Setting Up Pentaho CE Server

Document Author: juha-matti.varjonen@tyks.fi

Installing Pentaho CE

Install (or clone) fresh copy of Ubuntu server acconding to the instruction here: Building a new Research VM .

1) Prepare JAVA

   pentaho@pentahoce:/$ sudo apt-get install zip openjdk-8-jre openjdk-8-jdk

   Setup up JAVA_HOME:

   pentaho@pentahoce:/$ sudo su root -c "echo 'export JAVA_HOME="/usr/lib/jvm/java-8-openjdk-amd64"' >> /etc/environment"

2) Create a dedicated 'pentaho' user and give sudo access rights to 'pentaho' account.

3) Install the PostgreSQL server

   pentaho@pentahoce:/$ sudo add-apt-repository "deb https://apt.postgresql.org/pub/repos/apt/ trusty-pgdg main" pentaho@pentahoce:/$ sudo apt-get install postgresql-9.6
   pentaho@pentahoce:/$ sudo apt-get install pgadmin3

4) Edit pg_hba.conf file

Database administrative login by Unix domain socket

   local all postgres md5 local all all md5

5) Start PostgreSQL server

   pentaho@pentahoce:/$ sudo service postgresql start

6) Install Pentaho CE Server

   pentaho@pentahoce:~/tmp$ wget https://downloads.sourceforge.net/project/pentaho/Business%20Intel
   server-ce-7.1.0.0-12.zip-server-ce-7.1.0.0-12.zip

   Unzip the file to the '/opt/pentaho/'

   pentaho@pentahoce:~/tmp$ sudo unzip pentaho-server-ce-7.1.0.0-12.zip -d /opt/pentaho

   Create a pentaho-server sympolic link under '/opt/pentaho/' so the version control could be little bit easier in future

   pentaho@pentahoce:/opt/pentaho$ sudo mv pentaho-server/ pentaho-server-ce-7.1.0.0-12/
   pentaho@pentahoce:/opt/pentaho$ sudo chown -R pentaho pentaho-server-ce-7.1.0.0-12/
   pentaho@pentahoce:/opt/pentaho$ sudo chgrp -R pentaho pentaho-server-ce-7.1.0.0-12/
   pentaho@pentahoce:/opt/pentaho/pentaho-server$ sudo ln -s pentaho-server-ce-7.1.0.0-12/ /opt/pentaho/pentaho-server

7) Run the pentaho -postgresql scripts and if asked give the default password = 'password'

   pentaho@pentahoce:/opt/pentaho/pentaho-server$ sudo -u postgres psql -a -f /opt/pentaho/pentaho-server/data/postgresql/create_quartz_postgresql.sql pentaho@pentahoce:/opt/pentaho/pentaho-server$ sudo -u postgres psql -a -f /opt/pentaho/pentaho-server/data/postgresql/create_repository_post

pentaho@pentahoce:/opt/pentaho/pentaho-server$ sudo -u postgres psql -a -f /opt/pentaho/pentaho-server/data/postgresql/create_jcr_postgresql.sql

8) Change Pentaho settings for using PostgreSQL database for backend

context.xml

```
pentaho@pentahoce:/opt/pentaho/pentaho-server/tomcat/webapps/pentaho/META-INF$ sed -i
    s/"org.hsqldb.jdbcDriver"/"org.postgresql.Driver"/g context.xml
pentaho@pentahoce:/opt/pentaho/pentaho-server/tomcat/webapps/pentaho/META-INF$ sudo sed -i
    s/"jdbc:hsqldb:hsql:\/\/localhost\/hibernate"/"jdbc:postgresql:\/\/localhost:5432\/hibernate"/g
    context.xml
pentaho@pentahoce:/opt/pentaho/pentaho-server/tomcat/webapps/pentaho/META-INF$ sed -i s/"select
    count(\*) from     INFORMATION_SCHEMA.SYSTEM_SEQUENCES"/"select 1"/g context.xml
pentaho@pentahoce:/opt/pentaho/pentaho-server/tomcat/webapps/pentaho/META-INF$ sed -i
    s/"jdbc:hsqldb:hsql:\/\/localhost\/quartz"/"jdbc:postgresql:\/\/localhost:5432\/quartz"/g
    context.xml
```

applicationContext-spring-security-hibernate.properties

```
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system$ sed -i
    s/"org.hsqldb.jdbcDriver"/"org.postgresql.Driver"/g
    applicationContext-spring-security-hibernate.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system$ sed -i
    s/"jdbc:hsqldb:hsql:\/\/localhost\/hibernate"/"jdbc:postgresql:\/\/localhost:5432\/hibernate"/g
    applicationContext-spring-security-hibernate.properties
```

hibernate-settings.xml

```
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/hibernate$ sed -i
    s/"system\/hibernate\/hsql.hibernate.cfg.xml"/"system\/hibernate\/postgresql.hibernate.cfg.xml"/g
    hibernate-settings.xml
```

jdbc.properties

```
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"SampleData\/type=javax.sql.DataSource"/"#SampleData\/type=javax.sql.DataSource"/g
    jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"SampleData\/driver=org.hsqldb.jdbcDriver"/"#SampleData\/driver=org.hsqldb.jdbcDriver"/g
    jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"SampleData\/url=jdbc:hsqldb:hsql:\/\/localhost\/sampledata"/"#SampleData\/url=jdbc:hsqldb:hsql:\/\/localho
    jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"SampleData\/user=pentaho_user"/"#SampleData\/user=pentaho_user"/g jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"SampleData\/password=password"/"#SampleData\/password=password"/g jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"Hibernate\/driver=org.hsqldb.jdbcDriver"/"Hibernate\/driver=org.postgresql.Driver"/g
    jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"Hibernate\/url=jdbc:hsqldb:hsql:\/\/localhost\/hibernate"/"Hibernate\/url=jdbc:postgresql:\/\/localhost:54
    jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"Quartz\/driver=org.hsqldb.jdbcDriver"/"Quartz\/driver=org.postgresql.Driver"/g
    jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"Quartz\/url=jdbc:hsqldb:hsql:\/\/localhost\/quartz"/"Quartz\/url=jdbc:postgresql:\/\/localhost:5432\/quart
    jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"Shark\/type=javax.sql.DataSource"/"#Shark\/type=javax.sql.DataSource"/g jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"Shark\/driver=org.hsqldb.jdbcDriver"/"#Shark\/driver=org.hsqldb.jdbcDriver"/g jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"Shark\/url=jdbc:hsqldb:hsql:\/\/localhost\/shark"/"#Shark\/url=jdbc:hsqldb:hsql:\/\/localhost\/shark"/g
    jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"Shark\/user=sa"/"#Shark\/user=sa"/g jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"Shark\/password="/"#Shark\/password="/g jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"SampleDataAdmin\/type=javax.sql.DataSource"/"#SampleDataAdmin\/type=javax.sql.DataSource"/g
    jdbc.properties
```

```
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"SampleDataAdmin\/driver=org.hsqldb.jdbcDriver"/"#SampleDataAdmin\/driver=org.hsqldb.jdbcDriver"/g
    jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"SampleDataAdmin\/url=jdbc:hsqldb:hsql:\/\/localhost\/sampledata"/"#SampleDataAdmin\/url=jdbc:hsqldb:hsql:\
    jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"SampleDataAdmin\/user=pentaho_admin"/"#SampleDataAdmin\/user=pentaho_admin"/g jdbc.properties
pentaho@pentahoce:/opt/pentaho/pentaho-server/pentaho-solutions/system/simple-jndi$ sed -i
    s/"SampleDataAdmin\/password=password"/"#SampleDataAdmin\/password=password"/g jdbc.properties
```

9) Start Pentaho Server

   Make .sh files executable if needed

   sudo chmod +x /opt/pentaho/pentaho-server/*.sh

   Start server:

   cd /opt/pentaho/pentaho-server ./start-pentaho.sh

   Open Internet Browser and go to:

   http://127.0.0.1:8080/

Using SystemD to Start Pentaho

Intead of the traditional SysV init or Upstart procedure, Debian and Red Hat use *systemd* to manage system services. To see details of the system state, issue e.g. *systemctl status*.

The inner workings of systemd are explained in https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/part-Infrastructure_Services.html. System configuration is split into *units*, where the most important units are *services*, *targerts* (groups of units), *scopes* (externally created processes) and *slices* (a group of hierarchically organized units that manage system processes, like different user slices). Services are configured by writing *.service* files under */etc/systemd/system/* directory, which is reserved for unit files created or customized by the system administrator.

The service files consists of grouped key-value declaration directives and can be investigated with the *systemctl cat* directive. Note that the minus (-) sign after any '=' declatarion means "ignore errors".

```
pentaho@ctoolsbox:~$ systemctl cat pentaho.service

# /etc/systemd/system/pentaho.service
[Unit]
Description=Pentaho Server
After=network.target

[Service]
Type=forking
User=pentaho
Group=pentaho
ExecStart=/opt/pentaho/ctlscript.sh start
ExecStop=/opt/pentaho/ctlscript.sh stop
ExecReload=/opt/pentaho/ctlscript.sh restart
KillMode=process
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

Create this file and start the service with

```
sudo systemctl start pentaho.service
```

Finally, to start Pentaho server by default after every reboot, issue

```
sudo systemctl enable pentaho.service
```

Controlling Pentaho with SystemD

Examples:

```
sudo systemctl status pentaho.service
sudo systemctl start pentaho.service
sudo systemctl stop pentaho.service
```