

安装CDH7.1.1

cloud.tencent.com/developer/article/1717910

这篇文章是精通CDP系列的第一文章。

Cloudera与Hortonworks合并后，推出了全球领先的下一代数据平台Cloudera Data Platform。CDP为用户提供一个多云且多功能的平台，用于管理企业数据云。通过统一的平台界面，对数据进行整个生命周期管理，并提供一致的安全和治理服务。

Cloudera DataHub即CDH作为CDP的核心组件，集成了众多前沿数据技术，Ranger、Nifi、Flink、Ozone、Hive3，存储与计算分析，Cloudera Streams Management等等。这些我们会在系列文章中一一讲述。首先我们详细介绍下如何安装最新版CDH7.1.1。

前置准备

网络设置

修改网络IP，并设置为静态

```
vim /etc/sysconfig/network-scripts/ifcfg-ens192
```

主机名设置

使用hostnamectl set-hostname 修改主机名，配置/etc/hosts，并分发到所有节点

```
[root@cdp1 myshell]# cat /etc/hosts
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
#::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.0.213 rhel72.local rhel72
192.168.0.234 cdp1.hadoop.com
192.168.0.235 cdp2.hadoop.com
192.168.0.236 cdp3.hadoop.com
```



禁止Selinux

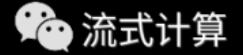
执行命令

```
./batch_cmd.sh node.list "setenforce 0"
```

关闭防火墙

```
./batch_cmd.sh node.list "systemctl disable firewalld"
./batch_cmd.sh node.list "systemctl status firewalld"
```

```
[root@cdp1 myshell]# ./batch_cmd.sh node.list "systemctl status firewalld"
spawn ssh -p 22 root@192.168.0.234 systemctl status firewalld
root@192.168.0.234's password:
● firewalld.service - firewalld - dynamic firewall daemon
  Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset: enabled)
  Active: inactive (dead)
  Docs: man:firewalld(1)
spawn ssh -p 22 root@192.168.0.235 systemctl status firewalld
root@192.168.0.235's password:
● firewalld.service - firewalld - dynamic firewall daemon
  Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset: enabled)
  Active: inactive (dead)
  Docs: man:firewalld(1)
spawn ssh -p 22 root@192.168.0.236 systemctl status firewalld
root@192.168.0.236's password:
● firewalld.service - firewalld - dynamic firewall daemon
  Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset: enabled)
  Active: inactive (dead)
  Docs: man:firewalld(1)
[root@cdp1 myshell]#
```



关闭透明大页面

使用脚本命令批量执行如下操作：

```
echo never > /sys/kernel/mm/transparent_hugepage/defrag
echo never > /sys/kernel/mm/transparent_hugepage/enabled
```

设置开机自动关闭透明大页面：

```
echo "if test -f /sys/kernel/mm/transparent_hugepage/enabled; then">> /etc/rc.d/rc.local
echo "echo never > /sys/kernel/mm/transparent_hugepage/enabled">> /etc/rc.d/rc.local
echo "fi">> /etc/rc.d/rc.local
echo "if test -f /sys/kernel/mm/transparent_hugepage/defrag; then">> /etc/rc.d/rc.local
echo "echo never > /sys/kernel/mm/transparent_hugepage/defrag">> /etc/rc.d/rc.local
echo "fi">> /etc/rc.d/rc.local
```

授权：

使用脚本命令批量执行

```
./batch_cmd.sh node.list "cat /etc/rc.d/rc.local"
./batch_cmd.sh node.list "chmod +x /etc/rc.d/rc.local"
./batch_cmd.sh node.list "ls -l /etc/rc.d/|grep rc.local"
```

```
[root@cdp1 myshell]# ./batch_cmd.sh node.list "ls -l /etc/rc.d/|grep rc.local"
spawn ssh -p 22 root@192.168.0.234 ls -l /etc/rc.d/|grep rc.local
root@192.168.0.234's password:
-rwxr-xr-x 1 root root 716 Jun  5 12:02 rc.local
spawn ssh -p 22 root@192.168.0.235 ls -l /etc/rc.d/|grep rc.local
root@192.168.0.235's password:
-rwxr-xr-x 1 root root 716 Jun  5 12:02 rc.local
spawn ssh -p 22 root@192.168.0.236 ls -l /etc/rc.d/|grep rc.local
root@192.168.0.236's password:
-rwxr-xr-x 1 root root 716 Jun  5 12:02 rc.local
[root@cdp1 myshell]#
```



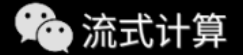
设置swappiness

```
sysctl vm.swappiness=30
echo 10 > /proc/sys/vm/swappiness
```

查看swappiness的值：

```
./batch_cmd.sh node.list "cat /proc/sys/vm/swappiness"
```

```
[root@cdp1 myshell]# ./batch_cmd.sh node.list "cat /proc/sys/vm/swappiness"
spawn ssh -p 22 root@192.168.0.234 cat /proc/sys/vm/swappiness
root@192.168.0.234's password:
30
spawn ssh -p 22 root@192.168.0.235 cat /proc/sys/vm/swappiness
root@192.168.0.235's password:
30
spawn ssh -p 22 root@192.168.0.236 cat /proc/sys/vm/swappiness
root@192.168.0.236's password:
30
```



安装ntp时间同步

所有节点下载安装ntp服务

使用脚本批量执行

```
./batch_cmd.sh node.list "yum -y install ntp"
```

Master节点修改 /etc/ntp.conf文件：

注释部分内容，添加以下内容：

```
server 127.127.1.0
fudge 127.127.1.0 stratum 10
```

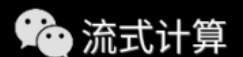
```
restrict ::1

# Hosts on local network are less restricted.
#restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap

# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
#server 0.rhel.pool.ntp.org iburst
#server 1.rhel.pool.ntp.org iburst
#server 2.rhel.pool.ntp.org iburst
#server 3.rhel.pool.ntp.org iburst
server 192.168.0.224 iburst
server 127.127.1.0
fudge 127.127.1.0 stratum 10

#broadcast 192.168.1.255 autokey # broadcast server
#broadcastclient # broadcast client
#broadcast 224.0.1.1 autokey # multicast server
#multicastclient 224.0.1.1 # multicast client
#manycastserver 239.255.254.254 # manycast server
#manycastclient 239.255.254.254 autokey # manycast client

# Enable public key cryptography.
#crypto
```



集群其他节点修改/etc/ntp.conf文件如下：

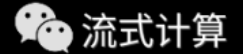
```
# the administrative functions.
restrict 127.0.0.1
restrict ::1

# Hosts on local network are less restricted.
#restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap

# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
#server 0.rhel.pool.ntp.org iburst
#server 1.rhel.pool.ntp.org iburst
#server 2.rhel.pool.ntp.org iburst
#server 3.rhel.pool.ntp.org iburst
server 192.168.0.234

#broadcast 192.168.1.255 autokey # broadcast server
#broadcastclient # broadcast client
#broadcast 224.0.1.1 autokey # multicast server
#multicastclient 224.0.1.1 # multicast client
#manycastserver 239.255.254.254 # manycast server
#manycastclient 239.255.254.254 autokey # manycast client

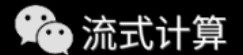
# Enable public key cryptography.
#crypto
```



启动ntpd服务，并设置ntpd服务开机自动启动，查看ntp服务状态：

```
[root@cdp1 myshell]# ./batch_cmd.sh node.list "systemctl status ntpd"
spawn ssh -p 22 root@192.168.0.234 systemctl status ntpd
root@192.168.0.234's password:
● ntpd.service - Network Time Service
   Loaded: loaded (/usr/lib/systemd/system/ntpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2020-06-18 01:23:00 EDT; 1 weeks 0 days ago
   Main PID: 5439 (ntpd)
   CGroup: /system.slice/ntpd.service
           └─5439 /usr/sbin/ntpd -u ntp:ntp -g

Jun 18 01:23:00 cdp1.hadoop.com ntpd[5439]: 0.0.0.0 c012 02 freq_set kernel 0.033 PPM
Jun 18 01:23:02 cdp1.hadoop.com ntpd[5439]: 0.0.0.0 c515 05 clock_sync
Jun 18 01:23:04 cdp1.hadoop.com ntpd[5439]: Listen normally on 3 ens192 192.168.0.234 UDP 123
Jun 18 01:23:04 cdp1.hadoop.com ntpd[5439]: new interface(s) found: waking up resolver
Jun 18 01:29:42 cdp1.hadoop.com ntpd[5439]: 0.0.0.0 0613 03 spike_detect -0.506648 s
Jun 18 01:45:32 cdp1.hadoop.com ntpd[5439]: 0.0.0.0 061c 0c clock_step -0.506637 s
Jun 18 01:45:31 cdp1.hadoop.com ntpd[5439]: 0.0.0.0 0614 04 freq_mode
Jun 18 01:45:32 cdp1.hadoop.com ntpd[5439]: 0.0.0.0 c618 08 no_sys_peer
Jun 18 02:01:15 cdp1.hadoop.com ntpd[5439]: 0.0.0.0 c612 02 freq_set kernel -2.189 PPM
Jun 18 02:01:15 cdp1.hadoop.com ntpd[5439]: 0.0.0.0 c615 05 clock_sync
spawn ssh -p 22 root@192.168.0.235 systemctl status ntpd
root@192.168.0.235's password:
● ntpd.service - Network Time Service
   Loaded: loaded (/usr/lib/systemd/system/ntpd.service; enabled; vendor preset: disabled)
```



查看同步状态：

```
[root@cdp1 myshell]# ./batch_cmd.sh node.list "ntpq -p"
spawn ssh -p 22 root@192.168.0.234 ntpq -p
root@192.168.0.234's password:
     remote           refid      st t when poll reach  delay  offset jitter
=====
*192.168.0.224 LOCAL(0)   11 u 153 1024 377   0.150  -0.027  0.351
LOCAL(0)   .LOCL.    10 l 7d 64    0    0.000   0.000  0.000
spawn ssh -p 22 root@192.168.0.235 ntpq -p
root@192.168.0.235's password:
     remote           refid      st t when poll reach  delay  offset jitter
=====
*cdp1.hadoop.com 192.168.0.224 12 u 993 1024 377   0.287  -0.509  0.387
spawn ssh -p 22 root@192.168.0.236 ntpq -p
root@192.168.0.236's password:
     remote           refid      st t when poll reach  delay  offset jitter
=====
*cdp1.hadoop.com 192.168.0.224 12 u 926 1024 377   0.283  -0.172  0.356
[root@cdp1 myshell]# █
```

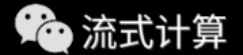


安装httpd

启动httpd，并设置开机自动启动，查看httpd状态

```
[root@cdp1 myshell]# ./batch_cmd.sh node.list "systemctl status httpd"
spawn ssh -p 22 root@192.168.0.234 systemctl status httpd
root@192.168.0.234's password:
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2020-06-18 01:23:07 EDT; 1 weeks 0 days ago
     Docs: man:httpd(8)
           man:apachectl(8)
  Main PID: 5666 (httpd)
   Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
    CGroup: /system.slice/httpd.service
            └─ 5666 /usr/sbin/httpd -DFOREGROUND
               25303 /usr/sbin/httpd -DFOREGROUND
               32165 /usr/sbin/httpd -DFOREGROUND
               32166 /usr/sbin/httpd -DFOREGROUND
               32167 /usr/sbin/httpd -DFOREGROUND
               32168 /usr/sbin/httpd -DFOREGROUND
               32169 /usr/sbin/httpd -DFOREGROUND

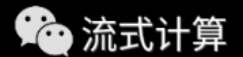
Jun 18 01:23:03 cdp1.hadoop.com systemd[1]: Starting The Apache HTTP Server...
Jun 18 01:23:07 cdp1.hadoop.com systemd[1]: Started The Apache HTTP Server.
Jun 21 03:27:01 cdp1.hadoop.com systemd[1]: Reloading The Apache HTTP Server.
Jun 21 03:27:03 cdp1.hadoop.com systemd[1]: Reloaded The Apache HTTP Server.
```



配置Cloudera Manager的repo源

在/etc/yum.repos.d/目录下新建cm.repo文件，并添加如下内容：

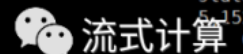
```
[root@cdp1 myshell]# cat /etc/yum.repos.d/cm.repo
[cmrepo]
name = cm7.1.1
baseurl = http://192.168.0.132/cdp7.1.1/cm/
enabled = true
gpgcheck = false
[root@cdp1 myshell]#
```



执行

```
yum clean all
yum repolist
```

```
[root@cdp1 myshell]# yum repolist
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to register.
repo id                repo name                status
Local                  os_repo                  5,152
cloudera-manager       Cloudera Manager, Version 6
cmrepo                  cm7.1.1                  6
repolist: 5,164
```



安装数据库

```
yum -y install mariadb mariadb-server
```

```
[root@cdpl myshell]# yum -y install mariadb mariadb-server
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to register.
Resolving Dependencies
--> Running transaction check
--> Package mariadb.x86_64 1:5.5.60-1.el7_5 will be installed
--> Processing Dependency: mariadb-libs(x86-64) = 1:5.5.60-1.el7_5 for package: 1:mariadb-5.5.60-1.el7_5.x86_64
--> Package mariadb-server.x86_64 1:5.5.60-1.el7_5 will be installed
--> Processing Dependency: perl-DBD-MySQL for package: 1:mariadb-server-5.5.60-1.el7_5.x86_64
--> Running transaction check
--> Package mariadb-libs.x86_64 1:5.5.60-1.el7_5 will be installed
--> Package perl-DBD-MySQL.x86_64 0:4.023-6.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
mariadb x86_64 1:5.5.60-1.el7_5 Local 8.6 M
mariadb-server x86_64 1:5.5.60-1.el7_5 Local 1 M
Installing for dependencies:
mariadb-libs x86_64 1:5.5.60-1.el7_5 Local 758 k
=====
```

启动mariadb数据库：systemctl start mariadb

设置mariadb数据库开机自启：systemctl enable mariadb

查看mariadb数据库服务状态：systemctl status mariadb

```
[root@cdpl myshell]# systemctl start mariadb
[root@cdpl myshell]# systemctl enable mariadb
Created symlink from /etc/systemd/system/multi-user.target.wants/mariadb.service to /usr/lib/systemd/system/mariadb.service.
[root@cdpl myshell]# systemctl status mariadb
● mariadb.service - MariaDB database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2020-06-25 13:11:03 EDT; 12s ago
     Main PID: 15017 (mysqld_safe)
    CGroup: /system.slice/mariadb.service
            └─15017 /bin/sh /usr/bin/mysqld_safe --basedir=/usr
              └─15178 /usr/libexec/mysqld --basedir=/usr --datadir=/var/lib/mysql --plugin-dir=/usr/lib64/mysql/plugin --log-error=/v...

Jun 25 13:11:01 cdpl.hadoop.com systemd[1]: Starting MariaDB database server...
Jun 25 13:11:01 cdpl.hadoop.com mariadb-prepare-db-dir[14984]: Database MariaDB is probably initialized in /var/lib/mysql already.
Jun 25 13:11:01 cdpl.hadoop.com mariadb-prepare-db-dir[14984]: If this is not the case, make sure the /var/lib/mysql is empty ...dir.
Jun 25 13:11:02 cdpl.hadoop.com mysqld_safe[15017]: 200625 13:11:02 mysqld_safe Logging to '/var/log/mariadb/mariadb.log'.
Jun 25 13:11:02 cdpl.hadoop.com mysqld_safe[15017]: 200625 13:11:02 mysqld_safe Starting mysqld daemon with databases from /var/lib/mysql
Jun 25 13:11:03 cdpl.hadoop.com systemd[1]: Started MariaDB database server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@cdpl myshell]#
```

配置mariadb，设置密码为password

/usr/bin/mysql_secure_installation

```
[root@cdpl lib]# /usr/bin/mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

Set root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!
```

创建集群所需的数据库

```
create database metastore default character set utf8;
CREATE USER 'hive'@'%' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON metastore.* TO 'hive'@'%';
FLUSH PRIVILEGES;
```

```
create database cm default character set utf8;
CREATE USER 'cm'@'%' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON cm.* TO 'cm'@'%';
FLUSH PRIVILEGES;
```

```
create database rm default character set utf8;
CREATE USER 'rm'@'%' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON rm.* TO 'rm'@'%';
FLUSH PRIVILEGES;
```

```
create database am default character set utf8;
CREATE USER 'am'@'%' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON am.* TO 'am'@'%';
FLUSH PRIVILEGES;
```

```
create database hue default character set utf8;
CREATE USER 'hue'@'%' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON hue.* TO 'hue'@'%';
FLUSH PRIVILEGES;
```

```
create database oozie default character set utf8;
CREATE USER 'oozie'@'%' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON oozie.* TO 'oozie'@'%';
FLUSH PRIVILEGES;
```

```
create database nav_ms default character set utf8;
CREATE USER 'nav_ms'@'%' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON nav_ms. * TO 'nav_ms'@'%';
FLUSH PRIVILEGES;
```

```
create database nav_as default character set utf8;
CREATE USER 'nav_as'@'%' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON nav_as. * TO 'nav_as'@'%';
FLUSH PRIVILEGES;
```

```
create database ranger default character set utf8;
CREATE USER 'rangeradmin'@'%' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON ranger. * TO 'rangeradmin'@'%';
FLUSH PRIVILEGES;
```

查看创建的数据库：

```

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| am |
| cm |
| hue |
| metastore |
| mysql |
| nav_as |
| nav_ms |
| oozie |
| performance_schema |
| ranger |
| rm |
+-----+
12 rows in set (0.00 sec)

MariaDB [(none)]> █

```



安装JDBC驱动

创建/usr/share/java/文件夹，并将驱动jar包改名为：mysql-connector-java.jar放进该目录：

```

[root@cdp1 lib]# ll /usr/share/java/
total 984
-rwxrwxrwx 1 root root 1006959 Jun  8 14:25 mysql-connector-java.jar
[root@cdp1 lib]# █

```



在master节点上安装JDK

```

[root@cdp1 java]# ll
total 4
drwxr-xr-x 9 root root 4096 Jun  5 13:14 jdk1.8.0_232-cloudera
[root@cdp1 java]# █

```

集群的部署安装

安装并启动cloudera-manager-server

在master节点上安装Cloudera-Manager-Server：

```
yum -y install cloudera-manager-server
```

```

[root@cdp1 myshell]# yum -y install cloudera-manager-server
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to register.
Resolving Dependencies
--> Running transaction check
--> Package cloudera-manager-server.x86_64 0:7.1.1-3274282.el7 will be installed
--> Processing Dependency: cloudera-manager-daemons = 7.1.1 for package: cloudera-manager-server-7.1.1-3274282.el7.x86_64
--> Running transaction check
--> Package cloudera-manager-daemons.x86_64 0:7.1.1-3274282.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                               Arch          Version           Repository        Size
=====
Installing:
cloudera-manager-server                x86_64        7.1.1-3274282.el7 cloudera-manager 12 k
Installing for dependencies:
cloudera-manager-daemons              x86_64        7.1.1-3274282.el7 cloudera-manager 1.4 G
=====
Transaction Summary
=====
Install 1 Package (+1 Dependent package)

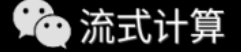
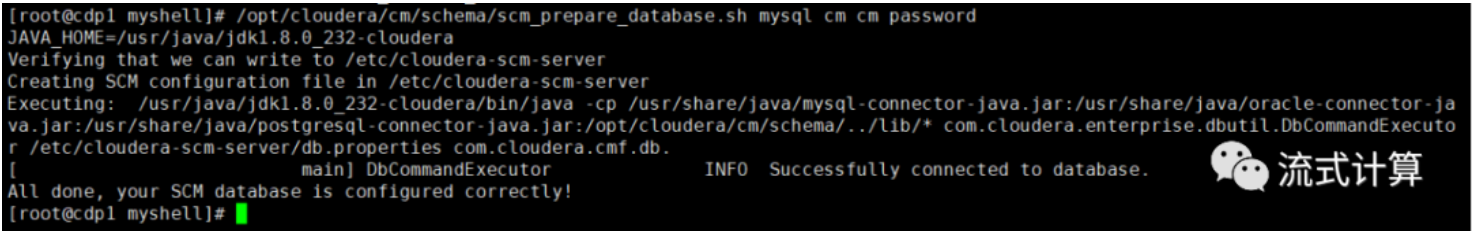
```



初始化数据库

```
/opt/cloudera/cm/schema/scm_prepare_database.sh mysql cm cm password
```

```
[root@cdp1 myshell]# /opt/cloudera/cm/schema/scm_prepare_database.sh mysql cm cm password
JAVA_HOME=/usr/java/jdk1.8.0_232-cloudera
Verifying that we can write to /etc/cloudera-scm-server
Creating SCM configuration file in /etc/cloudera-scm-server
Executing: /usr/java/jdk1.8.0_232-cloudera/bin/java -cp /usr/share/java/mysql-connector-java.jar:/usr/share/java/oracle-connector-java.jar:/usr/share/java/postgresql-connector-java.jar:/opt/cloudera/cm/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!
[root@cdp1 myshell]#
```



执行启动命令

```
systemctl start cloudera-scm-server
```

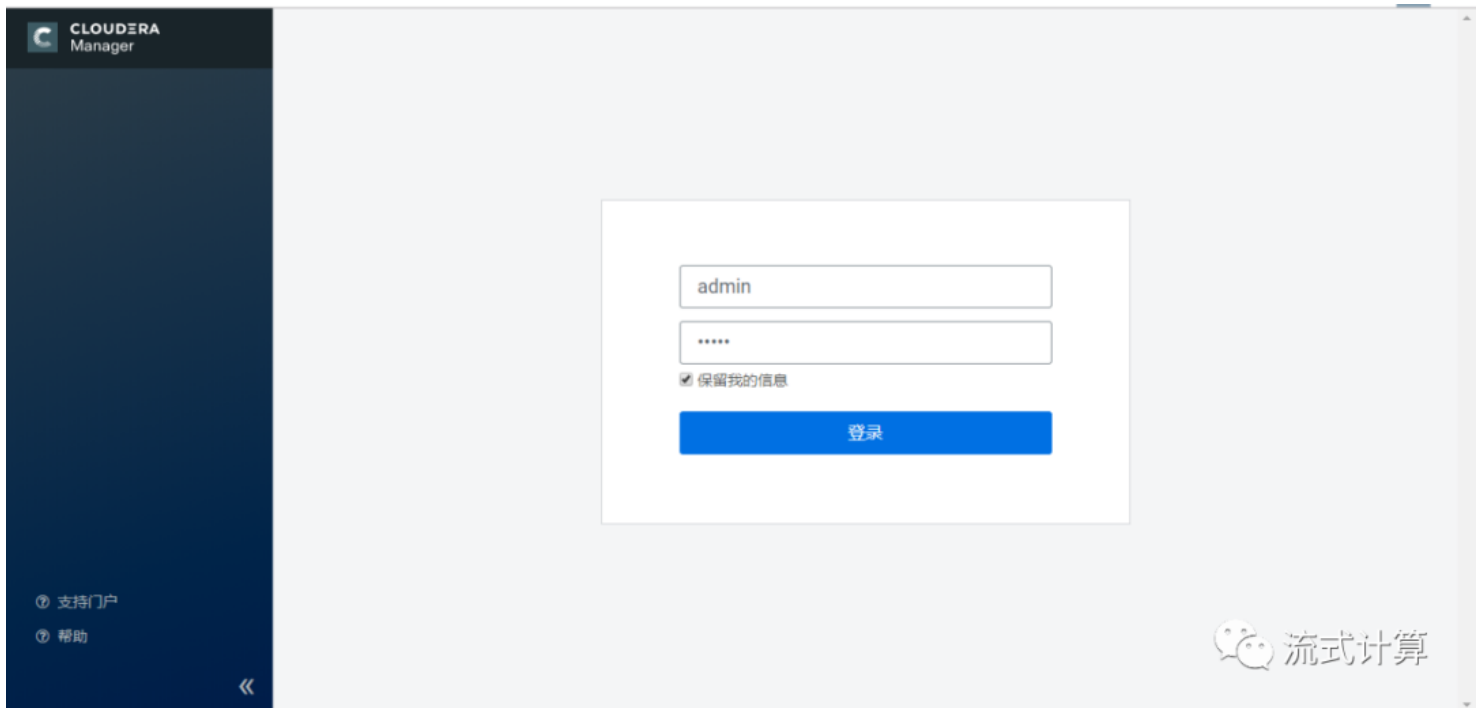
查看7180端口是否被监听

```
netstat -lnpt|grep 7180
```

登录

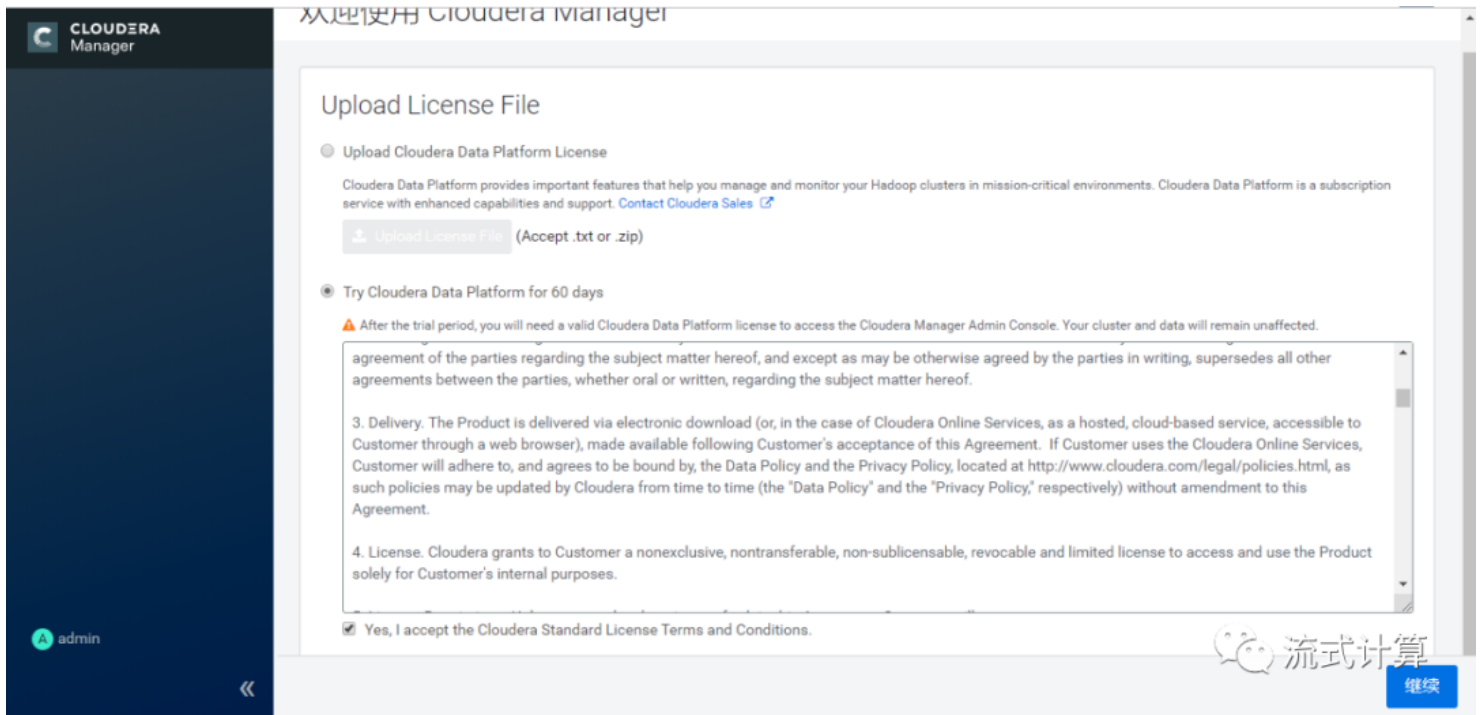
通过网页访问 <http://192.168.0.234:7180/cmf/login>

输入账号admin 密码 admin登录CM

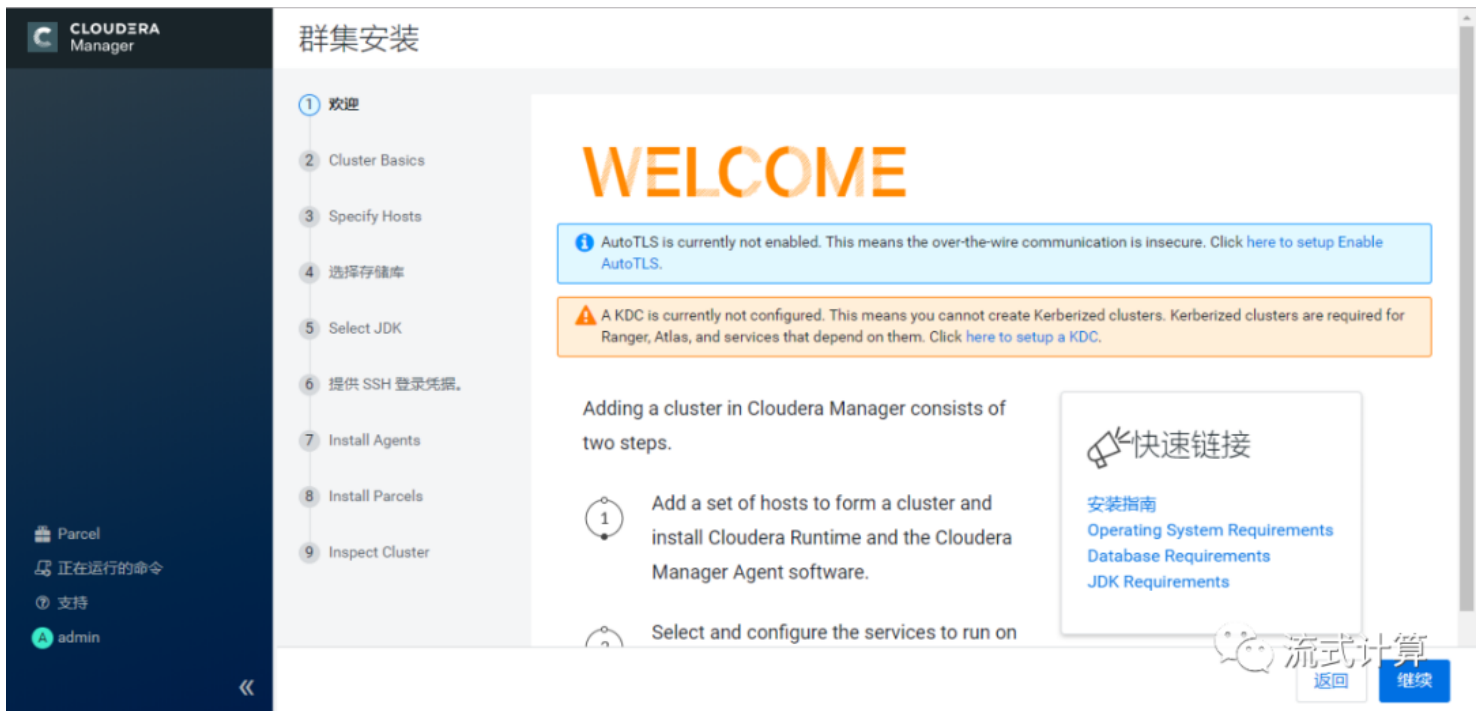


集群安装

选择60天试用版



点击继续



输入集群名称，使用默认的Cluster1

CLOUDERA
Manager


- Parcel
- 正在运行的命令
- 支持
- admin

群集安装

- 1 欢迎
- 2 Cluster Basics
- 3 Specify Hosts
- 4 选择存储库
- 5 Select JDK
- 6 提供 SSH 登录凭据。
- 7 Install Agents
- 8 Install Parcels
- 9 Inspect Cluster


Cluster Basics

群集名称



Regular Cluster

A Regular Cluster contains storage nodes, compute nodes, and other services such as metadata and security collocated in a single cluster.



返回
继续

指定安装主机

CLOUDERA
Manager

- Parcel
- 正在运行的命令
- 支持
- admin

群集安装

- 1 欢迎
- 2 Cluster Basics
- 3 Specify Hosts
- 4 选择存储库
- 5 Select JDK
- 6 提供 SSH 登录凭据。
- 7 Install Agents
- 8 Install Parcels
- 9 Inspect Cluster

Specify Hosts

应使用主机用于标识自身的同一主机名称 (FQDN) 来指定主机。Cloudera 建议包括 Cloudera Manager Server 的主机。这还将启用该主机的运行状况监控。

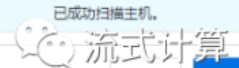
主机名称

提示: 使用模式 搜索主机名称或 IP 地址。

SSH 端口: 搜索

已扫描 3 个主机, 其中 3 个正在运行 SSH。
 单击第一个复选框, 按住 Shift 键并单击最后一个复选框以选择范围。

<input checked="" type="checkbox"/>	已扩展查询 ↑	主机名称 (FQDN)	IP 地址	当前受管	结果	
<input checked="" type="checkbox"/>		192.168.0.234	cdp1.hadoop.com	192.168.0.234	否	已成功扫描主机。
<input checked="" type="checkbox"/>		192.168.0.235	cdp2.hadoop.com	192.168.0.235	否	已成功扫描主机。
<input checked="" type="checkbox"/>		192.168.0.236	cdp3.hadoop.com	192.168.0.236	否	已成功扫描主机。



返回
继续

设置存储库

选择存储库

Cloudera Manager Agent

Cloudera Manager Agent 7.1.1 (#3274282) needs to be installed on all new hosts.

Repository Location Public Cloudera Repository

Ensure the above version is listed in <https://archive.cloudera.com/cm7> and that you have access to that repository. Requires direct Internet access on all hosts.

自定义存储库

示例: `http://LOCAL_SERVER/cloudera-repos/cm7/7.1.1`

Do not include operating system-specific paths in the URL. The path will be automatically derived. Learn more at [How to set up a custom repository](#).

CDH and other software

Cloudera 建议使用 parcel 来代替软件包进行安装, 因为 parcel 可以使服务二进制文件的部署和升级自动化, 让 Cloudera Manager 轻松地管理群集上的软件。如果选择不使用 parcel, 当有软件更新可用时, 将需要您手动升级群集中所有主机上的包, 并会阻止您使用 Cloudera Manager 的滚动升级功能。

选择方法 使用数据包

使用 Parcel (建议) [Parcel Repositories & Network Settings](#) [Parcel 设置](#)

版本 对于此 Cloudera Manager 版本 (7.1.1) 太新的版本不会显示。

Cloudera Runtime 7.1.1-1.cdh7.1.1.p0.2879197

返回 继续

安装JDK以及加密政策

群集安装

Select JDK

Selected Version	Cloudera Runtime 7.1
Supported JDK Version	OpenJDK 8, 11 or Oracle JDK 8, 11

[More details on supported JDK version.](#)

If you plan to use JDK 11, you will need to install it manually on all hosts and then select the **Manually manage JDK** option below.

Manually manage JDK

Please ensure that a supported JDK is already installed on all hosts. You will need to manage installing the unlimited strength JCE policy file, if necessary.

Install a Cloudera-provided version of OpenJDK

By proceeding, Cloudera will install a supported version of OpenJDK version 8.

Install a system-provided version of OpenJDK

By proceeding, Cloudera will install the default version of OpenJDK version 8 provided by the Operating System.

返回 继续

SSH登录，选择用户，并输入密码

提供 SSH 登录凭据。

安装 Cloudera 包需要有主机的 root 访问权限。此安装程序将通过 SSH 连接到您的主机，然后直接以 root 用户身份登录，或者以另一个具有变为 root 用户的无密码 sudo/pbrun 权限的用户身份登录。

登录到所有主机，作为：
 root
 其他用户

对以上选定的用户，您可通过密码或公钥身份验证连接。

身份验证方法：
 所有主机接受相同密码
 所有主机接受相同私钥

输入密码:

确认密码:

SSH 端口:

同时安装的数量:
(同时运行多个安装时将耗费大量的网络带宽和其他系统资源)

流式计算
[返回](#) [继续](#)

安装Agents

群集安装

Install Agents

已成功完成安装。

已成功完成 3 个主机中的 3 个。

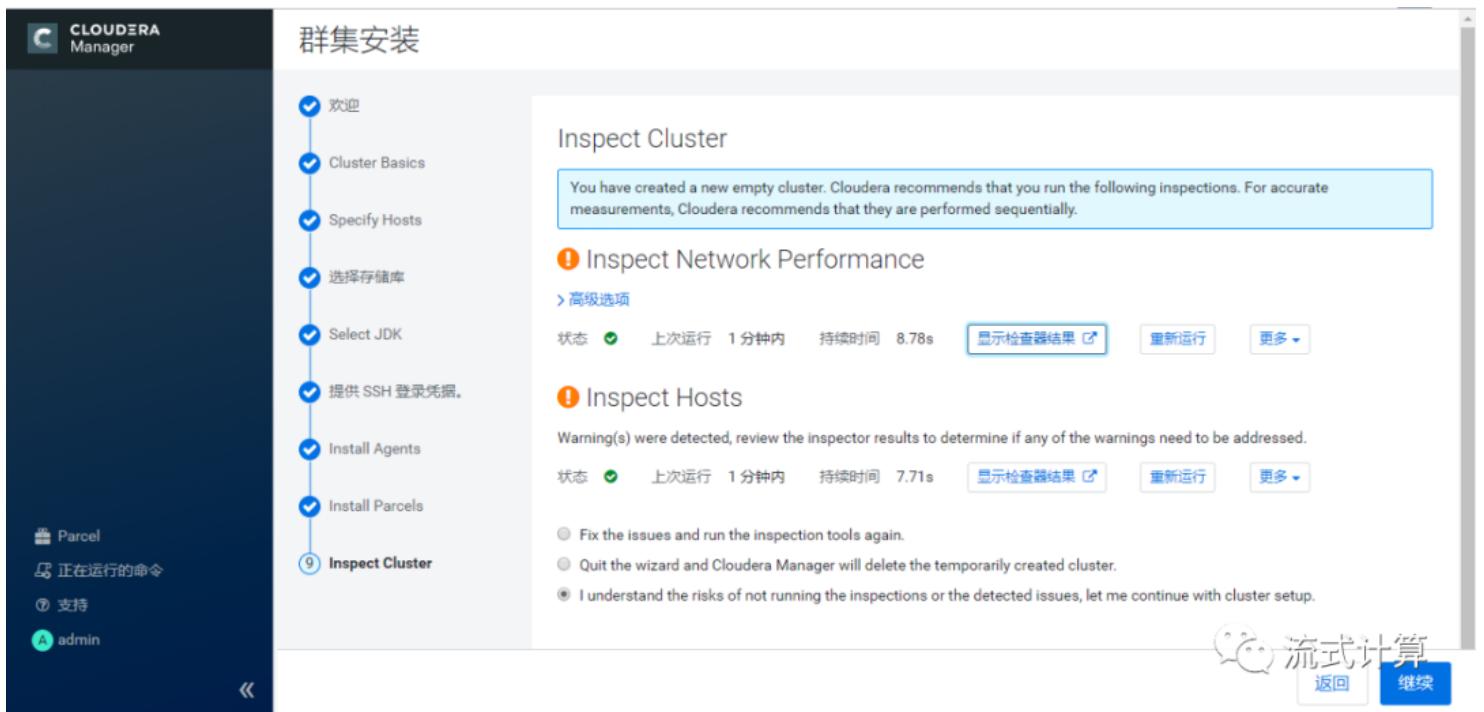
主机名称	IP 地址	进度	状态
cdp1.hadoop.com	192.168.0.234	<div style="width: 100%;"></div>	✓ 已成功完成安装。 详细信息
cdp2.hadoop.com	192.168.0.235	<div style="width: 100%;"></div>	✓ 已成功完成安装。 详细信息
cdp3.hadoop.com	192.168.0.236	<div style="width: 100%;"></div>	✓ 已成功完成安装。 详细信息

流式计算
[返回](#) [继续](#)

安装选定parcel



主机正确性检查，确保全部通过后点完成



组件安装，选择自己要安装的组件

群集设置

- 1 Select Services
- 2 自定义角色分配
- 3 数据库设置
- 4 Enter Required Parameters
- 5 审核更改
- 6 命令详细信息
- 7 汇总

Select Services

选择要安装的服务组合。

- Data Engineering**
 Process, develop, and serve predictive models.
 Services: HDFS, YARN (MR2 Included), YARN Queue Manager, Ranger, Atlas, Hive, Hive on Tez, Spark, Oozie, Hue, and Data Analytics Studio
- Data Mart**
 Browse, query, and explore your data in an interactive way.
 Services: HDFS, Ranger, Atlas, Hive, Impala, and Hue
- Operational Database**
 Real-time insights for modern data-driven business.
 Services: HDFS, Ranger, Atlas, and HBase
- 自定义服务**
 Choose your own services. Services required by chosen services will automatically be included.

流式计算

返回
继续

主机角色分配

群集设置

- 1 Select Services
- 2 自定义角色分配
- 3 数据库设置
- 4 Enter Required Parameters
- 5 审核更改
- 6 命令详细信息
- 7 汇总

自定义角色分配

您可在此处自定义新群集的角色分配。但如果分配不正确（例如，分配到某个主机上的角色太多）会影响服务性能。除非您有特殊需求，如已为特定角色预先选择特定主机，否则 Cloudera 不建议改变分配情况。

还可以按主机查看角色分配。 按主机查看

Kafka

Kafka Broker x 1 新建 <input type="text" value="cdp3.hadoop.com"/>	Kafka MirrorMaker <input type="text" value="选择主机"/>	Kafka Connect <input type="text" value="选择主机"/>
Gateway <input type="text" value="选择主机"/>		

Atlas

Atlas Server x 1 新建 <input type="text" value="cdp3.hadoop.com"/>	Gateway <input type="text" value="选择主机"/>
---	--

HBase

流式计算

返回
继续

CLUSTER Manager

Parcel
正在运行的命令
支持
admin

HBase

Master x 1 新建
cdp2.hadoop.com

RegionServer x 3 新建
与 DataNode 相同

HBase REST Server
选择主机

HBase Thrift Server
选择主机

HDFS

NameNode x 1 新建
cdp1.hadoop.com

SecondaryNameNode x 1 新建
cdp1.hadoop.com

Balancer x 1 新建
cdp1.hadoop.com

HttpFS
选择主机

NFS Gateway
选择主机

DataNode x 3 新建
所有主机

Hive

Gateway x 3 新建
cdp[1-3].hadoop.com

Hive Metastore Server x 1 新建
cdp1.hadoop.com

WebHCat Server
选择主机

HiveServer2
选择主机



CLUSTER Manager

Parcel
正在运行的命令
支持
admin

Cloudera Management Service

Service Monitor x 1 新建
cdp1.hadoop.com

Activity Monitor
选择一个主机

Host Monitor x 1 新建
cdp1.hadoop.com

Reports Manager x 1 新建
cdp1.hadoop.com

Event Server x 1 新建
cdp1.hadoop.com

Alert Publisher x 1 新建
cdp1.hadoop.com

Telemetry Publisher
选择一个主机

Ranger

Ranger Admin x 1 新建
cdp2.hadoop.com

Ranger Usersync x 1 新建
cdp2.hadoop.com

Ranger Tagsync x 1 新建
cdp2.hadoop.com

Solr

Solr Server x 1 新建
cdp2.hadoop.com

ZooKeeper

Server x 1 新建
cdp1.hadoop.com



数据库连接测试

CLUSTER SETUP

群集设置

Select Services

自定义角色分配

3 数据库设置

4 Enter Required Parameters

5 审核更改

6 命令详细信息

7 汇总

Parcel

正在运行的命令

支持

admin

数据库设置

配置和测试数据库连接。如果使用自定义数据库，请先依照 [Installation Guide](#) 的 [Installing and Configuring an External Database](#) 小节创建数据库。

Ranger ✓ Successful

类型: MySQL | 主机名称: cdp1.hadoop.com | 数据库名称: ranger | 用户名: rangeradmin

密码:

Hive ✓ Successful

类型: MySQL | Use JDBC URL Override: 否 | 主机名称: cdp1.hadoop.com | 数据库名称: metastore

用户名: hive | 密码:

Reports Manager ✓ Successful

当前分配的角色运行在 cdp1.hadoop.com 上。

返回 继续

测试成功，点击继续。设置Range相关参数

CLUSTER SETUP

群集设置

Select Services

自定义角色分配

数据库设置

4 Enter Required Parameters

5 审核更改

6 命令详细信息

7 汇总

Parcel

正在运行的命令

支持

admin

Enter Required Parameters

Ranger Admin User Initial Password Ranger (服务范围) 撤销 ?

rangeradmin_user_password:

Ranger Usersync User Initial Password Ranger (服务范围) 撤销 ?

rangerusersync_user_password:

Ranger Tagsync User Initial Password Ranger (服务范围) 撤销 ?

rangertagsync_user_password:

Ranger KMS Keyadmin User Initial Password Ranger (服务范围) 撤销 ?

keyadmin_user_password:

返回 继续

审核更改

群集设置

- Select Services
- 自定义角色分配
- 数据库设置
- Enter Required Parameters
- 5 审核更改**
- 命令详细信息
- 汇总

审核更改

Admin Password: atlas.admin.password | Cluster 1 > Atlas Server Default Group

Enable File Authentication: atlas.authentication.method.file | Cluster 1 > Atlas Server Default Group

Excluded Wire Encryption Protocols: atlas.ssl.exclude.protocols | Cluster 1 > Atlas Server Default Group | TLSv1.2

Initial Solr Shards for Atlas Collections: atlas_solr_shards | Cluster 1 > Atlas Server Default Group | 1

Initial Solr Replication Factor for Collections: atlas_solr_replication_factor | Cluster 1 > Atlas Server Default Group | 1

Atlas Max Heapsize: Cluster 1 > Atlas Server Default Group

返回 继续

集群设置

群集设置

- Select Services
- 自定义角色分配
- 数据库设置
- Enter Required Parameters
- 审核更改
- 6 命令详细信息**
- 汇总

首次运行 命令

状态 ✔ 已完成 | 上下文 Cluster 1 | 6月 26, 12:46:30 凌晨 | 114.74s

Successfully completed 11 steps.

✔ 已完成 1 个步骤 (共 1 个)。

Show All Steps
 Show Only Failed Steps
 Show Only Running Steps

<ul style="list-style-type: none"> Run a set of services for the first time 已成功完成 11 个步骤。 	6月 26, 12:46:33 凌晨	111.43s
<ul style="list-style-type: none"> 依次运行 13 步骤 	6月 26, 12:46:38 凌晨	106.38s

返回 继续

集群安装完成

CLOUDERA Manager

- Parcel
- 正在运行的命令
- 支持
- admin

群集设置

- ✓ Select Services
- ✓ 自定义角色分配
- ✓ 数据库设置
- ✓ Enter Required Parameters
- ✓ 审核更改
- ✓ 命令详细信息
- 7 汇总

汇总

✓ 服务已安装、配置并在群集中运行。

返回
完成



CLOUDERA Manager

搜索

- 群集
- 主机
- 诊断
- 审核
- 图表
- 备份
- 管理

- Parcel
- 正在运行的命令
- 支持
- admin

主页

Switch to Table View
+ 添加

状态 52 所有运行状况问题 | 配置 20 所有最新命令

Cluster 1

Cloudera Runtime 7.1.1 (Parcel)

3 主机	3
Atlas	1
CDP-INFRA-SOLR	2
HBase	9
HDFS	12 1
Hive	2
Hue	2 1
Impala	9 6
Kafka	3
Ranger	3
ZooKeeper	2 1

图表

30分钟 1小时 2小时 6小时 12小时 1天 7d 30d

群集 CPU

Cluster 1, 整个主机中的主机 CPU 使用率 1.5%

群集磁盘 IO

各磁盘中的 1.5M/s 各磁盘中的总磁 0

群集网络 IO

19/19