Chukwa Administration Guide

Table of contents

1. Purpose

The purpose of this document is to help you install and configure Chukwa.

2. Pre-requisites

2.1. Supported Platforms

GNU/Linux is supported as a development and production platform. Chukwa has been demonstrated on Hadoop clusters with 2000 nodes.

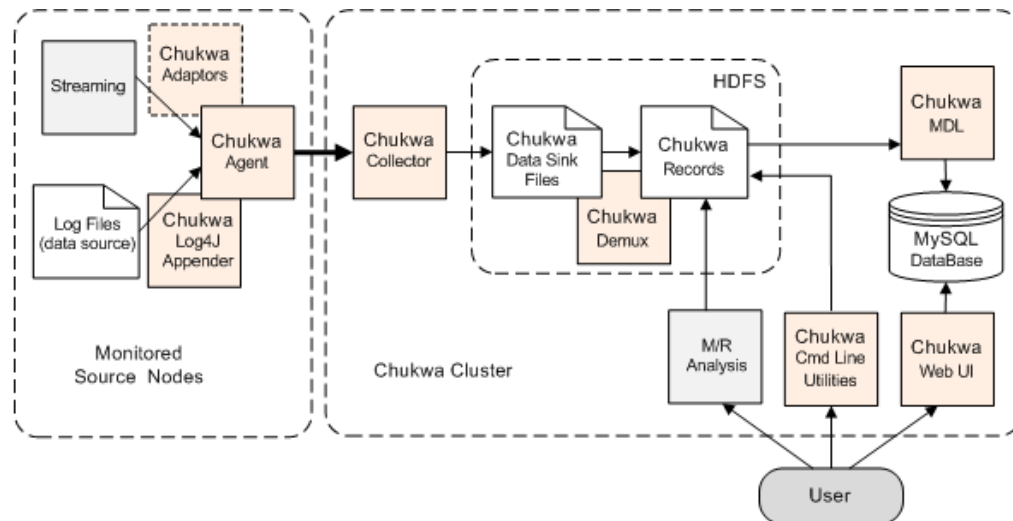## 2.2. Required Software

Required software for Linux include:

1. Java 1.6.10, preferably from Sun, installed (see http://java.sun.com/)
2. MySQL 5.1.30 (see Set Up the Database)
3. Hadoop cluster, installed (see http://hadoop.apache.org/)
4. ssh must be installed and sshd must be running to use the Chukwa scripts that manage remote Chukwa daemons

3. Install Chukwa

Chukwa is installed on:

- A hadoop cluster created specifically for Chukwa (referred to as the Chukwa cluster).
- The source nodes that Chukwa monitors (referred to as the monitored source nodes).

Chukwa can also be installed on a single node, in which case the machine must have at least 16 GB of memory.

3.1. General Install Procedure

1. Select one of the nodes in the Chukwa cluster:

*   Create a directory for the Chukwa installation (Chukwa will set the environment variable **CHUKWA_HOME** to point to this directory during the the install).
*   Move to the new directory.
*   Download and un-tar the Chukwa binary.
*   Configure the components for the Chukwa cluster (see Chukwa Cluster Deployment).
*   Configure the Hadoop configuration files (see Hadoop Configuration Files).
*   Zip the directory and deploy to all nodes in the Chukwa cluster.

2. Select one of the source nodes to be monitored:

*   Create a directory for the Chukwa installation (Chukwa will set the environment variable **CHUKWA_HOME** to point to this directory during the install).
*   Move to the new directory.
*   Download and un-tar the Chukwa binary.
*   Configure the components for the source nodes (see Monitored Source Node Deployment).
*   Configure the Hadoop configuration files (see Hadoop Configuration Files).
*   Zip the directory and deploy to all source nodes to be monitored.

3.2. Chukwa Binary

To get a Chukwa distribution, download a recent stable release of Hadoop from one of the Apache Download Mirrors (see Hadoop Chukwa Releases.

3.3. Chukwa Configuration Files

The Chukwa configuration files are located in the CHUKWA_HOME/conf directory. The configuration files that you modify are named ***.template.** To set up your Chukwa installation (configure various components), copy, rename, and modify the *.template files as necessary. For example, copy the chukwa-collector-conf.xml.template file to a file named chukwa-collector-conf.xml and then modify the file to include the cluster/group name for the source nodes.

The **default.properties** file contains default parameter settings. To override these default settings use the **build.properties** file. For example, copy the TODO-JAVA-HOME environment variable from the default.properties file to the build.properties file and change the setting.

3.4. Hadoop Configuration Files

The Hadoop configuration files are located in the HADOOP_HOME/conf directory. To setup Chukwa, you need to change some of the hadoop configuration files.

1. Copy CHUKWA_HOME/conf/hadoop-log4j.properties file to HADOOP_HOME/conf/log4j.properties
2. Copy CHUKWA_HOME/conf/hadoop-metrics.properties file to HADOOP_HOME/conf/hadoop-metrics.properties
3. Edit HADOOP_HOME/conf/hadoop-metrics.properties file and change @CHUKWA_LOG_DIR@ to your actual CHUKWA log dirctory (ie, CHUKWA_HOME/var/log)
4. ln -s HADOOP_HOME/conf/hadoop-site.xml CHUKWA_HOME/conf/hadoop-site.xml

4. Chukwa Cluster Deployment

This section describes how to set up the Chukwa cluster and related components.

4.1. 1. Set the Environment Variables

Edit the CHUKWA_HOME/conf/chukwa-env.sh configuration file:

- Set JAVA_HOME to your Java installation.
- Set HADOOP_JAR to $CHUKWA_HOME/hadoopjars/hadoop-0.18.2.jar
- Set CHUKWA_IDENT_STRING to the Chukwa cluster name.

4.2. 2. Set Up the Hadoop jar File

Do the following:

```
cp $HADOOP_HOME/lib hadoop-*-core.jar file $CHUKWA_HOME/hadoopjars
```

4.3. 3. Configure the Collector

Edit the CHUKWA_HOME/conf/chukwa-collector-conf.xml configuration file.

Set the writer.hdfs.filesystem property to the HDFS root URL.

4.4. 4. Set Up the Database

Set up and configure the MySQL database.

4.4.1. Install MySQL

Download MySQL 5.1 from the MySQL site.

```
tar fxvz mysql-*.tar.gz -C $CHUKWA_HOME/opt
cd $CHUKWA_HOME/opt/mysql-*
```

Configure and then copy the my.cnf file to the CHUKWA_HOME/opt/mysql-* directory:

```
./scripts/mysql_install_db
./bin/mysqld_safe&
./bin/mysqladmin -u root create <clustername>
./bin/mysql -u root <clustername> < $CHUKWA_HOME/conf/database_create_table
```

Edit the CHUKWA_HOME/conf/jdbc.conf configuration file.

Set the clustername to the MYSQL root URL:

```
<clustername>=jdbc:mysql://localhost:3306/<clustername>?user=root
```

Download the MySQL Connector/J 5.1 from the MySQL site, and place the jar file in $CHUKWA_HOME/lib.

4.4.2. Set Up MySQL for Replication

Start the MySQL shell:

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

From the MySQL shell, enter these commands (replace <username> and <password> with

actual values):

```
GRANT REPLICATION SLAVE ON *.* TO '<username>'@'%' IDENTIFIED BY
'<password>';
FLUSH PRIVILEGES;
```

4.4.3. Migrate Existing Data From Chukwa 0.1.1

Start the MySQL shell:

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

From the MySQL shell, enter these commands (replace <database_name> with an actual value):

```
use <database_name>
source /path/to/chukwa/conf/database_create_table.sql
source /path/to/chukwa/conf/database_upgrade.sql
```

4.5. 5. Start the Chukwa Processes

The Chukwa startup scripts are located in the CHUKWA_HOME/tools/init.d directory.

- Start the Chukwa collector script (execute this command only on those nodes that have the Chukwa Collector installed):

```
CHUKWA_HOME/tools/init.d/chukwa-collector start
```
- Start the Chukwa data processors script (execute this command only on the data processor node):

```
CHUKWA_HOME/tools/init.d/chukwa-data-processors start
```
- Create down sampling daily cron job:

```
CHUKWA_HOME/bin/downSampling.sh --config <path to chukwa conf> -n add
```

4.6. 6. Validate the Chukwa Processes

The Chukwa status scripts are located in the CHUKWA_HOME/tools/init.d directory.

- To obtain the status for the Chukwa collector, run:

```
CHUKWA_HOME/tools/init.d/chukwa-collector status
```
- To verify that the data processors are functioning correctly:

```
Visit the Chukwa hadoop cluster's Job Tracker UI for job status.
Refresh to the Chukwa Cluster Configuration page for the Job Tracker URL.
```

4.7. 7. Set Up HICC

The Hadoop Infrastructure Care Center (HICC) is the Chukwa web user interface. To set up HICC, do the following:

- Download apache-tomcat 6.0.18+ from Apache Tomcat and decompress the tarball to CHUKWA_HOME/opt.
- Copy CHUKWA_HOME/hicc.war to apache-tomcat-6.0.18/webapps.
- Start up HICC by running:

```
CHUKWA_HOME/bin/hicc.sh start
```
- Point your favorite browser to: http://<server>:8080/hicc

5. Monitored Source Node Deployment

This section describes how to set up the source nodes.

5.1. 1. Set the Environment Variables

Edit the CHUKWA_HOME/conf/chukwa-current/chukwa-env.sh configuration file:

- Set JAVA_HOME to the root of your Java installation.
- Set other environment variables as necessary.

```
export JAVA_HOME=/path/to/java
export HADOOP_HOME=/path/to/hadoop
export chuwaRecordsRepository="/chukwa/repos/"
export JDBC_DRIVER=com.mysql.jdbc.Driver
export JDBC_URL_PREFIX=jdbc:mysql://
```

5.2. 2. Configure the Agent

Edit the CHUKWA_HOME/conf/chukwa-current/chukwa-agent-conf.xml configuration file.

Enter the cluster/group name that identifies the monitored source nodes:

```
 <property>
    <name>chukwaAgent.tags</name>
    <value>cluster="demo"</value>
    <description>The cluster's name for this agent</description>
  </property>
```
Edit the CHUKWA_HOME/conf/chukwa-current/agents configuration file.

Create a list of hosts that are running the Chukwa agent:

```
localhost
localhost
localhost
```

Edit the CHUKWA_HOME/conf/collectors configuration file.

The Chukwa agent needs to know about the Chukwa collectors. Create a list of hosts that are running the Chukwa collector:

- This ...

```
<collector1HostName>
<collector2HostName>
<collector3HostName>
```

- Or this ...

```
http://<collector1HostName>:<collector1Port>/
http://<collector2HostName>:<collector2Port>/
http://<collector3HostName>:<collector3Port>/
```

5.3. 3. Configure Adaptors

Edit the CHUKWA_HOME/conf/initial_adaptors configuration file.

Define the default adaptors:

```
add
org.apache.hadoop.chukwa.datacollection.adaptor.filetailer.CharFileTailingAdaptorUTF8Ne
SysLog 0 /var/log/messages 0
```

Make sure Chukwa has a Read access to /var/log/messages.

5.4. 4. Start the Chukwa Processes

Start the Chukwa agent and system metrics processes on the monitored source nodes.

The Chukwa startup scripts are located in the CHUKWA_HOME/tools/init.d directory.

Run both of these commands on all monitored source nodes:

- Start the Chukwa agent script:

```
CHUKWA_HOME /tools/init.d/chukwa-agent start
```

- Start the Chukwa system metrics script:

```
CHUKWA_HOME /tools/init.d/chukwa-system-metrics start
```

5.5. 5. Validate the Chukwa Processes

The Chukwa status scripts are located in the CHUKWA_HOME/tools/init.d directory.

Verify that that agent and system metrics processes are running on all source nodes:

- To obtain the status for the Chukwa agent, run:

```
CHUKWA_HOME/tools/init.d/chukwa-agent status
```
- To obtain the status for the system metrics, run:

```
CHUKWA_HOME/tools/init.d/chukwa-system-metrics status
```

6. Troubleshooting Tips

6.1. UNIX Processes For Chukwa Agents

The system metrics data loader process names are uniquely defined by:

- org.apache.hadoop.chukwa.inputtools.plugin.metrics.Exec sar -q -r -n ALL 55
- org.apache.hadoop.chukwa.inputtools.plugin.metrics.Exec iostat -x -k 55 2
- org.apache.hadoop.chukwa.inputtools.plugin.metrics.Exec top -b -n 1 -c
- org.apache.hadoop.chukwa.inputtools.plugin.metrics.Exec df -l
- org.apache.hadoop.chukwa.inputtools.plugin.metrics.Exec CHUKWA_HOME/bin/../bin/netstat.sh

The Chukwa agent process name is identified by:

- org.apache.hadoop.chukwa.datacollection.agent.ChukwaAgent

Command line to use to search for the process name:

- ps ax | grep org.apache.hadoop.chukwa.datacollection.agent.ChukwaAgent

6.2. UNIX Processes For Chukwa Collectors

Chukwa Collector name is identified by:

- **org.apache.hadoop.chukwa.datacollection.collector.CollectorStub**

6.3. UNIX Processes For Chukwa Data Processes

Chukwa Data Processors are identified by:

- org.apache.hadoop.chukwa.extraction.demux.Demux
- org.apache.hadoop.chukwa.extraction.database.DatabaseLoader
- org.apache.hadoop.chukwa.extraction.archive.ChukwaArchiveBuilder

The processes are scheduled execution, therefore they are not always visible from the process list.

6.4. Checks for MySQL Replication

At slave server, MySQL prompt, run:

```
show slave status\G
```

Make sure both **Slave_IO_Running** and **Slave_SQL_Running** are both "Yes".

Things to check if MySQL replication fails:

- Make sure grant permission has been enabled on master MySQL server.
- Check disk space availability.
- Check Error status in slave status.

To reset MySQL replication, run these commands on MySQL:

```
STOP SLAVE;
CHANGE MASTER TO
  MASTER_HOST='hostname',
  MASTER_USER='username',
  MASTER_PASSWORD='password',
  MASTER_PORT=3306,
  MASTER_LOG_FILE='master2-bin.001',
  MASTER_LOG_POS=4,
  MASTER_CONNECT_RETRY=10;
START SLAVE;
```

6.5. Checks For Disk Full

If anything is wrong, use /etc/init.d/chukwa-agent and
CHUKWA_HOME/tools/init.d/chukwa-system-metrics stop to shutdown Chukwa. Look at
agent.log and collector.log file to determine the problems.

The most common problem is the log files are growing unbounded. Set up a cron job to
remove old log files:

```
 0 12 * * * CHUKWA_HOME/tools/expiration.sh 10 !CHUKWA_HOME/var/log nowait
```

This will set up the log file expiration for CHUKWA_HOME/var/log for log files older than
10 days.

6.6. Emergency Shutdown Procedure

If the system is not functioning properly and you cannot find an answer in the Administration
Guide, execute the kill command. The current state of the java process will be written to the
log files. You can analyze these files to determine the cause of the problem.

```
kill -3 <pid>
```