Projektstatusbericht

# Projektzusammenfassung

|  |  |  |
| --- | --- | --- |
| Berichtsdatum | Projektname | Erstellt von |
| 10 Mai 2016 | WP-NoSQL & Big Data | Patrick Steinhauer, Besnik |

# Entwicklungsentwurf

# Projektübersicht

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vorgang | % erledigt | Fälligkeitsdatum | Unterstützer | Notizen |
| Aufgabe | 95 % | 15.04.2016 |  |  |
| Aufgabe | 0% | 15.04.2016 |  |  |
|  |  |  |  |  |

# Historie der Risiken und Probleme

|  |  |  |
| --- | --- | --- |
| Problem | Zugewiesen an | Datum |
|  |  |  |
|  |  |  |
|  |  |  |

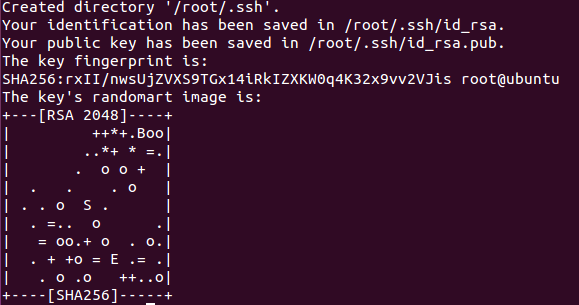
# Schlussfolgerungen/Empfehlungen

# installationsanleitung für hadoop

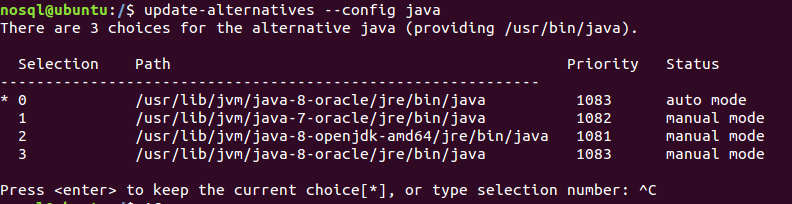
1. Als erstes muss JAVA auf dem Rechner installiert werden
   1. apt-get update
   2. apt-get install default-jdk
2. Nachdem JAVA installiert ist, kann man dies auf dessen Korrektheit folgendermaßen überprüfen:
   1. java -version
3. Als nächstes erstellt man SSH Zertifikate und installiert diese:
   1. Ssh-keygen -t rsa -P ‘‘
4. Wenn der Befehl eingegeben wurde kommt folgendes:

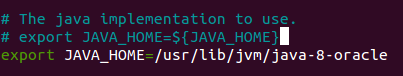
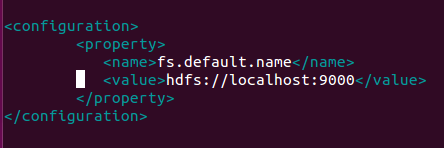
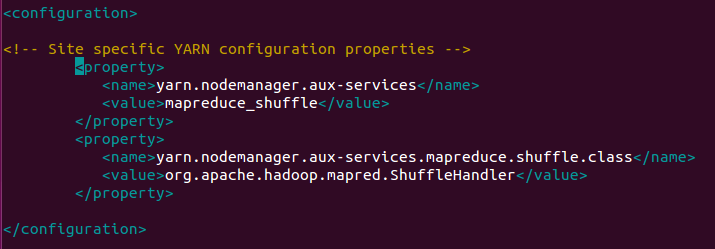
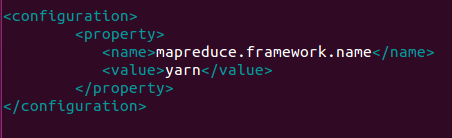
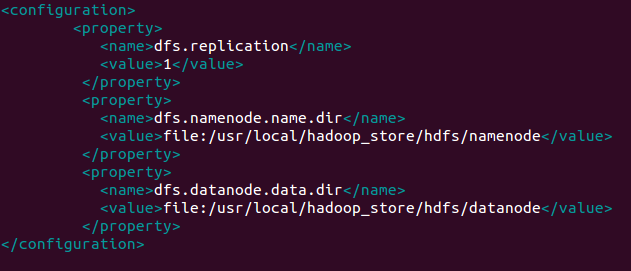


1. Wenn der Dialog erscheint einfach Enter drücken. Danach folgen diese Messages:



1. Nun benutzt man den folgenden Befehl, damit Hadoop SSH nutzen kann:
   1. sudo cat /root/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys
2. Als nächstes holen wir und Hadoop:
   1. sudo wget <http://apache.mirror.digionline.de/hadoop/common/stable/hadoop-2.7.2.tar.gz>
      1. **BEACHTE**: Link kann sich ändern. Falls dies der Fall sein sollte hier nachschauen und den Link anpassen:
      2. <http://www.apache.org/dyn/closer.cgi/hadoop/common/>
   2. sudo tar xfz hadoop-2.7.2.tar.gz
   3. sudo mv hadoop-2.7.2 /usr/local/hadoop
3. Jetzt müssen noch allerhand Konfigurationen gemacht werden:
   1. .bashrc
      1. Hierbei benötigen wir den Pfad zum JAVA jdk
         1. update-alternatives --config java



* + - * 1. Hier sieht man nun die Pfade der JDKs. Den richtigen Pfad für das aktuell genutzte wählen wir aus (in meinem Fall das oberste. Benötigt wird jedoch nur folgendes:
        2. /usr/lib/jvm/java-8-oracle
    1. Als nächstes öffnen wir die .bashrc mit einem Editor (nano z.B.):
       1. sudo nano ~/.bashrc
    2. Hier gehen wir nun an das Ende und fügen folgende Dinge ein:
       1. #HADOOP VARIABLES START
       2. export JAVA\_HOME=/usr/lib/jvm/java-8-oracle
       3. export HADOOP\_INSTALL=/usr/local/hadoop
       4. export PATH=$PATH:$HADOOP\_INSTALL/bin
       5. export PATH=$PATH:$HADOOP\_INSTALL/sbin
       6. export HADOOP\_MAPRED\_HOME=$HADOOP\_INSTALL
       7. export HADOOP\_COMMON\_HOME=$HADOOP\_INSTALL
       8. export HADOOP\_HDFS\_HOME=$HADOOP\_INSTALL
       9. export YARN\_HOME=$HADOOP\_INSTALL
       10. export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=$HADOOP\_INSTALL/lib/native
       11. export HADOOP\_OPTS="-Djava.library.path=$HADOOP\_INSTALL/lib"
       12. #HADOOP VARIABLES END
    3. Nachdem die .bashrc gespeichert wurde muss folgender Befehl ausgeführt werden, damit das System die neu erstellten Systemvariablen berücksichtigt:
       1. source ~/.bashrc
  1. Nun ändern wir die Datei des folgenden Pfades:
     1. sudo nano usr/local/hadoop/etc/hadoop/hadoop-env.sh
     2. Hier ändern wir nun das export JAVA\_HOME: 
  2. Als nächstes muss die folgende Datei ebenfalls angepasst werden:
     1. sudo nano usr/local/hadoop/etc/hadoop/core-site.xml
     2. Hier muss folgendes angepasst werden: 
  3. Anschließend ändern wir eine weitere xml Datei:
     1. sudo nano usr/local/hadoop/etc/hadoop/yarn-site.xml
     2. Hier muss folgendes angepasst werden: 
  4. Jetzt muss eine Datei umbenannt werden, was mittels des nächstes Befehls gemacht wird:
     1. sudo cp /usr/local/hadoop/etc/hadoop/mapred-site.xml.template /usr/local/hadoop/etc/hadoop/mapred-site.xml
     2. sudo nano /usr/local/hadoop/etc/hadoop/mapred-site.xml 
  5. Danach machen wir folgendes:
     1. sudo mkdir -p /usr/local/hadoop\_store/hdfs/namenode
     2. sudo mkdir -p /usr/local/hadoop\_store/hdfs/datanode
     3. sudo nano /usr/local/hadoop/etc/hadoop/hdfs-site.xml 
  6. Nun formatieren wir das Hadoop Filesystem:
     1. sudo su
     2. /usr/local/hadoop/bin/hdfs namenode -format 🡪 aus dem root Verzeichnis
     3. cd /usr/local/hadoop/sbin/
        1. ./start-dfs.sh
        2. ./start-yarn.sh
        3. jps

Diese Schritte müssen ggf. vor Schritt g. noch gemacht werden!

sudo apt-get update

sudo apt-get install openssh-server

sudo ufw allow 22

**Wenn bei dem folgenden Befehl eine Passphrase benötigt wird muss folgendes gemacht werden:**

ssh localhost 🡪 Passphrase? Dann mache folgendes:

ssh-keygen -t dsa -P '' -f ~/.ssh/id\_dsa

cat ~/.ssh/id\_dsa.pub >> ~/.ssh/authorized\_keys

<http://stackoverflow.com/questions/30820456/start-dfs-sh-command-not-found>

<http://stackoverflow.com/questions/13909568/starting-hadoop-process-using-start-all-sh-runs-into-issues>

<http://askubuntu.com/questions/51925/how-do-i-configure-a-new-ubuntu-installation-to-accept-ssh-connections>

<http://stackoverflow.com/questions/11315177/hadoop-java-io-ioexception-while-running-namenode-format>

<https://www.digitalocean.com/community/tutorials/how-to-install-hadoop-on-ubuntu-13-10>

Other helpful Links / Befehle

sudo chown -R myusername my-project-folder 🡪 username per: whoami 🡪 Löst Problem mit readonly status

<http://www.tutorialspoint.com/hbase/hbase_drop_table.htm>

<https://learnhbase.wordpress.com/2013/03/02/hbase-shell-commands/>

<http://stackoverflow.com/questions/5837698/converting-any-object-to-a-byte-array-in-java>

<http://projects.spring.io/spring-hadoop/#quick-start>

<http://www.cloudera.com/documentation/enterprise/5-4-x/topics/admin_hbase_scanning.html>

<http://www.cloudera.com/documentation/enterprise/5-4-x/topics/admin_writing_data_to_hbase.html>

<https://www.cloudera.com/documentation/enterprise/5-6-x/topics/admin_hbase_import.html>