. If you have modified any configuration files, DO NOT PROCEED before backing them up.

Step 5: Download CDH4

On Red Hat-compatible systems:

- Download the CDH4 Package:
 - Click the entry in the table below that matches your Red Hat or CentOS system, choose **Save File**, and save the file to a directory to which you have write access (it can be your home directory).

For OS Version	Click this Link
Red Hat/CentOS/Oracle 5	Red Hat/CentOS/Oracle 5 link
Red Hat/CentOS 6 (32-bit)	Red Hat/CentOS 6 link (32-bit)

Upgrading CDH in a Cloudera Managed Deployment

For OS Version	Click this Link
Red Hat/CentOS 6 (64-bit)	Red Hat/CentOS 6 link (64-bit)

b. Install the RPM:

```
$ sudo yum --nogpgcheck localinstall cloudera-cdh-4-0.noarch.rpm
```

Note

For instructions on how to add a CDH4 yum repository or build your own CDH4 yum repository, see [Installing CDH4 On Red Hat-compatible systems](#).

- (Optionally) add a repository key on each system in the cluster. Add the Cloudera Public GPG Key to your repository by executing one of the following commands:

- **For Red Hat/CentOS/Oracle 5 systems:**

```
$ sudo rpm --import  
http://archive.cloudera.com/cdh4/redhat/5/x86_64/cdh/RPM-GPG-KEY-cloudera
```

-
-
-

- **For Red Hat/CentOS 6 systems:**

```
$ sudo rpm --import  
http://archive.cloudera.com/cdh4/redhat/6/x86_64/cdh/RPM-GPG-KEY-cloudera
```

On SUSE systems:

- Download the CDH4 Package:
 - Click [this link](#), choose **Save File**, and save it to a directory to which you have write access (it can be your home directory).

b. Install the RPM:

```
$ sudo rpm -i cloudera-cdh-4-0.noarch.rpm
```

Note

For instructions on how to add a repository or build your own repository, see [Installing CDH4 on SUSE Systems](#).

2. Update your system package index by running:

```
$ sudo zypper refresh
```

3. (Optionally) add a repository key on each system in the cluster. Add the Cloudera Public GPG Key to your repository by executing the following command:

o **For all SUSE systems:**

```
$ sudo rpm --import  
http://archive.cloudera.com/cdh4/sles/11/x86_64/cdh/RPM-GPG-  
KEY-cloudera
```

On Ubuntu and Debian systems:

1. Download the CDH4 Package:

a. Click one of the following:

[this link for a Squeeze system](#), or
[this link for a Lucid system](#)
[this link for a Precise system](#).

b. Install the package. Do one of the following:

Choose **Open with** in the download window to use the package manager, or
Choose **Save File**, save the package to a directory to which you have write access (it can be your home directory) and install it from the command line, for example:

```
sudo dpkg -i Downloads/cdh4-repository_1.0_all.deb
```

Note

For instructions on how to add a repository or build your own repository, see [Installing CDH4 on Ubuntu Systems](#).

2. (Optionally) add a repository key on each system in the cluster. Add the Cloudera Public GPG Key to your repository by executing one of the following commands:

- **For Ubuntu Lucid systems:**

```
$ curl -s
http://archive.cloudera.com/cdh4/ubuntu/lucid/amd64/cdh/archive
.key | sudo apt-key add -
```

- **For Ubuntu Precise systems:**

```
$ curl -s
http://archive.cloudera.com/cdh4/ubuntu/precise/amd64/cdh/archi
ve.key | sudo apt-key add -
```

- **For Debian Squeeze systems:**

```
$ curl -s
http://archive.cloudera.com/cdh4/debian/squeeze/amd64/cdh/archi
ve.key | sudo apt-key add -
```

Step 6: Re-Install HDFS, MapReduce, and the CDH4 Components

- Use one of the following commands to install CDH4 packages on every host in your cluster:

On Red Hat/CentOS/Oracle systems:

```
$ sudo yum -y install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop
hadoop-hdfs hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client
hadoop-0.20-mapreduce hbase hive oozie oozie-client pig zookeeper
mahout
```

On SUSE systems:

```
$ sudo zypper install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop
hadoop-hdfs hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client
hadoop-0.20-mapreduce hbase hive oozie oozie-client pig zookeeper
mahout
```

On Debian/Ubuntu systems:

```
$ sudo apt-get install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop
hadoop-hdfs hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client
hadoop-0.20-mapreduce hbase hive oozie oozie-client pig zookeeper
mahout
```

- To install the `hue-common` package and all Hue applications on the Hue machine, install the `hue` meta-package.

Important

If you used the Hue Authorization Manager with CDH3, you must remove the `hue-userman` package, and disable or remove the Authorization Manager repository before installing the new version of Hue. The repository is the one you installed when you [configured Authorization Manager](#). For example, on a Red Hat system, the repository file is `/etc/yum.repos.d/cloudera-authman.repo` by default. Either remove this file or add a line that reads `enabled=0` (or, if there is already a line that reads `enabled=1`, change the 1 to a 0).

To install the `hue` meta-package on Red Hat/CentOS/Oracle systems:

```
$ sudo yum install hue
```

To install the `hue` meta-package on SUSE systems:

```
$ sudo zypper install hue
```

To install the `hue` meta-package on Debian/Ubuntu systems:

```
$ sudo apt-get install hue
```

- If you moved `/usr/lib/oozie/libext` to a temporary location in [Step 1](#), copy its contents (not the directory itself) back to the new `/usr/lib/oozie/libext` now.

Step 7: Disable Start on Boot for Hue, Oozie, and HttpFS:

- To prevent Hue from starting on the Hue machine:

```
$ sudo /sbin/chkconfig hue off
```

- To prevent Oozie from starting on system boot on every machine on which it is installed:
 - On Red Hat-compatible and SUSE systems:

Upgrading CDH in a Cloudera Managed Deployment

```
$ sudo /sbin/chkconfig oozie off
```

- On Ubuntu and Debian systems:

```
$ sudo /usr/sbin/update-rc.d oozie disable
```

- To prevent HttpFS from starting on system boot:

- On Red Hat-compatible and SUSE systems:

```
$ sudo /sbin/chkconfig hadoop-httpfs off  
$ sudo service hadoop-httpfs stop
```

- On Ubuntu and Debian systems:

```
$ sudo /usr/sbin/update-rc.d hadoop-httpfs disable  
$ sudo service hadoop-httpfs stop
```

Step 8: Upgrade the HDFS Metadata and the Cluster Configuration

Important

- If you have not already [backed up your configuration data](#), do so now.
- Before you proceed, click on the **Hosts** tab in Cloudera Manager and make sure that all hosts are up and running CDH4.

To upgrade the cluster

1. In the Cloudera Manager Admin Console, click the **Services** tab, click **Actions** and click **Upgrade Cluster**.
2. Click **Upgrade Cluster** to confirm you want to upgrade the cluster.

If you are already on the **Services** page, the **Upgrade Cluster** may not be available. If this occurs, refresh the page.

Cloudera Manager updates the configuration, upgrades HDFS metadata, and upgrades the Oozie database. Upgrading CDH from CDH3 or CDH4 Beta 1 requires these changes.

Step 9: Finalize the HDFS Metadata Upgrade

After ensuring that the CDH4 upgrade has succeeded and that everything is running smoothly, finalize the HDFS metadata upgrade. It is not unusual to wait days or even weeks before finalizing the upgrade.

To finalize the HDFS metadata upgrade

1. In the Cloudera Manager Admin Console, click the **Services** tab and click the HDFS service.
2. Click **Actions** and click **Finalize Metadata Upgrade**.
3. Click **Finalize Metadata Upgrade** to confirm you want to complete this process.

Cloudera Manager finalizes the metadata upgrade. The upgrade is now complete.

Upgrading to the Latest Version of CDH4 in a Cloudera Managed Deployment

Before You Begin

Important

- **Use the right instructions:** the following instructions describe how to upgrade to the latest CDH4 release from an earlier CDH4 release in a Cloudera Managed Deployment. **If you are upgrading from a CDH3 release, use the instructions under [Upgrading CDH3 to CDH4 in a Cloudera Managed Deployment](#) instead.**

- Before upgrading, be sure to read about the latest [Incompatible Changes](#) and [Known Issues and Work Arouds in CDH4](#) in the [CDH4 Release Notes](#).
- If you are upgrading a cluster that is part of a production system, be sure to plan ahead. As with any operational work, be sure to reserve a maintenance window with enough extra time allotted in case of complications. The Hadoop upgrade process is well understood, but it is best to be cautious. For production clusters, Cloudera recommends allocating up to a full day maintenance window to perform the upgrade, depending on the number of hosts, the amount of experience you have with Hadoop and Linux, and the particular hardware you are using.

Upgrading CDH in a Cloudera Managed Deployment

Upgrading Unmanaged Components

Upgrading unmanaged components is a process that is separate from upgrading managed components. Upgrade the unmanaged components before proceeding to upgrade managed components. For example, if you have unmanaged Flume installed, upgrade that before proceeding to upgrade managed components. Components that you might have installed that are not managed by Cloudera Manager include:

- [Flume 1.x](#)
- [Sqoop](#)
- [Pig](#)
- [Hive](#)
- [Whirr](#)
- [Mahout](#)

Step 1. Stop all the CDH Services on All Hosts

You must stop all Hadoop services before upgrading CDH. **To stop all services**

1. In the Cloudera Manager Admin Console, click the **Services** tab.
2. Click the top **Actions** button that corresponds to the cluster and choose **Stop**. Click **Stop** in the confirmation screen.

The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Note

As necessary, repeat the process of stopping services for each cluster.

3. For each Cloudera Management Service entry, click **Actions** and click **Stop**. Click **Stop** in the confirmation screen.

The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Repeat this process for all clusters hosting CDH4 machines to be upgraded.

Step 2. Back up the HDFS Metadata on the NameNode

Important

Do the following when you are sure that all Hadoop services have been shut down. **It is particularly important that the NameNode service is not running so that you can make a consistent backup.**

Cloudera recommends backing up HDFS metadata on a regular basis, as well as before a major upgrade.

1. On the **Services** page of Cloudera Manager, click the HDFS service, then the **Configuration** tab. Navigate to the **NameNode** category and find **NameNode Data Directory**.
2. From the command line on the NameNode machine, back up that directory; for example, if the data directory is `/mnt/hadoop/hdfs/name`, do the following as root:

```
# cd /mnt/hadoop/hdfs/name
# tar -cvf /root/nn_backup_data.tar .
```

You should see output like this:

```
./
./current/
./current/fsimage
./current/fstime
./current/VERSION
./current/edits
./image/
./image/fsimage
```

3. Check the output.

Warning

If you see a file containing the word *lock*, the NameNode is probably still running. Repeat the preceding steps, starting by shutting down the Hadoop services.

Step 3. Upgrade Managed Components

There are a variety of strategies that you can use to upgrade to the latest version of CDH4.

- You can use your operating system's package management tools to update all packages to the latest version using standard repositories. This approach works well because it minimizes the

Upgrading CDH in a Cloudera Managed Deployment

amount of configuration required and uses the simplest commands. Be aware that this can take a considerable amount of time if you have not upgraded the system recently.

- You can target the `cloudera.com` repository that is added during a typical install, only updating Cloudera components. This limits the scope of updates to be completed, so the process takes less time. This will not work if you created and used a custom repository.
- You can use a custom repository. This process can be more complicated, but enables updating Cloudera components for CDH machines that are not connected to the Internet.

Updating Everything

You can update all components on your system, including Cloudera components. Note that this may take a significant amount of time. To update all packages on your system, use the following command:

- On Red Hat systems:

```
$ sudo yum update
```

- On SLES systems:

```
$ sudo zypper up
```

- On Ubuntu/Debian systems:

```
$ sudo apt-get upgrade
```

Once you complete the process of updating all components, proceed to [Start the Services you Stopped](#).

Updating Cloudera Components Using Default Repositories

To install the new version, you can upgrade from Cloudera's repository by adding an entry to your operating system's package management configuration file. The repository location varies by operating system.

Operating System	Configuration File Repository Entry
Red Hat	http://archive.cloudera.com/cdh4/redhat/6/x86_64/cdh/4/
SLES	http://archive.cloudera.com/cdh4/sles/11/x86_64/cdh/4/
Debian Squeeze	[arch=amd64] http://archive.cloudera.com/cdh4/debian/squeeze-squeeze-cdh4-contrib

Operating System	Configuration File Repository Entry
Ubuntu Lucid	[arch=amd64] http://archive.cloudera.com/cdh4/ubuntu/lucid/amd64/cdh lucid-cdh4 contrib
Ubuntu Precise	[arch=amd64] http://archive.cloudera.com/cdh4/ubuntu/precise/amd64/cdh precise-cdh4 contrib

For example, under Red Hat, to upgrade from Cloudera's repository you can run commands such as the following on the CDH host to update only CDH:

```
$ sudo yum clean all
$ sudo yum update --disablerepo='*' --enablerepo=cloudera-cdh4
```

Note

– cloudera-cdh4 is the name of the repository on your system; the name is usually in square brackets on the first line of the repo file, in this example `/etc/yum.repos.d/cloudera-cdh4.repo`:

```
[chris@ca727 yum.repos.d]$ more cloudera-cdh4.repo
[cloudera-cdh4]
...
```

– `yum clean all` cleans up yum's cache directories, ensuring that you download and install the latest versions of the packages.

– If your system is not up to date, and any underlying system components need to be upgraded before this yum update can succeed, yum will tell you what those are.

On a SLES system, use commands like this to clean cached repository information and then update only the CDH components. For example:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://archive.cloudera.com/cdh4/sles/11/x86_64/cdh/4
```

To verify the URL, open the Cloudera repo file in `/etc/zypp/repos.d` on your system (for example `/etc/zypp/repos.d/cloudera-cdh4.repo`) and look at the line beginning

```
baseurl=
```

Upgrading CDH in a Cloudera Managed Deployment

Use that URL in your `sudo zypper up -r` command.

On a Debian/Ubuntu system, use commands like this to clean cached repository information and then update only the CDH components. First:

```
$ sudo apt-get clean
```

After cleaning the cache, use one of the following upgrade commands to upgrade CDH.

Precise:

```
$ sudo apt-get upgrade -t precise-cdh4
```

Lucid:

```
$ sudo apt-get upgrade -t lucid-cdh4
```

Squeeze:

```
$ sudo apt-get upgrade -t squeeze-cdh4
```

At the end of this process you should have the most recent versions of the CDH packages installed on the host and you can now proceed to [Start the Services you Stopped](#).

Updating Cloudera Components Using Custom Repositories

You can create your own repository, as described in [Appendix A - Understanding Custom Installation Solutions](#). Creating your own repository is necessary if you are upgrading a cluster that does not have access to the Internet.

If you used a custom repository to complete the installation of current files and now you want to update using a custom repository, the details of the steps to complete the process are variable.

In general, begin by updating any existing custom repository that you will use with the installation files you wish to use. This can be completed in a variety of ways. For example, you might use `wget` to copy the necessary installation files. Once the installation files have been updated, use the custom repository you established for the initial installation to update CDH.

Red Hat

On a Red Hat system ensure you have a custom repo that is configured to use your internal repository. For example, if you could have custom repo file in `/etc/yum.conf.d/` called `cdh_custom.repo` in which you specified a local repository. In such a case, you might use the following commands:

```
$ sudo yum clean all
$ sudo yum update --disablerepo='*' --enablerepo=cdh_custom
```

SLES

On a SLES system, use commands such as the following to clean cached repository information and then update only the CDH components:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://internalserver.example.com/path_to_cdh_repo
```

Debian/Ubuntu

Use a command that targets upgrade of your CDH distribution using the custom repository specified in your `apt` configuration files. These files are typically either the `/etc/apt/apt.conf` file or in various files in the `/etc/apt/apt.conf.d/` directory. Information about your custom repository must be included in the repo files. The general form of entries in Debian/Ubuntu is:

```
deb http://server.example.com/directory/ dist-name pool
```

For example, the entry for the default repo is:

```
deb http://us.archive.ubuntu.com/ubuntu/ precise universe
```

On a Debian/Ubuntu system, use commands such as the following to clean cached repository information and then update only the CDH components:

```
$ sudo apt-get clean
$ sudo apt-get upgrade -t your_cdh_repo
```

Step 4. Start the Services you Stopped

You can now start the services that you stopped in Step 1. Proceed as follows:

1. In the Cloudera Manager Admin Console, click the **Services** tab.
2. Click the top **Actions** button that corresponds to the cluster and choose **Start**.

Upgrading CDH in a Cloudera Managed Deployment

The **Command Details** window shows the progress of starting services.

When **All services successfully started** appears, the task is complete and you may close the **Command Details** window.

Repeat this process for all clusters that you previously stopped.

Upgrading to the Latest Version of CDH3 in a Cloudera Managed Deployment

Before You Begin

Important

Before upgrading, be sure to read about the latest [Incompatible Changes](#) and [Known Issues and Work Arounds in CDH3](#) in the [CDH3 Release Notes](#).

Note

If you are upgrading a cluster that is part of a production system, be sure to plan ahead. As with any operational work, be sure to reserve a maintenance window with enough extra time allotted in case of complications. The Hadoop upgrade process is well understood, but it is best to be cautious. For production clusters, Cloudera recommends allocating up to a full day maintenance window to perform the upgrade, depending on the number of hosts, the amount of experience you have with Hadoop and Linux, and the particular hardware you are using.

Upgrading Unmanaged Components

Upgrading unmanaged components is a process that is separate from upgrading managed components. Upgrade the unmanaged components before proceeding to upgrade managed components. For example, if you have unmanaged Flume installed, upgrade that before proceeding to upgrade managed components. Components that you might have installed that are not managed by Cloudera Manager include:

- [Flume 0.9.x](#)
- [Flume 1.x](#)
- [Sqoop](#)
- [Pig](#)
- [Hive](#)
- [Whirr](#)
- [Mahout](#)

Step 1. Stop all the CDH Services on All Hosts

You must stop all Hadoop services before upgrading CDH. **To stop all services**

1. In the Cloudera Manager Admin Console, click the **Services** tab.
2. Click the top **Actions** button that corresponds to the cluster and choose **Stop**. Click **Stop** in the confirmation screen.

The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Note

As necessary, repeat the process of stopping services for each cluster.

3. For each Cloudera Management Service entry, click **Actions** and click **Stop**. Click **Stop** in the confirmation screen.

The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Repeat this process for all clusters hosting CDH3 machines to be upgraded.

Step 2. Back up the HDFS Metadata on the NameNode

Important

Do the following when you are sure that all Hadoop services have been shut down. **It is particularly important that the NameNode service is not running so that you can make a consistent backup.**

Cloudera recommends backing up HDFS metadata on a regular basis, as well as before a major upgrade.

1. On the **Services** page of Cloudera Manager, click the HDFS service, then the **Configuration** tab. Navigate to the **NameNode** category and find **NameNode Data Directory**.
2. From the command line on the NameNode machine, back up that directory; for example, if the data directory is `/mnt/hadoop/hdfs/name`, do the following as root:

Upgrading CDH in a Cloudera Managed Deployment

```
# cd /mnt/hadoop/hdfs/name
# tar -cvf /root/nn_backup_data.tar .
```

You should see output like this:

```
./
./current/
./current/fsimage
./current/fstime
./current/VERSION
./current/edits
./image/
./image/fsimage
```

3. Check the output.

Warning

If you see a file containing the word *lock*, the NameNode is probably still running. Repeat the preceding steps, starting by shutting down the Hadoop services.

Step 3. Upgrade Managed Components

There are a variety of strategies that you can use to upgrade to the latest version of CDH3.

- You can use your operating system's package management tools to update all packages to the latest version using standard repositories. This approach works well because it minimizes the amount of configuration required and uses the simplest commands. Be aware that this can take a considerable amount of time if you have not upgraded the system recently.
- You can target the `cloudera.com` repository that is added during a typical install, only updating Cloudera components. This limits the scope of updates to be completed, so the process takes less time. This will not work if you created and used a custom repository.
- You can use a custom repository. This process can be more complicated, but enables updating Cloudera components for CDH machines that are not connected to the Internet.

Updating Everything

You can update all components on your system, including Cloudera components. Note that this may take a significant amount of time. To update all packages on your system, use the following command:

- On Red Hat systems:

```
$ sudo yum update
```

- On SLES systems:

```
$ sudo zypper up
```

- On Ubuntu/Debian systems:

```
$ sudo apt-get upgrade
```

Once you complete the process of updating all components, proceed to [Start the Services you Stopped](#).

Updating Cloudera Components Using Default Repositories

To install the new version, you can upgrade from Cloudera's repository by adding an entry to your operating system's package management configuration file. The repository location varies by operating system.

Operating System	Configuration File Repository Entry
Red Hat	http://archive.cloudera.com/redhat/cdh/3/
SLES	http://archive.cloudera.com/sles/11/x86_64/cdh/3/
Debian Squeeze	deb http://archive.cloudera.com/debian/ squeeze-cdh3 contrib
Ubuntu Lucid	deb http://archive.cloudera.com/debian/ lucid-cdh3 contrib
Ubuntu Maverick	deb http://archive.cloudera.com/debian/ maverick-cdh3 contrib

For example, under Red Hat, to upgrade from Cloudera's repository you can run commands such as the following on the CDH host to update only CDH:

```
$ sudo yum clean all
$ sudo yum update --disablerepo='*' --enablerepo=cloudera-cdh3
```

Note

Upgrading CDH in a Cloudera Managed Deployment

– `cloudera-cdh3` is the name of the repository on your system; the name is usually in square brackets on the first line of the repo file, in this example `/etc/yum.repos.d/cloudera-cdh3.repo`:

```
[chris@ca727 yum.repos.d]$ more cloudera-cdh3.repo
[cloudera-cdh3]
...
```

– `yum clean all` cleans up yum's cache directories, ensuring that you download and install the latest versions of the packages.

– If your system is not up to date, and any underlying system components need to be upgraded before this yum update can succeed, yum will tell you what those are.

On a SLES system, use commands like this to clean cached repository information and then update only the CDH components. For example:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://archive.cloudera.com/sles/11/x86_64/cdh/
```

The apt configuration files specify repository information. These files are typically either the `/etc/apt/apt.conf` file or in various files in the `/etc/apt/apt.conf.d/` directory. Review the contents of that file to find the Cloudera repository.

On a Debian/Ubuntu system, use commands like this to clean cached repository information and then update only the CDH components. First:

```
$ sudo apt-get clean
```

After cleaning the cache, use one of the following upgrade commands to upgrade CDH.

Maverick:

```
$ sudo apt-get upgrade -t maverick-cdh3
```

Lucid:

```
$ sudo apt-get upgrade -t lucid-cdh3
```

Squeeze:

```
$ sudo apt-get upgrade -t squeeze-cdh3
```

At the end of this process you should have the most recent versions of the CDH packages installed on the host and you can now proceed to [Start the Services you Stopped](#).

Updating Cloudera Components Using Custom Repositories

You can create your own repository, as described in [Appendix A - Understanding Custom Installation Solutions](#). Creating your own repository is necessary if you are upgrading a cluster that does not have access to the Internet.

If you used a custom repository to complete the installation of current files and now you want to update using a custom repository, the details of the steps to complete the process are variable.

In general, begin by updating any existing custom repository that you will use with the installation files you wish to use. This can be completed in a variety of ways. For example, you might use `wget` to copy the necessary installation files. Once the installation files have been updated, use the custom repository you established for the initial installation to update CDH.

Red Hat

On a Red Hat system ensure you have a custom repo that is configured to use your internal repository. For example, if you could have custom repo file in `/etc/yum.conf.d/` called `cdh_custom.repo` in which you specified a local repository. In such a case, you might use the following commands:

```
$ sudo yum clean all
$ sudo yum update --disablerepo='*' --enablerepo=cdh_custom
```

SLES

On a SLES system, use commands such as the following to clean cached repository information and then update only the CDH components:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://internalserver.example.com/path_to_cdh_repo
```

Specifying the Racks for Hosts

Debian/Ubuntu

Use a command that targets upgrade of your CDH distribution using the custom repository specified in your `apt` configuration files. These files are typically either the `/etc/apt/apt.conf` file or in various files in the `/etc/apt/apt.conf.d/` directory. Information about your custom repository must be included in the repo files. The general form of entries in Debian/Ubuntu is:

```
deb http://server.example.com/directory/ dist-name pool
```

For example, the entry for the default repo is:

```
deb http://us.archive.ubuntu.com/ubuntu/ precise universe
```

On a Debian/Ubuntu system, use commands such as the following to clean cached repository information and then update only the CDH components:

```
$ sudo apt-get clean
$ sudo apt-get upgrade -t your_cdh_repo
```

Step 4. Start the Services you Stopped

You can now start the services that you stopped in Step 1. Proceed as follows:

1. In the Cloudera Manager Admin Console, click the **Services** tab.
2. Click the top **Actions** button that corresponds to the cluster and choose **Start**.

The **Command Details** window shows the progress of starting services.

When **All services successfully started** appears, the task is complete and you may close the **Command Details** window.

Repeat this process for all clusters that you previously stopped.

Specifying the Racks for Hosts

Cloudera Manager includes internal rack awareness scripts, but you must specify the racks where the hosts in your cluster are located. If your cluster contains more than 10 hosts, Cloudera recommends that you specify the rack for each host. HDFS and MapReduce will automatically use the racks you specify.

Note

Cloudera Manager supports multi-level rack specifications. For example, you could specify the rack `/rack1`, but you could also specify `/group5/rack3` to indicate the third rack in the fifth group.

To specify the racks for hosts:

1. Click the **Hosts** tab.
2. Select the host(s) for a particular rack, such as all hosts for `/rack123`. Use shift-click to select multiple hosts at a time.
3. Click **Set Rack**.
4. Enter a rack name or ID that starts with a slash `/`, such as `/rack123` or `/aisle1/rack123`, and then click **Set Rack**.

After assigning racks, consider restarting affected services as described in [Starting, Stopping, and Restarting Services](#). Rack assignments are not automatically updated for running services.

Testing the Installation

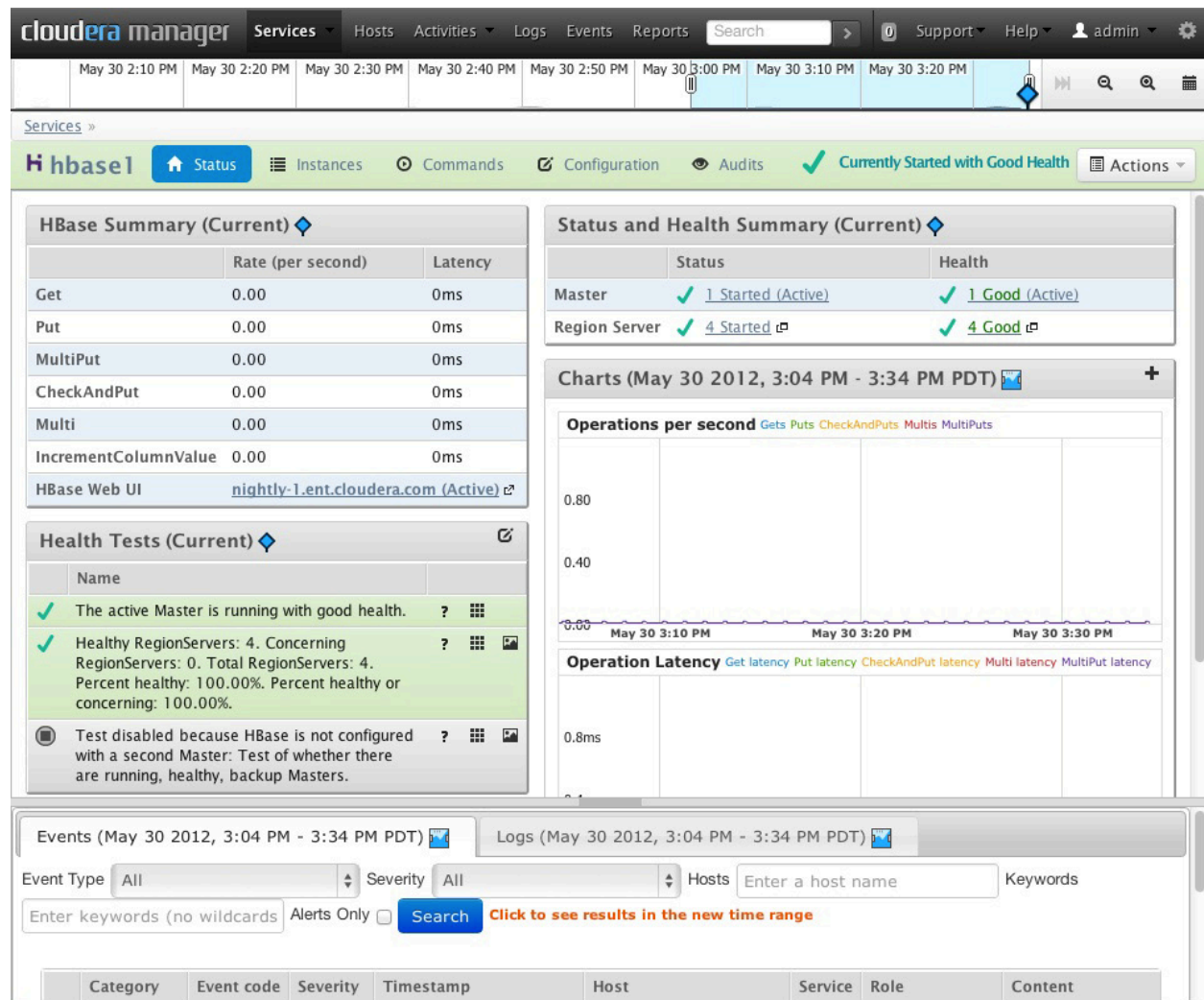
You can check the Cloudera Manager installation on your cluster by logging in to the Cloudera Manager Admin Console and reviewing the contents of the Services tab. The left side of the screen should look something like this:

The screenshot displays the Cloudera Manager Admin Console interface. The top navigation bar includes tabs for Services, Hosts, Activities, Logs, Events, and Reports. Below the navigation bar, the main content area shows the 'Services (Current)' tab for 'Cluster 1 - CDH3'. The services are listed in a table with columns for Name, Type, Status, Health, and Role Counts. All services are shown as 'Started' and 'Good'.

Name	Type	Status	Health	Role Counts
hbase1	HBase	Started	Good	4 Region Servers, 1 Master
hdfs1	HDFS	Started	Good	1 SecondaryNameNode, 1 NameNode, 1 Balancer, 4 DataNodes
hue1	Hue	Started	Good	1 Beeswax Server, 1 Hue Server, 1 Job Designer
mapreduce1	MapReduce	Started	Good	1 JobTracker, 4 TaskTrackers
oozie1	Oozie	Started	Good	1 Oozie Server
zookeeper1	ZooKeeper	Started	Good	1 Server

Enabling the Oozie Web Console

You can also click on each service to see more detailed information. For example, if you click the hbase1 link, you might see something like this:



Enabling the Oozie Web Console

Using an External Database for Oozie

By default, Cloudera Manager uses Derby for Oozie's database. If necessary, you can configure Oozie to use an external database.

Oozie supports the following databases:

- MySQL 5.0 and 5.5
- PostgreSQL 8.4 and 9.0
- Oracle 11g

To configure Cloudera Manager to use an external database as the database for Oozie:

1. In Cloudera Manager, navigate to the **Services** page and click the Oozie service instance.
2. Click **Configuration**.
3. Specify the settings for **Oozie Server database type**, **Oozie Server database name**, **Oozie Server database host**, **Oozie Server database user**, and **Oozie Server database password**.

Note

If you are using MySQL or Oracle, ensure that `mysql-connector-java.jar` or `ojdbc6-11.2.0.3.0.jar` respectively are placed in `/usr/lib/oozie/libext`.

Using an External Database for Hue

By default, Cloudera Manager uses SQLite for Hue's database. If necessary, you can configure Cloudera Manager to use an external database such as MySQL as the database for Hue. The procedure described in this topic illustrates how to migrate the Hue database from the default SQLite installation to another database, such as a MySQL database.

To configure Cloudera Manager to use an external database for Hue

1. Create a new database and grant privileges to a Hue user to manage this database. For example:

```
mysql> create database hue;
Query OK, 1 row affected (0.01 sec)
mysql> grant all on hue.* to 'hue'@'localhost' identified by
'secretpassword';
Query OK, 0 rows affected (0.00 sec)
```

2. Using the Cloudera Manager Admin Console, click the service instance for the Hue database you are reconfiguring.

The Hue service instance page in Cloudera Manager Admin Console appears.

3. Click **Actions** and click **Stop**. Confirm you want to stop the service by clicking **Stop**.
If the Hue service is already stopped, skip this step.
4. Click **Actions** for the Hue service, and click **Dump Database**. Confirm you want to dump the database by clicking **Dump Database**.
5. Click **Configuration**. In the **Category** pane, click the instance of **Database** under **Service-Wide**.
6. Specify the settings for **Hue's Database Type**, **Hue's Database Hostname**, **Hue's Database Port**, **Hue's Database Username**, **Hue's Database Password**, and **Hue's Database Name**.
For example, for a MySQL database on the local host, you might use the following values:

Using Custom Java Home Locations

```
Hue's Database Type = mysql
Hue's Database Hostname = localhost
Hue's Database Port = 3306
Hue's Database Username = hue
Hue's Database Password = secretpassword
Hue's Database Name = hue
```

7. Click **Actions** and click **Synchronize Database**.
8. Delete a table from the Hue database using the following command:

```
$ mysql -uhue -psecretpassword -e "DELETE FROM
hue.django_content_type;"
```

9. In Hue service instance page, click **Actions**, and click **Load Database**. Confirm you want to load the database by clicking **Load Database**.
10. Using the Cloudera Manager Admin Console, click **Actions** for the Hue service, and click **Start**. Confirm you want to start the service by clicking **Start**.

Using Custom Java Home Locations

Java, which Cloudera services require, may be installed at a custom location. In a such a case, Cloudera services may be unable to find this resource. If the JDK cannot be found, services such as MapReduce or HDFS may not start. If you installed the JDK to a custom location, you may need to modify the system configuration to ensure the JDK can be found.

For more information on installing the JDK, see [Install the Java Development Kit Installation for CDH3](#) or [Install the Java Development Kit Installation for CDH4](#).

If Java is installed at a custom location, update system settings so the custom location is used, and then restart Cloudera Manager Agent on the host where the failing service is assigned. Note that you must know the custom java location that was used established during the JDK installation process. Cloudera provides two ways to ensure Cloudera services can find your JDK installation.

Modifying CMF_AGENT_JAVA_HOME

In many cases, modifying the `CMF_AGENT_JAVA_HOME` environment variable is an effective solution for updating the configuration to accommodate a custom `JAVA_HOME`. Modifying the `CMF_AGENT_JAVA_HOME` environment variable enables all services on the host to find the JDK.

To modify the `CMF_AGENT_JAVA_HOME` environment variable

1. Open `/etc/default/cloudera-scm-agent`.

2. Set the `CMF_AGENT_JAVA_HOME` environment variable to the java home in your environment. For example, you might modify the file to include the following line:

```
export CMF_AGENT_JAVA_HOME=/usr/custom_java
```

3. Save and close the `cloudera-scm-agent` file.
4. Restart the Cloudera Manager Agent using the following command:

```
sudo service cloudera-scm-agent restart
```

Modifying Service Settings

You can modify service settings to use your custom `JAVA_HOME`. This is done as an alternative to modifying the `CMF_AGENT_JAVA_HOME` environment variable. Modifying service settings to use a custom `JAVA_HOME` applies to all nodes in the cluster, but you must repeat this process of updating `JAVA_HOME` for all services.

To modify service settings to use your custom `JAVA_HOME`

1. Open the Cloudera Manager Admin Console, click any service that fails to start because the JDK location is misconfigured, and click **Configuration**.
2. Under **Service-Wide**, click **Advanced**.
3. Click the **Value** cell for the **Service Environment Safety Valve** and add your custom java home to the property here.
For example, you might enter the value `JAVA_HOME=/opt/java/jdk6`.
4. Click **Save Changes**.
5. If your deployment includes Cloudera Management services, add your custom `JAVA_HOME` value to `/usr/share/cmf/bin/cmf-server`.
6. In the Cloudera Manager Admin Console, for the service you are configuring, click **Actions** and click **Restart**.
7. Repeat this process for all services that need the updated `JAVA_HOME` value.

To see how Cloudera Manager chooses a default JDK, review the contents of `/usr/lib64/cmf/service/common/cloudera-config.sh`.

Deploying Clients

Client configuration files are generated automatically by Cloudera Manager based on the services you install.

Cloudera Manager deploys these configurations automatically at the end of the installation workflow. You can also download the client configuration files to deploy them manually.

If you modify the configuration of your cluster, you may need to redeploy the client configuration files. For example, a service whose status is "Running with outdated configuration" indicates you may need to redeploy those files.

See [Deploying Client Configuration Files](#) in the *Cloudera Manager User Guide* for information on downloading client configuration files, or redeploying them through Cloudera Manager.

Uninstalling Cloudera Manager

If necessary, you can use the following instructions to uninstall the Cloudera Manager Server and Agents.

Recovering from a failed install

If you have come to this page because your installation did not complete (for example, if it was interrupted by a virtual machine timeout), and you want to proceed with the installation, do the following:

1. Remove files and directories:

```
$ sudo rm -Rf /usr/share/cmf /var/lib/cloudera* /var/cache/yum/cloudera*
```

2. Run the installer again.

Uninstalling Cloudera Manager Server and Agents

Step 1: Stop all services

You must stop all Hadoop services before uninstalling the Cloudera Manager Server and Agents. **To stop all services**

1. In the Cloudera Manager Admin Console, click the **Services** tab.
2. Click the top **Actions** button that corresponds to the cluster and choose **Stop**. Click **Stop** in the confirmation screen.

The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Note

As necessary, repeat the process of stopping services for each cluster.

3. For each Cloudera Management Service entry, click **Actions** and click **Stop**. Click **Stop** in the confirmation screen.

The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Step 2: Uninstall the Cloudera Manager Server.

The commands for uninstalling the Cloudera Manager Server depend on the method you used to install the Cloudera Manager Server. Refer to section below that corresponds to the method you used to install the Cloudera Manager Server. This process described below also removes the embedded PostgreSQL database, if you installed that option. If you did not use the PostgreSQL database, omit the `cloudera-manager-server-db` steps.

If you used the `cloudera-manager-installer.bin` file:

If you installed the Cloudera Manager Server using `cloudera-manager-installer.bin`, run the following command on the Cloudera Manager Server host:

```
$ sudo /usr/share/cmf/uninstall-cloudera-manager.sh
```

Note

If the `uninstall-cloudera-manager.sh` is not installed on your cluster, use the following instructions to uninstall the Cloudera Manager Server.

Uninstalling Cloudera Manager

If you did not use the cloudera-manager-installer.bin file:

If you installed the Cloudera Manager Server using a different installation method such as Puppet, run the following commands on the Cloudera Manager Server host.

1. Stop the Cloudera Manager Server and its database:

```
sudo /sbin/service cloudera-scm-server stop
sudo /sbin/service cloudera-scm-server-db stop
```

2. Uninstall the Cloudera Manager Server and its database:

On Red Hat systems:

```
sudo yum remove cloudera-manager-server
sudo yum remove cloudera-manager-server-db
```

On SLES systems:

```
sudo zypper -n rm --force-resolution cloudera-manager-server
sudo zypper -n rm --force-resolution cloudera-manager-server-db
```

On Debian/Ubuntu systems:

```
sudo apt-get remove cloudera-manager-server
sudo apt-get remove cloudera-manager-server-db
```

Step 3: On all Agent hosts, uninstall CDH and the Cloudera Manager Agents.

On all Agent hosts, run the following commands as root to uninstall the Cloudera Manager Agent and CDH on your cluster machines.

1. Stop the Cloudera Manager agent.

On Red Hat/SLES systems:

```
$ sudo /sbin/service cloudera-scm-agent hard_stop
```

On Debian/Ubuntu systems:

```
$ sudo /usr/sbin/service cloudera-scm-agent hard_stop
```

2. Uninstall the CDH and Cloudera Manager packages:

On Red Hat/SLES systems:

```
$ sudo rpm -e --allmatches $(rpm -qa | grep -ehadoop -ecloudera -ehue
-eoozie -ehbase -eimpala -eflume -ehive)
```

On Debian/Ubuntu systems:

```
$ sudo apt-get remove -y --purge $(dpkg -l | grep -ehadoop -ecloudera
-ehue -eoozie -ehbase -eimpala -eflume -ehive | awk '{ print $2 }' )
```

3. Run this command:

On Red Hat systems:

```
$ sudo yum clean all
```

On SLES systems:

```
$ sudo zypper clean
```

On Debian/Ubuntu systems:

```
$ sudo apt-get clean
```

Step 4: On all Agent hosts, remove all Cloudera Manager data.

This step permanently removes Cloudera Manager data. If you want to be able to access any of this data in the future, you must back it up before removing it. If you used an embedded PostgreSQL database, that data is stored in `/var/lib/cloudera-scm-server-db`. To remove all Cloudera Manager data, run the following command as root:

```
$ sudo rm -Rf /usr/share/cmf /var/lib/cloudera* /var/cache/yum/cloudera*
```

Step 5: On all Agent hosts, kill any running Cloudera Manager and Hadoop processes.

On all Agent hosts, kill any running Cloudera Manager and Hadoop processes:

```
$ for u in hdfs mapred cloudera-scm hbase hue zookeeper oozie hive impala
flume; do sudo kill $(ps -u $u -o pid=); done
```

Troubleshooting Installation and Upgrade Problems

Note

This step should not be necessary if you stopped all the services and the Cloudera Manager agent correctly.

Step 6: Remove the Cloudera Manager lock file.

On all Agent hosts, run this command to remove the Cloudera Manager lock file:

```
$ sudo rm /tmp/.scm_prepare_node.lock
```

Note

After uninstalling Cloudera Manager, you may want to keep or remove the Hadoop data on your cluster. The previous instructions do not remove the Hadoop data. To find out where the Hadoop data directories are located, you can navigate to the **Configuration** tab for the HDFS and MapReduce services in the Cloudera Manager Admin Console, and search for the **Data Directory** property setting.

Troubleshooting Installation and Upgrade Problems

Use the information in this section to troubleshoot installation problems. For information on known issues, see [Known Issues and Work Arounds in Cloudera Manager 4](#).

Symptom	Problem	What to Do
"Failed to start server" reported by cloudera-manager-installer.bin. /var/log/cloudera-scm-server/cloudera-scm-server.log contains a message beginning Caused by: java.lang.ClassNotFoundException: com.mysql.jdbc.Driver...	You may have SELinux enabled.	You can disable SELinux by running <pre>sudo setenforce 0</pre> on the Cloudera Manager Server host. To disable it permanently, edit /etc/selinux/config.
Installation interrupted and installer won't restart.	You need to do some manual cleanup.	See Uninstalling Cloudera Manager .

Symptom	Problem	What to Do
Cloudera Manager Server fails to start. Server is configured to use a MySQL database to store information about service configuration.	Tables may be configured with the ISAM engine.	Make sure that the InnoDB engine is configured, not the MyISAM engine. To check what engine your tables are using, run the following command from the MySQL shell: <code>mysql> show table status;</code>
Agents fail to connect to server. Error 113 ('No route to host') in <code>cloudera-scm-agent.log</code> .	You may have SELinux or <code>iptables</code> enabled.	Check <code>/var/log/cloudera-scm-server/cloudera-scm-server.log</code> on the Server system and <code>/var/log/cloudera-scm-agent/cloudera-scm-agent.log</code> on the Agent system(s). Disable SELinux and <code>iptables</code> .
Some cluster hosts do not appear when you click Find Hosts in install or update wizard.	You may have network connectivity problems.	<ul style="list-style-type: none"> • Make sure all cluster hosts have SSH port 22 open. • Check other common causes of loss of connectivity such as firewalls and interference from SELinux.
"Access denied" in install or update wizard during database configuration for Activity Monitor, Report Manager, or Service Monitor.	Hostname mapping or permissions are incorrectly set up.	<ul style="list-style-type: none"> • For hostname configuration, see Configuring Network Names. • For permissions, make sure the values you enter into the wizard match those you used when you configured the databases. For more information,

Symptom	Problem	What to Do
		see Checking Database Hostnames .
Activity Monitor, Report Manager, or Service Monitor databases fail to start.	MySQL binlog format problem.	Set <code>binlog_format=mixed</code> in <code>/etc/my.cnf</code> . For more information, see this MySQL bug report . See also Installing and Configuring Databases .
You have upgraded the Cloudera Manager Server to 4.1, but now cannot start services.	You may have mismatched versions of the Cloudera Manager Server and Agents.	Make sure you have upgraded the Cloudera Manager Agents on all host machines to 4.1. (The previous version of the Agents will heartbeat with the new version of the Server, but you can't start HDFS and MapReduce with this combination.)
Cloudera services fail to start.	Java may not be installed or may be installed at a custom location.	See Using Custom Java Home Locations for more information on resolving this issue.
The Service Monitor, Activity Monitor, or Host Monitor display a status of BAD in the Cloudera Manager Admin Console. The log file contains the following message: ERROR 1436 (HY000): Thread stack overrun: 7808 bytes used of a 131072 byte stack, and 128000 bytes needed. Use 'mysqld -O thread_stack=#' to specify a bigger stack.	The MySQL thread stack is too small.	<ol style="list-style-type: none"> 1. Update the <code>thread_stack</code> value in <code>my.cnf</code> to 256KB. The <code>my.cnf</code> file is normally located in <code>/etc</code> or <code>/etc/mysql</code>. 2. Restart the mysql service: <div style="border: 1px dashed black; padding: 10px; margin: 10px 0;"> <pre>\$ sudo service mysql restart</pre> </div> 3. Restart the failed service using the Cloudera Manager Admin Console.

Symptom	Problem	What to Do
The Service Monitor or Activity Monitor agents fail to start. Logs contain the error <code>read-committed isolation not safe for the statement binlog format</code> .	The <code>binlog_format</code> is not set to <code>mixed</code> .	Modify the <code>mysql.cnf</code> file to include the entry for <code>binlog format</code> as specified in Installing and Configuring a MySQL Database .
Attempts to reinstall older versions of CDH or Cloudera Manager using Yum fails.	It is possible to install, uninstall, and reinstall CDH and Cloudera Manager. In certain cases, this does not complete as expected. If you install Cloudera Manager 4 and CDH 4, then uninstall Cloudera Manager and CDH, and then attempt to install CDH 3.7 and Cloudera Manager 3.7, incorrect cached information may result in the installation of an incompatible version of the Oracle JDK.	To resolve this issue, you must clear information in the yum cache. Clear cache information as follows: <ol style="list-style-type: none"> 1. Connect to the CDH host. 2. Execute either of the following commands: <pre>\$ yum --enablerepo='*' clean all</pre> or <pre>\$ rm -rf /var/cache/yum/cloudera*</pre> 3. After clearing cache information, proceed with installing CDH 3.7 and Cloudera Manager 3.7.

Checking Database Hostnames

The value you enter into the wizard as the database hostname **must** match the value you entered for the hostname (if any) when you [configured the database](#).

For example, if you entered the following for the Activity Monitor database

Troubleshooting Installation and Upgrade Problems

```
grant all on activity_monitor.* TO 'amon_user'@'localhost' IDENTIFIED BY 'amon_password';
```

the value you enter here for the database hostname must be `localhost`.

On the other hand, if you had entered the following when you created the database

```
grant all on activity_monitor.* TO 'amon_user'@'myhost1.myco.com' IDENTIFIED BY 'amon_password';
```

the value you enter here for the database hostname must be `myhost1.myco.com`.

If you did not specify a host, or used a wildcard to allow access from any host, you can enter either the fully-qualified domain name (FQDN) here, or `localhost`. For example, if you entered

```
grant all on activity_monitor.* TO 'amon_user'@'%' IDENTIFIED BY 'amon_password';
```

the value you enter here for the database hostname can be either the FQDN or `localhost`.

Similarly, if you entered

```
grant all on activity_monitor.* TO 'amon_user' IDENTIFIED BY 'amon_password';
```

the value you enter here for the database hostname can be either the FQDN or `localhost`.

Recovering from Cloudera Manager Host Failures

Cloudera Manager uses databases to store information about the Cloudera Manager system and jobs. If the machine hosting Cloudera Manager fails, it is possible to re-establish the installation if the database information is still available. Database information is typically available for either of the following reasons:

- You backed up the database.
- The database and Cloudera Manager are on separate servers and the database server is still available.

Before beginning this process, find the failed machine's name IP address and hostname. It is not absolutely necessary to have the old Cloudera Manager server name and IP address, but it simplifies the process. You could use a new IP address and hostnames, but this would require updating the configuration of every agent to use this new information. Because it is easier to use the old server name and address in most cases, using a new hostname and IP address is not described.

To restore a Cloudera Manager when the database server is available

1. Identify a new server on which to install Cloudera Manager. Assign the failed Cloudera Manager server's IP address and hostname to the new server.

Note

If the agents were configured with the server's hostname, you do not need to assign the old machine's IP address to the new host. Simply assigning the hostname will suffice.

2. Install Cloudera Manager on a new server, using the method described under [Step 2: Install the Cloudera Manager Server](#).
Do not install the other components, such as CDH and databases, as those should still exist in your environment
3. Update `/etc/cloudera-scm-server/db.properties` with the necessary information so Cloudera Manager server connects to the restored database. This information is typically the database name, database instance name, user name, and password.
4. Start the Cloudera Manager server.

To restore a Cloudera Manager deployment from database backups when the database server is not available

1. Identify a new server on which to install Cloudera Manager. Assign the failed Cloudera Manager server's IP address and hostname to the new server.

Note

If the agents were configured with the server's hostname, you do not need to assign the old machine's IP address to the new host. Simply assigning the hostname will suffice.

2. Install Cloudera Manager on a new server, using whatever method you used before, as described in [Step 2: Install the Cloudera Manager Server](#).
3. Install the database packages on the machines that will host the restored database.
This could be the same server on which you have just installed Cloudera Manager or it could be a different server. The details of which package to install varies based on which database was initially installed on your system. If you used an external MySQL, PostgreSQL, or Oracle database, reinstall that now. If you used the embedded PostgreSQL database, you will need to install the `cloudera-manager-server-db` package as described in [Installing an Embedded PostgreSQL Database](#). After installing that package, you must initialize and start the database as described in [Configuring Your Systems to Support PostgreSQL](#).
4. Restore the backed up databases to the new database installations.

Troubleshooting Installation and Upgrade Problems

5. Update `/etc/cloudera-scm-server/db.properties` with the necessary information so Cloudera Manager server connects to the restored database. This information is typically the database name, database instance name, user name, and password.
6. Start the Cloudera Manager server.

At this point, Cloudera Manager should resume functioning as it did before the failure. Because you restored the database from the backup, the server should accept the running state of the agents, meaning it will not terminate any running Hadoop processes.

This process is similar with secure clusters, though additional files in `/etc/cloudera-scm-server` must be restored in addition to the database.

Changing Embedded PostgreSQL Database Passwords

When Cloudera Manager installs and configures embedded PostgreSQL databases, it creates user accounts and passwords. You may wish to change passwords associated with the embedded PostgreSQL database accounts. To change these passwords, you must know what the original password was, but since the accounts were automatically created, this information is often unknown.

To achieve the goal of changing the password, you can retrieve the user name or password, as well as other database information.

- The Cloudera Manager service connects to the database using the `scm` account. Information about this account is stored in the `db.properties` file.
- The root account for the database is the `cloudera-scm` account. Information about this account is stored in the `generated_password.txt` file.

To find information about the PostgreSQL database user account that the SCM service uses, read the `/etc/cloudera-scm-server/db.properties` file:

```
# cat /etc/cloudera-scm-server/db.properties

Auto-generated by scm_prepare_database.sh
#
Sat Oct 1 12:19:15 PDT 201
#
com.cloudera.cmf.db.type=postgresql
com.cloudera.cmf.db.host=localhost:7432
com.cloudera.cmf.db.name=scm
com.cloudera.cmf.db.user=scm
com.cloudera.cmf.db.password=TXqEESuhj5
```

To find information about the root account for the database, read the `/var/lib/cloudera-scm-server-db/data/generated_password.txt` file:

```
# cat /var/lib/cloudera-scm-server-db/data/generated_password.txt

MnPwGeWaip

The password above was generated by
/usr/share/cmf/bin/initialize_embedded_db.sh (part of the cloudera-scm-
server-db package)
and is the password for the user 'cloudera-scm' for the database in the
current directory.

Generated at Fri Jun 29 16:25:43 PDT 2012.
```

Once you have gathered passwords, you can change the passwords for users, if desired.

Getting Help and Support

Cloudera Support

Cloudera can help you install, configure, optimize, tune, and run Hadoop for large scale data processing and analysis. Cloudera supports Hadoop whether you run our distribution on servers in your own data center, or on hosted infrastructure services such as Amazon EC2, Rackspace, SoftLayer, or VMware's vCloud.

If you are a Cloudera customer, you can:

- Create a [Cloudera Support Ticket](#).
- Visit the [Cloudera Knowledge Base](#).
- Learn how to [register for an account](#) to create a support ticket.

Community Support

Register for the [Cloudera Manager Users group](#).

Register for the [CDH Users group](#).

Report Issues

Cloudera tracks software and documentation bugs and enhancement requests for CDH on issues.cloudera.org. Your input is appreciated, but before filing a request, please search Jira for existing issues and send a message to the CDH user's list, cdh-user@cloudera.org, or the CDH developer's list, cdh-dev@cloudera.org.

Appendix A - Understanding Custom Installation Solutions

Get Announcements about New CDH and Cloudera Manager Releases

Cloudera provides the following public mailing lists that send announcements about new CDH and Cloudera Manager product releases and updates:

- To receive CDH release announcements, subscribe to the [CDH-announce](#) list.
- To receive Cloudera Manager release announcements, subscribe to the [CM-announce](#) list.

Appendix A - Understanding Custom Installation Solutions

Cloudera hosts software repositories that you can use to install products such as Cloudera Manager or CDH. These repositories are effective solutions in most cases, but custom installation solutions are sometimes required. Using the software repositories requires client access over the Internet and results in the installation of the latest version of products. An alternate solution is required if:

- You need to install older product versions. For example, in a CDH cluster, all hosts must run the same CDH version. After completing an initial installation, you may want to add nodes. This could be to increase the size of your cluster to handle larger tasks or to replace older hardware.
- The hosts on which you want to install Cloudera products are not connected to the Internet, so they are unable to reach the Cloudera repository. Some organizations choose to partition parts of their network from outside access. Isolating segments of a network can provide greater assurance that valuable data is not compromised by individuals out of maliciousness or for personal gain. In such a case, the isolated computers are unable to access Cloudera's software repositories for new installations or upgrades.

In both of these cases, using a custom repository solution allows you to meet the needs of your organization, whether that means installing older versions of Cloudera software or installing any version of Cloudera software on machines that are disconnected from the Internet.

Understanding How Package Management Tools Work

Before getting into the details of how to configure a custom package management solution in your environment, it can be useful to have more information about:

- How package management tools work
- Which tools come with which operating systems
- Each tool's configuration files

How Do Packaging and Package Management Tools Interact?

Packages (`rpm` or `deb` files) help ensure that installations complete successfully by encoding each package's dependencies. That means that if you request the installation of a solution, all required elements can be installed at the same time. For example, `hadoop-0.20-hive` depends on `hadoop-0.20`. Package management tools, such as `yum` (RedHat), `zypper` (SUSE), or `apt-get` (Debian/Ubuntu) are

tools support find any required packages. For example, for RedHat, you might enter `yum install hadoop-0.20-hive`. Yum would inform you that the hive package also requires installing `hadoop-0.20` and offer to complete that installation for you. Zypper and apt-get provide similar functionality.

How Do Package Management Tools Find all Available Packages?

Package management tools rely on a list of repositories. Information about the tool's repository is stored in configuration files, the location of which varies according to the particular package management tool.

- Yum on RedHat/CentOS: `/etc/yum.repos.d`
- Zypper on SUSE: `/etc/zypp/zypper.conf`
- Apt-get on Debian/Ubuntu: `/etc/apt/apt.conf` (Additional repositories are specified using `*.list` files in the `/etc/apt/sources.list.d/` directory.)

For example, on a typical CentOS system, you might find:

```
[user@localhost ~]$ ls -l /etc/yum.repos.d/
total 24
-rw-r--r-- 1 root root 2245 Apr 25 2010 CentOS-Base.repo
-rw-r--r-- 1 root root 626 Apr 25 2010 CentOS-Media.repo
```

Inside those `.repo` files are pointers to one or many repositories. There are similar pointers inside configuration files for zypper and apt-get. In the following snippet from `CentOS-Base.repo`, there are two repositories defined: one named `Base` and one named `Updates`. The `mirrorlist` parameter points to a website which has a list of places where this repository can be downloaded.

```
# ...
[base]
name=CentOS-$releasever - Base
mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch
&repo=os
#baseurl=http://mirror.centos.org/centos/$releasever/os/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-5

#released updates
[updates]
name=CentOS-$releasever - Updates
mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch
&repo=updates
#baseurl=http://mirror.centos.org/centos/$releasever/updates/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-5
# ...
```

Appendix A - Understanding Custom Installation Solutions

You can list the repositories you have enabled. The command varies according to operating system:

- RedHat/CentOS: `yum repolist`
- SUSE: `zypper repos`
- Debian/Ubuntu: `Apt-get` does not include a command to display sources, but you can determine sources by reviewing the contents of `/etc/apt/sources.list` and any files contained in `/etc/apt/sources.list.d/`.

The following shows an example of what you might find on a CentOS system in `repolist`:

```
[root@localhost yum.repos.d]$ yum repolist
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
* addons: mirror.san.fastserv.com
* base: centos.eecs.wsu.edu
* extras: mirrors.ecvps.com
* updates: mirror.5ninesolutions.com
repo id                                repo name
status
addons                                CentOS-5 - Addons
enabled:      0
base                                    CentOS-5 - Base
enabled: 3,434
extras                                CentOS-5 - Extras
enabled:      296
updates                                CentOS-5 - Updates
enabled: 1,137
repolist: 4,867
```

How Do I Use Package Management Tools To Install Older Versions of Cloudera Manager?

As previously mentioned, Cloudera Manager installation solutions install the most recent version of the product. This ensures that you install the latest features and bug fixes. While having the latest version of Cloudera Manager is valuable, in some cases it may be necessary to install previous versions. This can be accomplished by downloading previous versions of the software, using that to create your own repository, and then installing from that repository.

Note

The details for each of the steps listed in **To install a previous version of Cloudera Manager** are described in [Creating and Using your own Repository](#).

To install a previous version of Cloudera Manager

1. Download RPM or DEB files containing the desired version of Cloudera Manager.
To find previous versions of Cloudera Manager, see [Cloudera Manager Download Information](#).

Find the correct version and operating system for your environment and download the RPM or DEB.

2. Extract the RPM or DEB and create a repository, as described in the topic [Creating and Using your own Repository](#). The process described at that location includes the following groups of tasks:
 - a. Move the RPM or DEB to a directory of your choosing.
 - b. Create the repository. You might need to create some files and directories manually and you will need to use your operating system's utility to create the files needed for a repository. For example, you might use `createrepo` for CentOS or SUSE or `reprepro` for Debian or Ubuntu.
 - c. Install a web server.
 - d. Publish the repository on the web server.
 - e. Create or modify client configuration files and distribute the files to enable clients to find the new repositories.
3. After completing these steps, you have established the environment necessary to install a previous version of Cloudera Manager. Proceed with the installation process, being sure to target the newly created repository with your package management tool.

Creating and Using your own Repository

Custom repositories leverage package management solutions, which were designed to provide a convenient, efficient means to install software on many computers. This topic describes how to create a local package repository and then how to direct machines in your environment to use that repository.

To create a repository, you simply put the RPMs or DEBs you want to host in one directory. Then complete tasks with `createrepo` or `reprepro`, and then publish the resulting repository on a website.

Step 1: Download Installation Files

Creating a custom repository requires RPM or DEB files. If you already have these files, proceed to the next step. Otherwise, download the appropriate versions of Cloudera Manager. Available versions can be found at [Cloudera Manager Download Information](#).

Step 2: Prepare the RPM or DEB Files

Move the RPMs or DEBs to a directory that you will use for your repository. Suppose you have downloaded the Oracle JDK and want to host it internally to ease installation on various machines. After you have unpacked the RPMs or DEBs, you would have a collection of those files in your directory. For example, for a CentOS system, you might have:

```
$ls -l
total 79392
```

```
-rw-r--r-- 1 user group 70530937 Mar 13 14:42 jdk-6u24-linux-amd64.rpm
-rw-r--r-- 1 user group 499375 Mar 13 14:42 sun-javadb-client-10.6.2-
1.1.i386.rpm
-rw-r--r-- 1 user group 14627 Mar 13 14:42 sun-javadb-common-10.6.2-
1.1.i386.rpm
-rw-r--r-- 1 user group 4080625 Mar 13 14:42 sun-javadb-core-10.6.2-
1.1.i386.rpm
-rw-r--r-- 1 user group 969861 Mar 13 14:42 sun-javadb-demo-10.6.2-
1.1.i386.rpm
-rw-r--r-- 1 user group 4865183 Mar 13 14:42 sun-javadb-docs-10.6.2-
1.1.i386.rpm
-rw-r--r-- 1 user group 201273 Mar 13 14:42 sun-javadb-javadoc-10.6.2-
1.1.i386.rpm
```

Step 3: Create a Repository

You can use `createrepo` or `reprepo` to create a repository. If you don't have `createrepo` or `reprepo` installed, install it using the following command:

- RedHat/CentOS: `yum install createrepo`
- SUSE: `zypper install createrepo`
- Debian/Ubuntu: `apt-get install reprepo`

Creating a CentOS/RHEL/Oracle/SUSE Repository

Having expanded the RPMs to a directory and ensured `createrepo` is installed, you can now create a repository. When you run `createrepo` in the directory that contains the files you will use to create a repo, the program creates an extra directory with some XML files that describing the repository. For example:

```
$ createrepo .
7/7 - sun-javadb-javadoc-10.6.2-1.1.i386.rpm
Saving Primary metadata
Saving file lists metadata
Saving other metadata
$ ls -l
total 79400
-rw-r--r-- 1 user group 70530937 Mar 13 14:42 jdk-6u24-linux-amd64.rpm
drwxr-xr-x 2 user group 4096 Mar 13 14:45 repodata/
-rw-r--r-- 1 user group 499375 Mar 13 14:42 sun-javadb-client-10.6.2-
1.1.i386.rpm
-rw-r--r-- 1 user group 14627 Mar 13 14:42 sun-javadb-common-10.6.2-
1.1.i386.rpm
-rw-r--r-- 1 user group 4080625 Mar 13 14:42 sun-javadb-core-10.6.2-
1.1.i386.rpm
-rw-r--r-- 1 user group 969861 Mar 13 14:42 sun-javadb-demo-10.6.2-
```

```
1.1.i386.rpm
-rw-r--r-- 1 user group 4865183 Mar 13 14:42 sun-javadb-docs-10.6.2-
1.1.i386.rpm
-rw-r--r-- 1 user group 201273 Mar 13 14:42 sun-javadb-javadoc-10.6.2-
1.1.i386.rpm
$ ls -l repodata/
total 64
-rw-r--r-- 1 user group 32997 Mar 13 14:45 filelists.xml.gz
-rw-r--r-- 1 user group 482 Mar 13 14:45 other.xml.gz
-rw-r--r-- 1 user group 2429 Mar 13 14:45 primary.xml.gz
-rw-r--r-- 1 user group 951 Mar 13 14:45 repomd.xml
```

After this command completes, the specified RPMs have been added to your private repository.

Note

Over time, you may need to install multiple versions Cloudera software. To achieve this, you may need to update repository information. Do not create a situation where there are multiple repo files that package management tools could use to install Cloudera software. Either overwrite existing repo files with the new information or create a new repo file and delete the old one. If your machines have multiple repo files, installations may not complete as expected.

Creating a Debian/Ubuntu Repository

Having expanded the DEBs to a directory and ensured `reprepo` is installed, you can now create a repository.

To create a Debian/Ubuntu repository:

1. Create a directory that will host the repository and creating a configuration directory and file. The repo directory can have any location and name. The configuration directory must be named `conf` and be at the root level of the repository directory you create. For example:

```
$ mkdir /tmp/repo
$ mkdir /tmp/repo/conf
```

2. Create a configuration file that contains basic information about the repository. For example, after creating such a file, you could check its contents as follows:

```
$ cat /tmp/repo/conf/distributions
Origin: Cloudera
Label: Cloudera
Suite: stable
Codename: cloudera
Version: 0.1
Architectures: i386 amd64 source
```

```
Components: contrib
Description: Cloudera
```

3. Run `reprepo` on DEB files.

If you had expanded the DEBs to the `/tmp/repo/` directory, you might use the following command:

```
$ find /tmp/repo -name \*.deb -exec reprepo -Vb repo includedeb
cloudera {} \;
```

After this command completes, the specified DEBs have been added to your private repository.

Step 4: Install a Web Server

The repository is typically hosted using HTTP on a machine inside your network. If you already have a web server in your organization, you can move the repository directory, which will include both the RPMs and the `repodata/` subdirectory, to some a location hosted by the web server. If you are able to use an existing web server, then note the URL and skip to [Modifying Clients to Find Repos.](#)

An easy web server to install is the Apache HTTPD.

To install Apache HTTPD:

1. Install Apache HTTPD. You may need to respond to some prompts to confirm you want to complete the installation.

For RedHat/CentOS:

```
[root@localhost yum.repos.d]$ yum install httpd
```

For SUSE:

```
[root@localhost zypp]$ zypper install httpd
```

For Debian/Ubuntu:

```
[root@localhost apt]$ apt-get install httpd
```

2. Start Apache HTTPD:

For RedHat

```
[root@localhost tmp]$ /sbin/service httpd start
Starting httpd: [ OK ]
```

For SUSE

```
[root@localhost tmp]$ /etc/init.d/apache2 start
Starting httpd: [ OK ]
```

For Debian/Ubuntu

```
[root@localhost tmp]$ /etc/init.d/apache2 start
Starting httpd: [ OK ]
```

Step 5: Publish Repository Files

Move your files to the web server directory and modify file permissions. For example, you might use the following commands:

```
[root@localhost tmp]$ mv /tmp/repo /var/www/html
[root@localhost tmp]$ chmod -R ugo+rX /var/www/html/repo
```

After moving files and changing permissions, visit `http://<hostname>:80/repo` to verify that you see an index of files. Note that Apache may have been configured to not show indexes, which is also acceptable.

Modifying Clients to Find Repos

Having established the repository, modify the clients so they find the repository.

For RedHat/CentOS systems:

Create files on client systems with the following information and format, where `hostname` is the name of the web server you created in the previous step:

```
[myrepo]
name=myrepo
baseurl=http://hostname/repo
enabled=1
gpgcheck=0
```

See `man yum.conf` for more details. Put that file into `/etc/yum.repos.d/myrepo.repo` on all of your host machines to enable them to find the packages that you are hosting.

For SLES systems:

Use the `zypper` utility to update client system repo information by issuing the following command:

```
$ zypper addrepo http://hostname/repo alias
```

For Debian/Ubuntu systems:

Add a new `list` file to `/etc/apt/sources.list.d/` on client systems. For example, you might create the file `/etc/apt/sources.list.d/my-private-cloudera-repo.list`. In that file, create an entry to your newly created repository. For example:

```
$ cat /etc/apt/sources.list.d/my-private-cloudera-repo.list
deb http://hostname/repo cloudera
```

After adding your `.list` file, ensure `apt-get` uses the latest information by issuing the following command:

```
$ sudo apt-get update
```

After completing these steps, you have established the environment necessary to install a previous version of Cloudera Manager or install Cloudera Manager to machines that are not connected to the Internet. Proceed with the installation process, being sure to target the newly created repository with your package management tool.