Oozie Installation Guide

A guide to install to Oozie

edureka!

Version 2.2

Oozie Installation Guide

Table of Contents

Pre-requisites:

→ JDK 1.6 or higher

→ Maven tool for building oozie

Basic Information:

→ Oozie is a workflow scheduler/management tool which allows managing and scheduling the Pig/Mapreduce jobs.

→ It can at a time work with either MRV1 or YARN.

→ Uses embedded Tomcat server for deploying war file necessary for GUI, and Embedded Derby DB for maintaining job related information.

→ You may choose to use a different Tomcat server apart from the one which comes with the default to build to deploy the Oozie application, in order to do that set the CATALINA_OPTS in the oozie-env.sh to the directory where your tomcat is installed.

→ The GUI of oozie listens on http port 11000 by default, which again is configurable in the oozie-env.sh file.

Steps to Install:

1. Download a tar.gz file from the web. (http://mirror.nexcess.net/apache/oozie/4.0.1/oozie-4.0.1.tar.gz) you will have a file named oozie-4.0.1.tar.gz at the download location. If using a CLI use **wget** command to download.
2. Extract the file using the following command. We'll call this OOZIE_EXTRACT_LOCATION

   tar -xvf oozie-4.0.1.tar.gz
3. cd oozie-4.0.1/bin
4. Execute the ***mkdistro.sh*** script. This script may take multiple parameters.

   ./mkdistro.sh -DskipTests -Dhadoop.version=2.0.0-alpha

   If the -Dhadoop.version parameter is skipped it will build oozie against Hadoop 1. Currently only 2.0.0-alpha version of hadoop is supported by Oozie. Make sure you give -DskipTests since it will expedite the build process and avoid any errors which result in build failure while running tests.

5. After the successful execution of the above command, navigate to the distro directory in the extracted tar location.

   cd $extracted_Oozie_location/distro/target

6. You will see a tar file created with the following name ***oozie-4.0.1-distro.tar.gz***

7. Copy the tar file to the location where you want to install oozie.

8. Extract the tar file ***tar -xvf oozie-4.0.1-distro.tar.gz*** , you will now have a folder oozie-4.0.1-distro which has the oozie binaries. We'll call this OOZIE_HOME

9. Create a soft link to this directory, so that any changes to the directory name or an upgrade will not result in changing the location in multiple files, it will only result in changing the softlink pointing.

10. ***ls -s oozie-4.0.1-distro oozie***

11. Make available the binaries of Oozie in the PATH variable either by doing an export (will only last till the session persist) or modify the .bashrc file (will only apply to one user) or make an entry in the /etc/profile file (root priviledges required, applicable for all the users)

    export PATH=$PATH:$OOZIE_HOME/bin

    where OOZIE_HOME=location where you have extracted the distibution, step 8.

    If you are editing the .bashrc or profile file do not forget to do a ***source $filename*** which would reload the PATH variable with the new added changes.

12. The next step is to make sure Oozie has all the related jars available to it.

    Create a directory libext in $OOZIE_HOME, and copy all the jar file residing in the corresponding distribution in

    cp $OOZIE_EXTRACT_LOCATION/hadooplibs/hadoop-{$version}/target/hadooplibs/hadooplib-1.1.1.oozie-4.0.1/ $OOZIE_HOME/libext

    where {$version} should be replaced by the hadoop version against which oozie was built i.e. 1 or 2

13. ***cd $OOZIE_HOME/libext*** here you will find all hadoop related jars i.e. client and core jars

14. By default oozie is built against hadoop-1.1.1 so if you are using any other distribution of hadoop 1 then you will have to replace the jars.

    cp $HADOOP_HOME/hadoop-c*.jar $OOZIE_HOME/libext

    This will copy the hadoop-client and hadoop-core jars, make sure you delete or move the old jars which may result to conflict if kept at the same location.

15. Another tool which is required in order to run Oozie UI is the ExtJS, download the extJs 2.2 zip file and paste it in the $OOZIE_HOME/libext folder. No need to extract the file.

16. Now we look at the configuration changes required on the hadoop site. Edit the $HADOOP_HOME/conf/core-site.xml and add the following properties:

```
<property>
  <name>hadoop.proxyuser.[OOZIE-USER].hosts</name>
  <value>hostname</value>
  <description>The hostname on which you plan to start the oozie server</description>
</property>
<property>
  <name>hadoop.proxyuser.[OOZIE-USER].groups</name>
  <value>groupname</value>
  <description>The group in which you the OOZIE-USER belongs</description>
</property>
```

Restart your Hadoop services.

17. Prepare the war to be deployed so that the GUI can be seen:

```
$OOZIE_HOME/bin/oozie-setup.sh prepare-war
```

18. Create database for Oozie to use:

```
$OOZIE_HOME/bin/ooziedb.sh create -sqlfile $filename -run
```

$filename = name of any file in which the schema will be written and then executed.

19. Start the Oozie server:

```
$OOZIE_HOME/bin/oozied.sh start
```

20. Open a browser and go to link to verify if Oozie server has started.

```
http://$OOZIE_SERVER:11000/oozie
```

NOTE: Nowhere have we created a variable called OOZIE_HOME neither using export/.bashrc/profile file. The variable $OOZIE_HOME used above is only for representational purposes. Oozie will automatically calculate its own home i.e. OOZIE_HOME

In order to submit different kind of jobs to Oozie i.e. Pig, hive, mapreduce you will have to set each projects home. i.e. HADOOP_HOME, HBASE_HOME, HIVE_HOME, PIG_HOME etc.