

# Installation de Apache STORM

## Version 1.2

**Rédacteurs V1 :** Philippe Lacomme ([placomme@isima.fr](mailto:placomme@isima.fr)), Raksmei Phan ([phan@isima.fr](mailto:phan@isima.fr))

**Date :** 12 août 2015

**Rédacteurs V2 :** Philippe Lacomme ([placomme@isima.fr](mailto:placomme@isima.fr)), Raksmei Phan ([phan@isima.fr](mailto:phan@isima.fr)),  
étudiants en projets ISIMA (Soriano Baptiste et Zougari Yannis)

**Date :** 25 mai 2016

**Installation réalisée sur :** Ubuntu 15.04

**Environnement :** Vmware

### Sources :

<https://www.youtube.com/watch?v=3SHGsKfm1RM&spfreload=10>

<http://10jumps.com/blog/storm-installation-single-machine>

### Licence :

Ce document est une compilation d'information parfois en Anglais ou en Français librement accessibles sur Internet.

Permission vous est donnée de copier, distribuer et/ou modifier ce document selon les termes de la Licence GNU Free Documentation License, version 1.3 ou ultérieure publiée par la Free Software Foundation ; sans section inaltérable, sans texte de première page de couverture et sans texte de dernière page de couverture. Une copie de cette licence en anglais est consultable sur le site suivant : <http://www.gnu.org/licenses/fdl.html>

### Étape 1. Ajouter les librairies nécessaire au système

Ceci peut se faire en ligne de commande via un terminal. Les commandes à utiliser sont les suivantes :

```
sudo apt-get update
sudo apt-get install openjdk-7-jdk
sudo apt-get install git -y
sudo apt-get install libtool -y
sudo apt-get install automake -y
sudo apt-get install uuid-dev -y
sudo apt-get install g++ -y
sudo apt-get install gcc-multilib -y
```

```
sudo apt-get install libtool-bin -y
sudo apt-get install tree
```

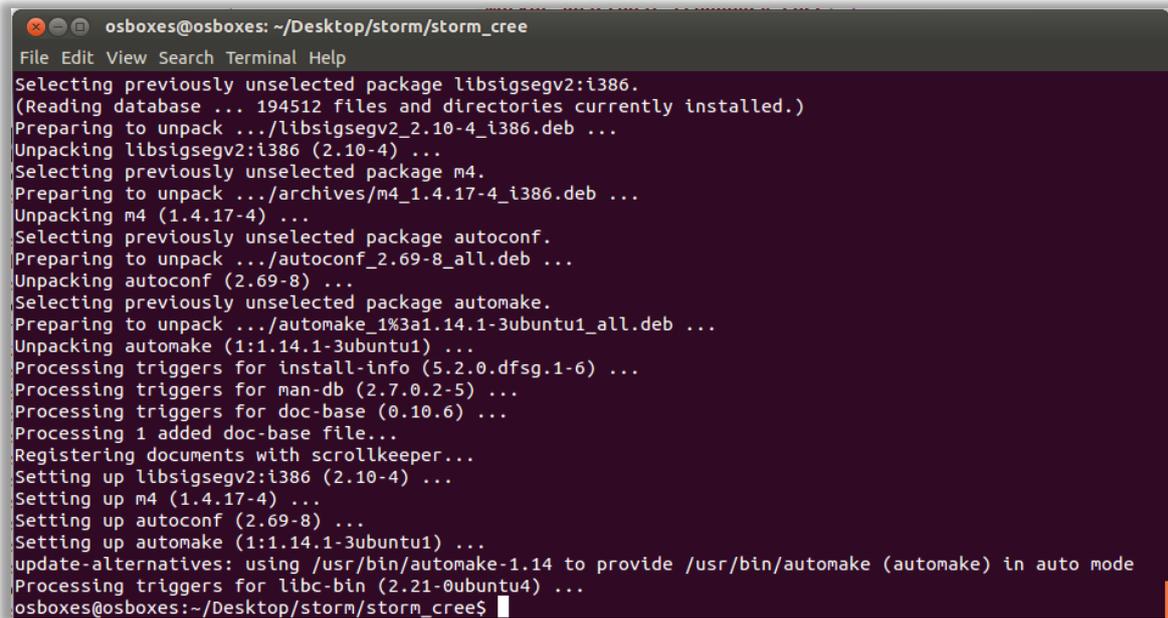


Figure 1. Mise à jour via le terminal

Si le système possède le gestionnaire de paquets Synaptic est installé, on peut directement utiliser l'interface graphique pour la mise à jour.

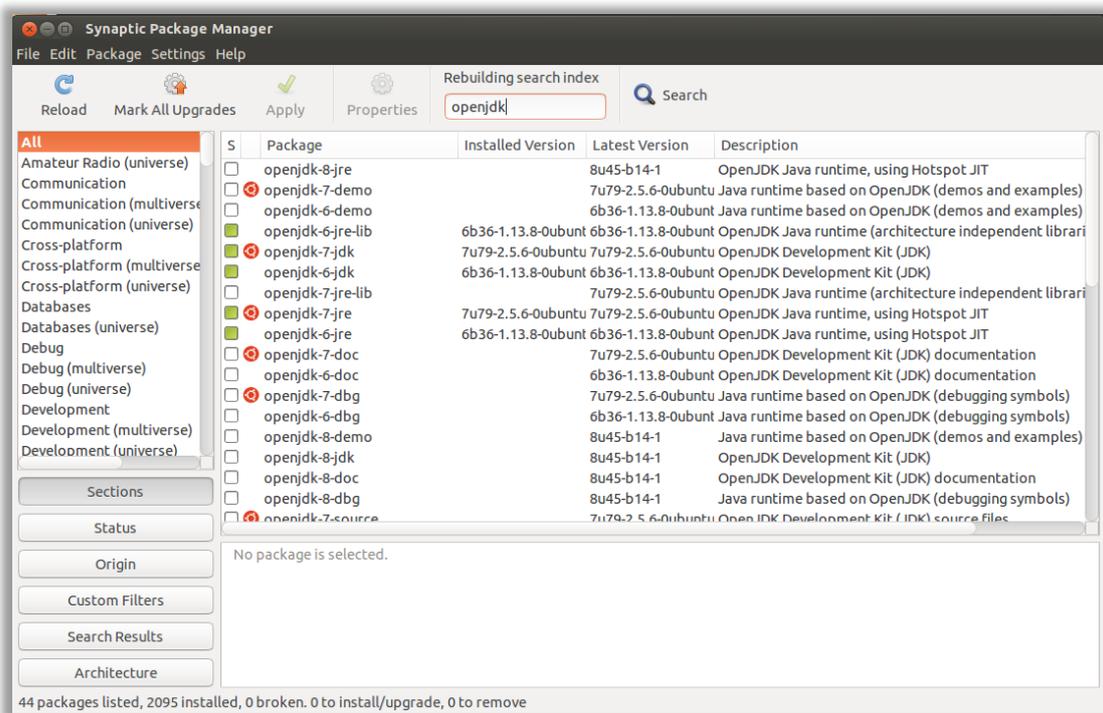


Figure 2. Mise à jour via Synaptic

Il faut prévoir de 5 à 15 min en fonction de la machine et de la qualité de la liaison Internet.

Par la suite, toutes les opérations d'installation se feront dans un répertoire nommé **installation** directement sur le **Bureau**.

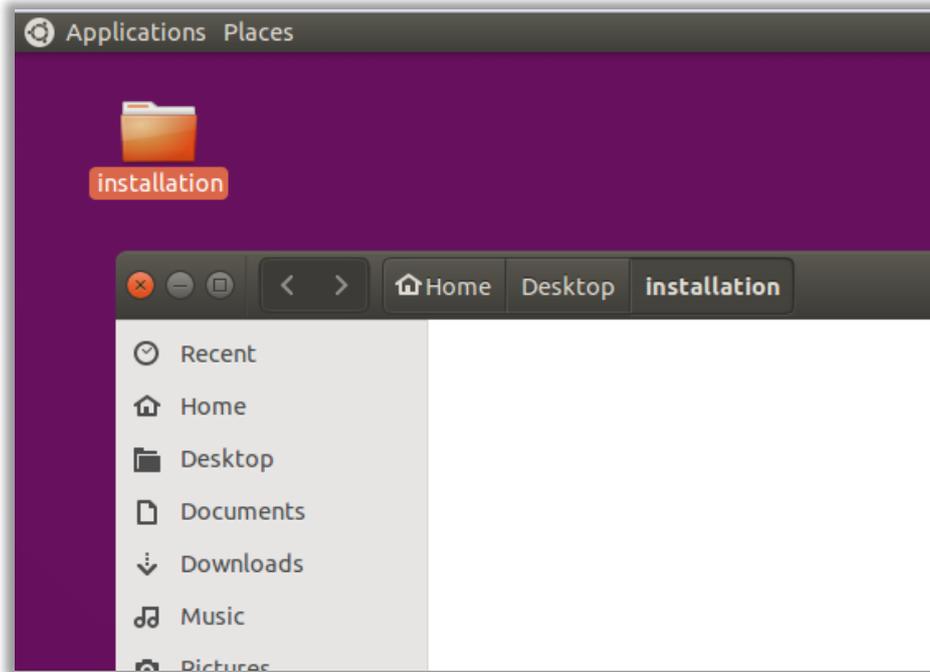


Figure 3. Répertoire de travail sur le Bureau d'Ubuntu (version 15)

## **Etape 2. Installation de Zookeeper**

Deux solutions existent pour l'installation de Zookeeper. Il est possible d'utiliser le gestionnaire Synaptic intégré à Ubuntu ou bien de le télécharger directement du site d'Apache.

Pour la première solution, il suffit de chercher le mot "Zookeeper" dans le gestionnaire Synaptic comme le montre la Figure 4.

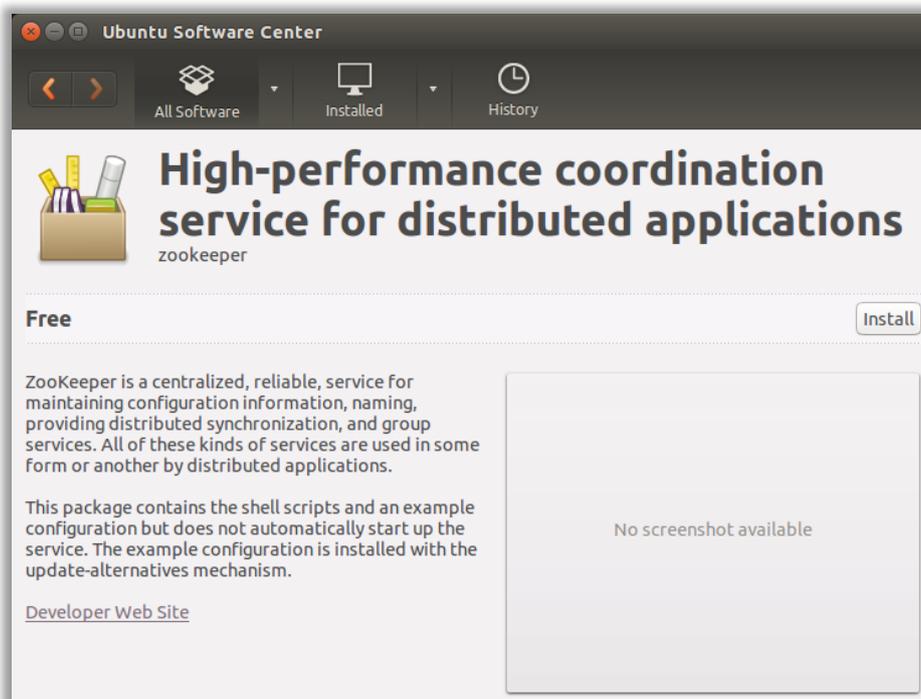


Figure 4. Installation Zookeeper à partir du gestionnaire Synaptic

Une fois l'installation de Zookeeper avec Synaptic réalisée, il suffit de le lancer à partir de son répertoire d'installation : /usr/share/zookeeper/bin, avec la commande `zkServer.sh start` (Figure 5).

```
raksme@raksme-VirtualBox: /usr/share/zookeeper/bin
raksme@raksme-VirtualBox: /usr/share/zookeeper/bin$ sudo
usage: sudo -h | -K | -k | -V
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user]
[command]
usage: sudo [-AbEHknPS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p
prompt] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p
prompt] [-u user] file ...
raksme@raksme-VirtualBox: /usr/share/zookeeper/bin$ ./zkServer.sh start
JMX enabled by default
Using config: /etc/zookeeper/conf/zoo.cfg
Starting zookeeper ... ./zkServer.sh: line 104: /var/lib/zookeeper/zookeeper_ser
ver.pid: Permission denied
FAILED TO WRITE PID
raksme@raksme-VirtualBox: /usr/share/zookeeper/bin$ ./zkServer.sh: line 100: /v
ar/log/zookeeper/zookeeper.out: Permission denied
^C
raksme@raksme-VirtualBox: /usr/share/zookeeper/bin$ sudo ./zkServer.sh start
[sudo] password for raksme:
JMX enabled by default
Using config: /etc/zookeeper/conf/zoo.cfg
Starting zookeeper ... STARTED
raksme@raksme-VirtualBox: /usr/share/zookeeper/bin$
```

Figure 5 Démarrage de zookeeper (avec l'installation automatique de Synaptic)

Il faut se rendre sur le site suivant qui est le site de référence (Figure 6) :

<http://zookeeper.apache.org/releases.html>



Figure 6 Site de Zookeeper

En utilisant la section Download, il faut récupérer la dernière version qui se trouve dans un répertoire **Stable** quelque soit le serveur choisi (voir Figure 7)



Figure 7 Choix d'une version Stable

Au moment de l'écriture de ce tuto, la version disponible est la version 3.4.6 (Figure 8).

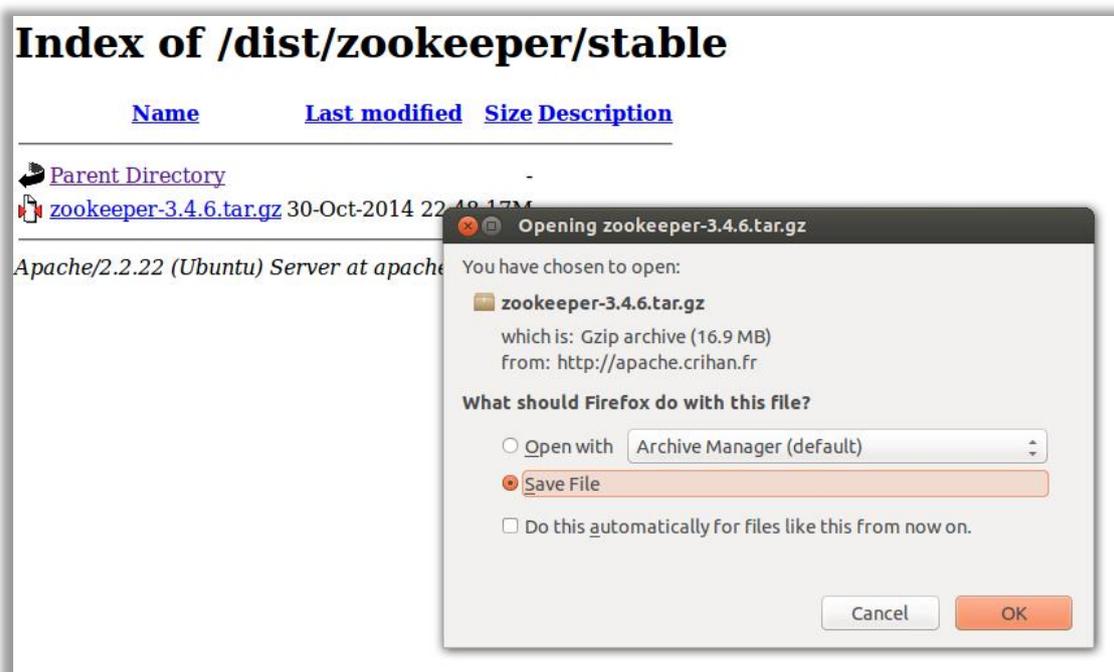
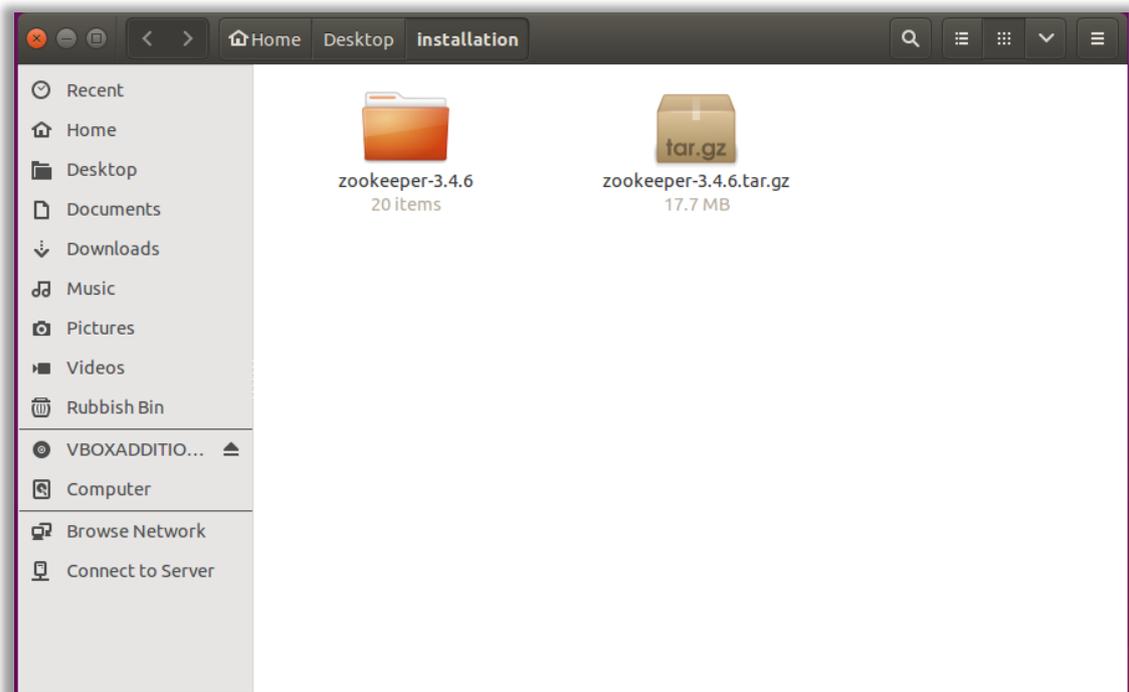


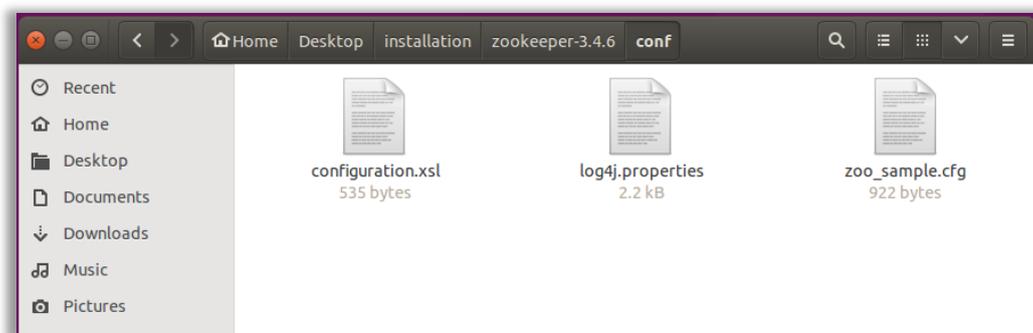
Figure 8 Téléchargement de Zookeeper

Une fois l'archive obtenue (fichier .tar.gz), il faut ouvrir l'archive et récupérer le contenu qui se présente sous la forme d'un sous-répertoire (Figure 9).



*Figure 9 Récupération de l'archive Zookeeper*

Il faut se rendre dans le répertoire `/conf` et éditer le fichier **zoo\_sample.cfg** (Figure 10).



*Figure 10. Contenu du dossier "conf"*

Le plus simple consiste à sauvegarder ce fichier en changeant son nom en **zoo.cfg** (Figure 11).

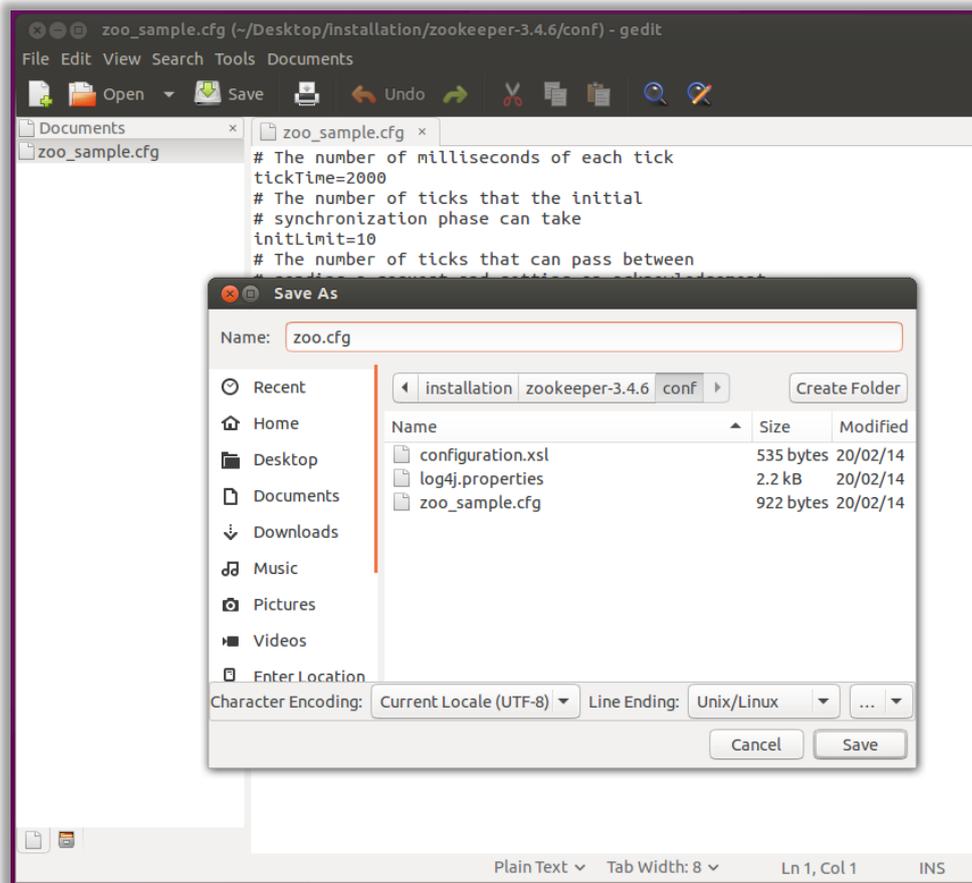


Figure 11. Copie du fichier

Après sauvegarde, le fichier apparaît dans le répertoire conf (figure 12).



Figure 12. Contenu répertoire conf

Le contenu du fichier **zoo.cfg** doit être mis à jour en modifiant **dataDir**. Le plus simple consiste à créer un répertoire nommé **data** dans le répertoire **zookeeper** (Figure 13)

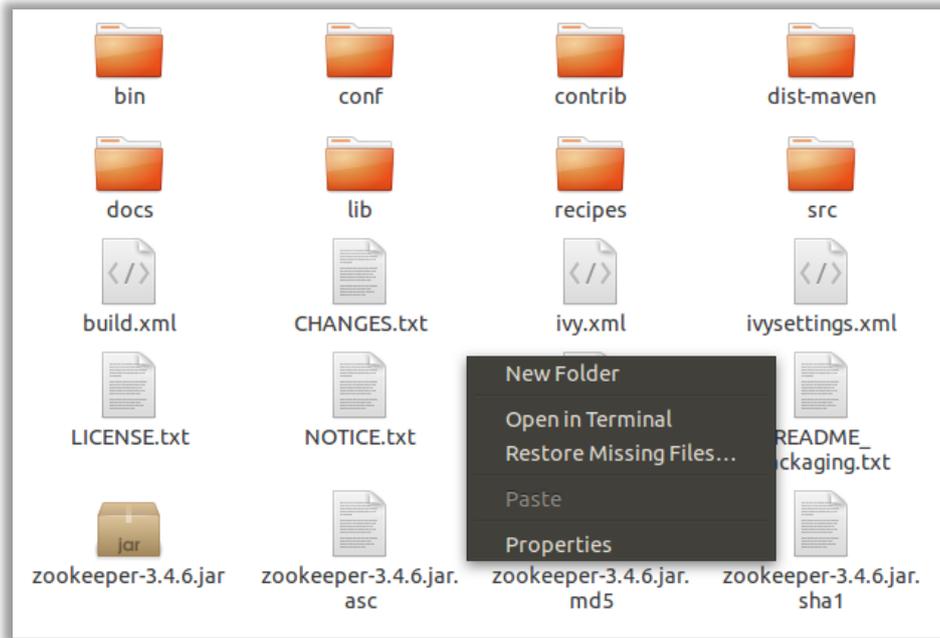


Figure 13. Création d'un nouveau répertoire

On peut ensuite, via le terminal, vérifier le chemin d'accès en faisant :

**cd /home/<users>/Desktop/installation/zookeeperxxxx/data** et **pwd** (Figure 14)

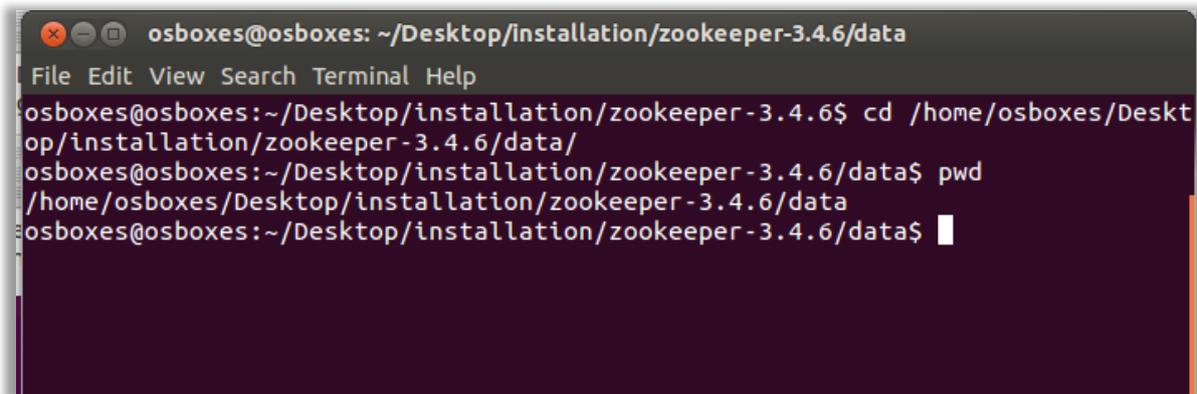


Figure 14. Vérification du chemin d'accès

On peut ensuite modifier **dataDir** comme indiqué sur la figure 15.

```

*zoo.cfg (~/Desktop/Installation/zookeeper-3.4.6/conf) - gedit
File Edit View Search Tools Documents
Open Save Undo
Documents *zoo.cfg x
zoo.cfg # The number of milliseconds of each tick
tickTime=2000
# The number of ticks that the initial
# synchronization phase can take
initLimit=10
# The number of ticks that can pass between
# sending a request and getting an acknowledgement
syncLimit=5
# the directory where the snapshot is stored.
# do not use /tmp for storage, /tmp here is just
# example sakes.
dataDir=/home/osboxes/Desktop/Installation/zookeeper-3.4.6/data
# the port at which the clients will connect
clientPort=2181
# the maximum number of client connections.
# increase this if you need to handle more clients
#maxClientCnxns=60
#
# Be sure to read the maintenance section of the
# administrator guide before turning on autopurge.
#
# http://zookeeper.apache.org/doc/current/
zookeeperAdmin.html#sc_maintenance
#
# The number of snapshots to retain in dataDir
#autopurge.snapRetainCount=3
# Purge task interval in hours
# Set to "0" to disable auto purge feature
#autopurge.purgeInterval=1
Plain Text Tab Width: 8 Ln 28, Col 27 INS

```

Figure 15. Le fichier après modification

Il faut ensuite vérifier les chemins d'accès à la version de Java. Dans un terminal, taper :

**update-alternatives --config java**

ce qui doit donner un résultat similaire à celui de la Figure 16.

```

osboxes@osboxes: ~
File Edit View Search Terminal Help
osboxes@osboxes:~$ update-alternatives --config java
There are 2 choices for the alternative java (providing /usr/bin/java).

  Selection    Path                                          Priority  Status
-----
*  0            /usr/lib/jvm/java-7-openjdk-1.386/jre/bin/java  1071    auto mode
   1            /usr/lib/jvm/java-6-openjdk-1.386/jre/bin/java  1061    manual mode
   2            /usr/lib/jvm/java-7-openjdk-1.386/jre/bin/java  1071    manual mode

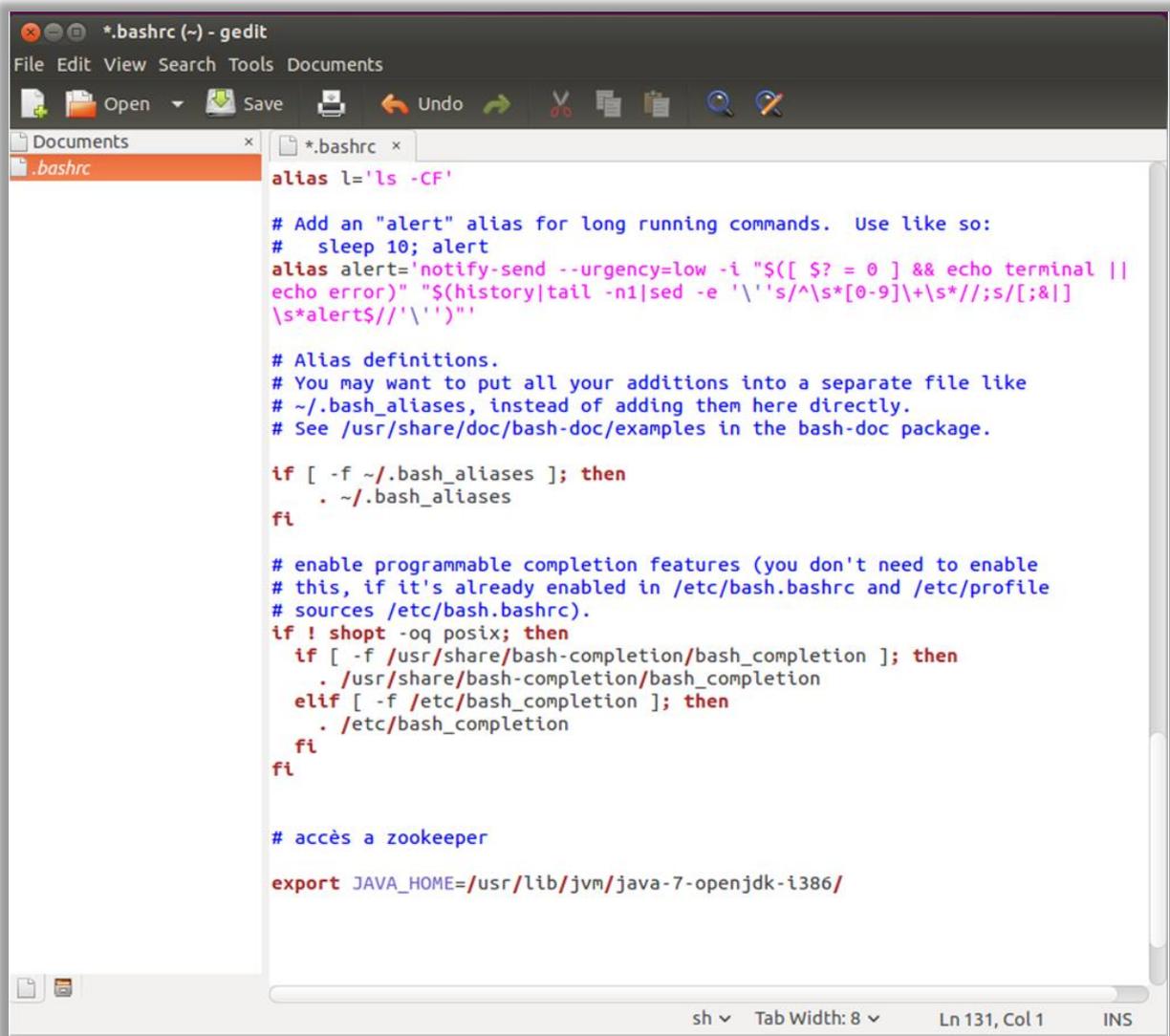
Press enter to keep the current choice[*], or type selection number:

```

Figure 16. Les versions de java disponibles

Dans un terminal, tapez la commande : **sudo gedit ~/.bashrc**

A la fin du fichier, il faut ajouter **export JAVA\_HOME** avec un chemin d'accès au JRE. Sur la machine de test, cela donne le résultat de la Figure 17.



```
alias l='ls -CF'

# Add an "alert" alias for long running commands. Use like so:
# sleep 10; alert
alias alert='notify-send --urgency=low -i "${ $? = 0 }" && echo terminal ||
echo error" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;&]
\s*alert$//'\''")'

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi

# accès a zookeeper

export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-i386/
```

Figure 17. Ajout d'une variable JAVA\_HOME

Il faut ensuite recommencer avec **ZOOKEEPER\_HOME** et modifier ensuite la variable **PATH**.

La fin du fichier doit être similaire à celle-ci :

```
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-i386/
export ZOOKEEPER_HOME=/home/osboxes/Desktop/installation/zookeeper-3.4.6
export PATH=$PATH:$JAVA_HOME/bin:$ZOOKEEPER_HOME/bin
```

Dans le terminal tapez : **source ~/.bashrc**

On peut ensuite, se rendre dans le répertoire bin de zookeeper et démarrer le serveur comme le montre la Figure 18.

```
osboxes@osboxes: ~/Desktop/installation/zookeeper-3.4.6/bin
File Edit View Search Terminal Help
osboxes@osboxes:~$ cd $ZOOKEEPER_HOME/bin
osboxes@osboxes:~/Desktop/installation/zookeeper-3.4.6/bin$ ls
README.txt  zkCli.cmd  zkEnv.cmd  zkServer.cmd
zkCleanup.sh zkCli.sh  zkEnv.sh  zkServer.sh
osboxes@osboxes:~/Desktop/installation/zookeeper-3.4.6/bin$ ./zkServer.sh start
JMX enabled by default
Using config: /home/osboxes/Desktop/installation/zookeeper-3.4.6/bin/./conf/zoo.cfg
Starting zookeeper ... STARTED
osboxes@osboxes:~/Desktop/installation/zookeeper-3.4.6/bin$
```

Figure 18. Démarrage de zookeeper

On peut ensuite taper la commande JPS et vérifier si les tâches fonctionnent. On doit obtenir le résultat de la Figure 19.

```
osboxes@osboxes: ~/Desktop/installation/zookeeper-3.4.6/bin
File Edit View Search Terminal Help
osboxes@osboxes:~$ cd $ZOOKEEPER_HOME/bin
osboxes@osboxes:~/Desktop/installation/zookeeper-3.4.6/bin$ ls
README.txt  zkCli.cmd  zkEnv.cmd  zkServer.cmd
zkCleanup.sh zkCli.sh  zkEnv.sh  zkServer.sh
osboxes@osboxes:~/Desktop/installation/zookeeper-3.4.6/bin$ ./zkServer.sh start
JMX enabled by default
Using config: /home/osboxes/Desktop/installation/zookeeper-3.4.6/bin/./conf/zoo.cfg
Starting zookeeper ... STARTED
osboxes@osboxes:~/Desktop/installation/zookeeper-3.4.6/bin$ jps
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar
3437 Jps
3420 QuorumPeerMain
osboxes@osboxes:~/Desktop/installation/zookeeper-3.4.6/bin$
```

Figure 19. Vérification

### **Etape 3. Installation de ZeroMQ**

Il faut se rendre sur le site suivant qui est le site de référence (Figure 20) :

<http://download.zeromq.org/>



Figure 20. Le site principal de ZeroMQ

Choisir une version récente. Au moment où ce tuto est écrit, il s'agit de la version 4.1.2. Il faut télécharger l'archive (Figure 21). Par la suite, nous utilisons la version 2.1.17.

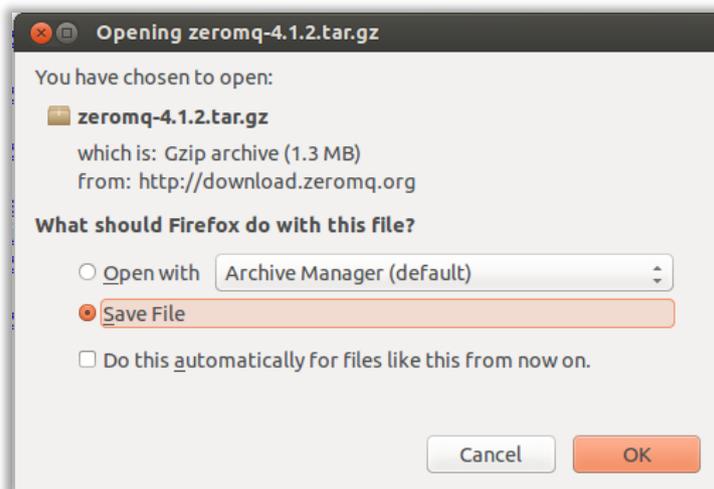


Figure 21. Choix d'une version de ZeroMQ et téléchargement

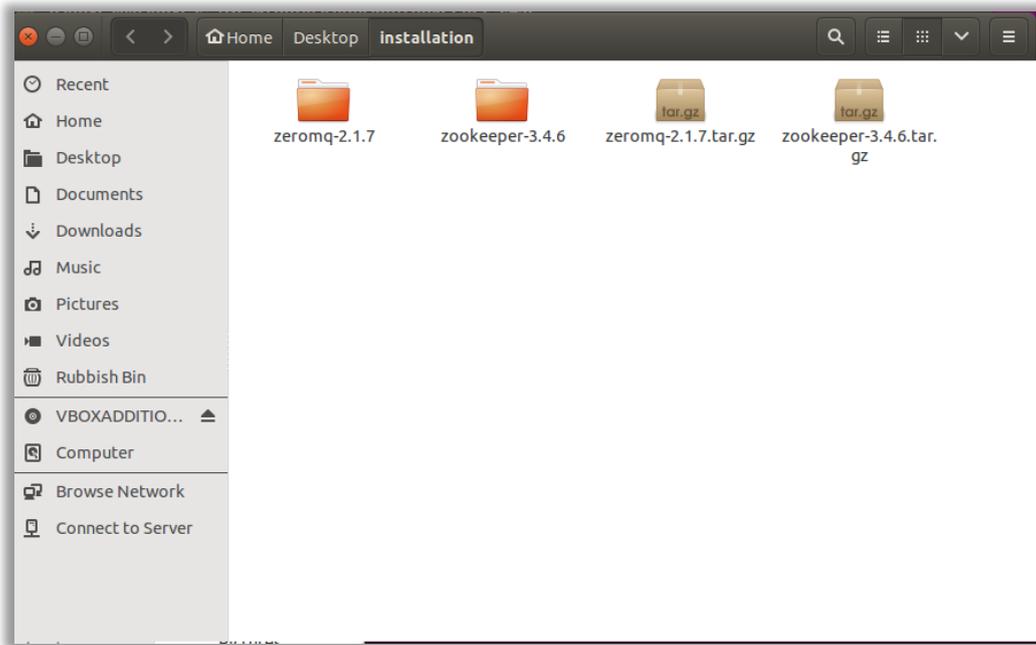


Figure 22. Contenu du répertoire de téléchargement

Il faut ouvrir un terminal et se rendre dans le répertoire **zerom** puis taper: **./configure** comme le montre la Figure 23.

```
osboxes@osboxes: ~/Desktop/installation/zeromq-2.1.7
File Edit View Search Terminal Help
osboxes@osboxes:~/Desktop/installation$ cd zeromq-2.1.7/
osboxes@osboxes:~/Desktop/installation/zeromq-2.1.7$ ./configure
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /bin/mkdir -p
checking for gawk... no
checking for mawk... mawk
checking whether make sets $(MAKE)... yes
checking how to create a ustar tar archive... gnutar
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking for style of include used by make... GNU
```

Figure 23. Exécution de la commande **./configure**

Il faut ensuite poursuivre en tapant : **make**.

Patientez 2 à 3 min en fonction de la machine utilisée (Figure 24).

```
osboxes@osboxes: ~/Desktop/installation/zeromq-2.1.7
File Edit View Search Terminal Help
osboxes@osboxes:~/Desktop/installation/zeromq-2.1.7$ make
Making all in src
make[1]: Entering directory '/home/osboxes/Desktop/installation/zeromq-2.1.7/src'
make all-am
make[2]: Entering directory '/home/osboxes/Desktop/installation/zeromq-2.1.7/src'
CXX libzmq_la-clock.lo
CXX libzmq_la-command.lo
CXX libzmq_la-ctx.lo
CXX libzmq_la-connect_session.lo
CXX libzmq_la-decoder.lo
CXX libzmq_la-device.lo
CXX libzmq_la-devpoll.lo
CXX libzmq_la-dist.lo
CXX libzmq_la-encoder.lo
CXX libzmq_la-epoll.lo
CXX libzmq_la-err.lo
CXX libzmq_la-fq.lo
```

Figure 24. Exécution de la commande `make`

Il faut ensuite poursuivre en tapant : **make install**.

La dernière étape consiste à cloner le dépôt **jzmq** dans le répertoire (Figure 25).

`git clone https://github.com/nathanmarz/jzmq.git`

```
osboxes@osboxes:~/Desktop/installation/zeromq-2.1.7$ git clone https://github.com/nathanmarz/jzmq.git
Cloning into 'jzmq'...
remote: Counting objects: 611, done.
remote: Total 611 (delta 0), reused 0 (delta 0), pack-reused 611
Receiving objects: 100% (611/611), 348.65 KiB | 0 bytes/s, done.
Resolving deltas: 100% (239/239), done.
Checking connectivity... done.
```

Figure 25. Copie du dépôt `jzmq`

Il faut taper ensuite les commandes qui suivent :

```
cd jzmq
```

```
sed -i 's/classdist_noinst.stamp/classnoinst.stamp/g' src/Makefile.am
```

Si `libtool` n'est pas installé, il faut aussi réaliser l'installation.

```
sudo apt-get install libtool -y
```

```
sudo apt-get install libtool-bin -y
```

Sur la machine de test où `libtool` est déjà présent, on obtient le terminal dans l'état de la Figure 26.

```

osboxes@osboxes:~/Desktop/installation/zeromq-2.1.7$ cd jzmq/
osboxes@osboxes:~/Desktop/installation/zeromq-2.1.7/jzmq$ sed -i 's/classdist_noinst.stamp/cl
' src/Makefile.am
osboxes@osboxes:~/Desktop/installation/zeromq-2.1.7/jzmq$ sudo apt-get install libtool -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
libtool is already the newest version.
0 to upgrade, 0 to newly install, 0 to remove and 221 not to upgrade.
osboxes@osboxes:~/Desktop/installation/zeromq-2.1.7/jzmq$ █

```

*Figure 26. Installation en cours*

Il faut ensuite lancer **autogen.sh** et patienter quelques secondes (Figure 27).

```

osboxes@osboxes:~/Desktop/installation/zeromq-2.1.7/jzmq$ ./autogen.sh
autoreconf: Entering directory `.'
autoreconf: configure.in: not using Gettext
autoreconf: running: aclocal -I config --force -I config
aclocal: warning: autoconf input should be named 'configure.ac', not 'configure.in'
autoreconf: configure.in: tracing
autoreconf: running: libtoolize --copy --force
libtoolize: putting auxiliary files in AC_CONFIG_AUX_DIR, `config'.
libtoolize: copying file `config/ltmain.sh'
libtoolize: putting macros in AC_CONFIG_MACRO_DIR, `config'.
libtoolize: copying file `config/libtool.m4'
libtoolize: copying file `config/ltoptions.m4'
libtoolize: copying file `config/ltsugar.m4'
libtoolize: copying file `config/ltversion.m4'
libtoolize: copying file `config/lt-obsolete.m4'
aclocal: warning: autoconf input should be named 'configure.ac', not 'configure.in'

```

*Figure 27. Exécution de autogen.sh*

**Tapez ensuite :**

- ./configure
- make
- make install

**Etape 4. Installation de STORM**

La page de téléchargement est la suivante :

<https://storm.apache.org/downloads.html>

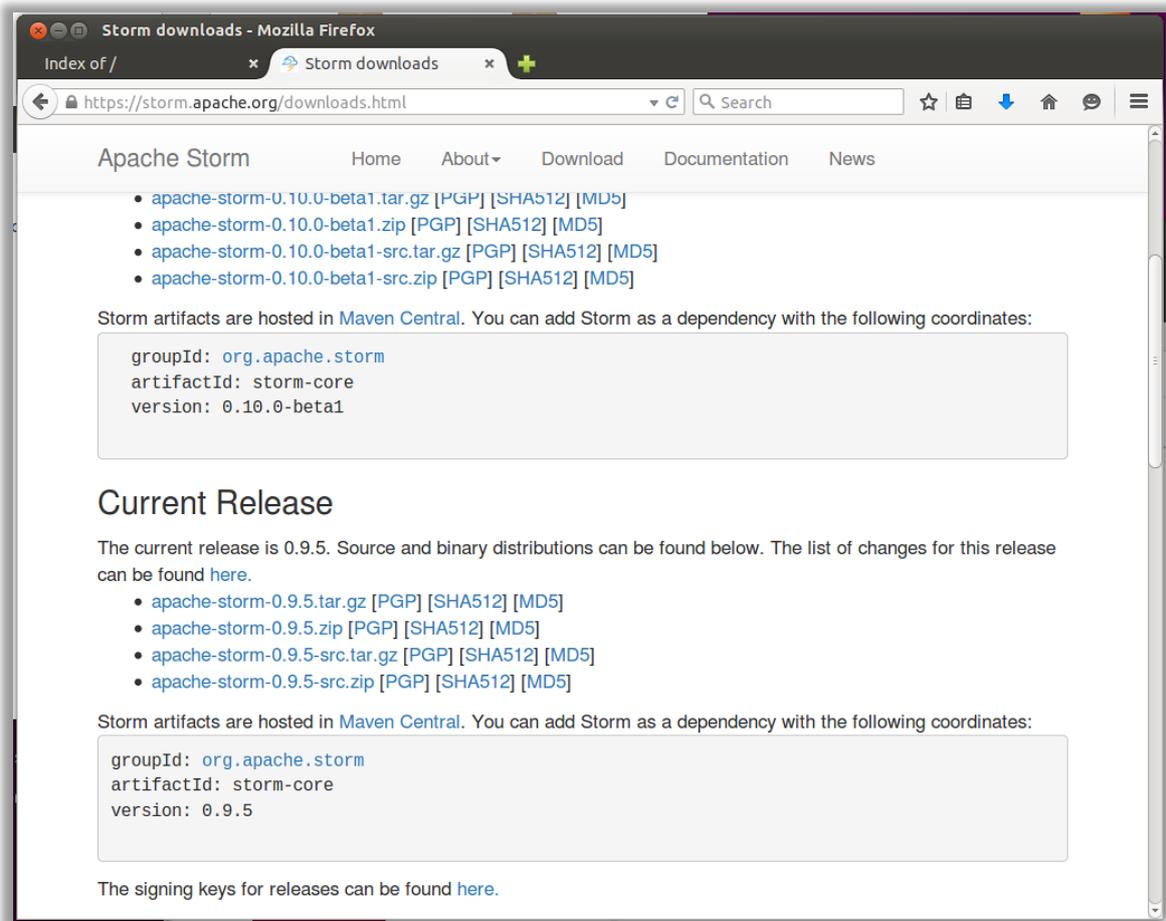


Figure 28. Le site Apache Storm

On retiendra de préférence la dernière version stable, ici la version 0.9.5. (Figure 29).

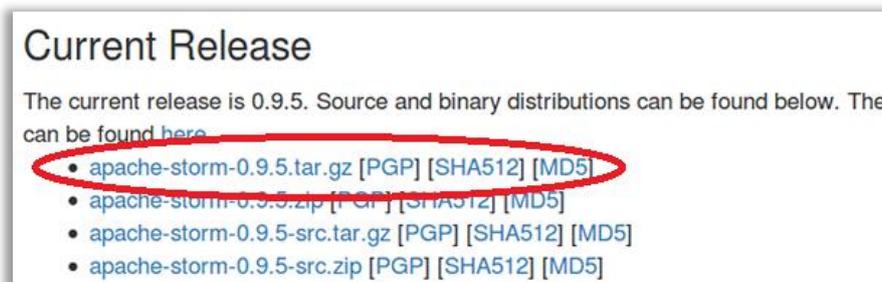


Figure 29. Choix de la version

Pour des raisons pratiques, on peut récupérer le contenu de l'archive et le déposer dans le répertoire **installation** qui ressemble alors à celui de la Figure 30.

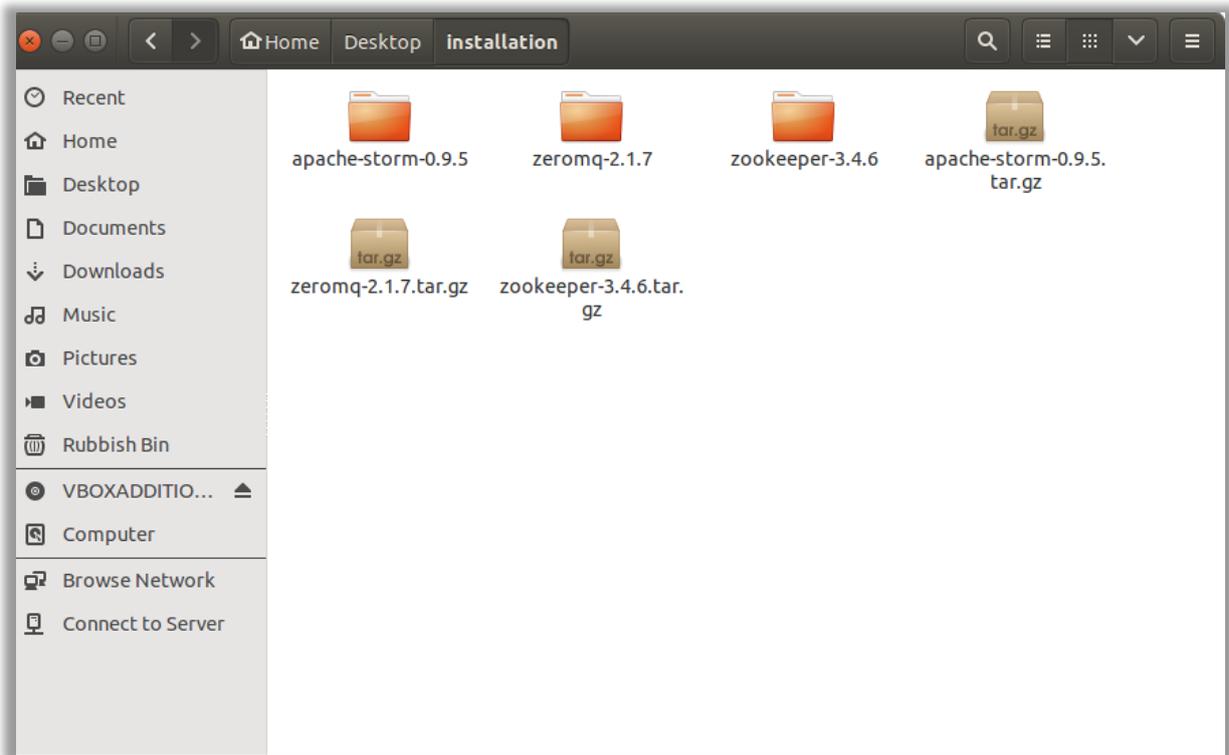


Figure 30. Nouveau répertoire

Il est fortement recommandé de modifier le nom du répertoire pour un nom plus simple (Figure 31).

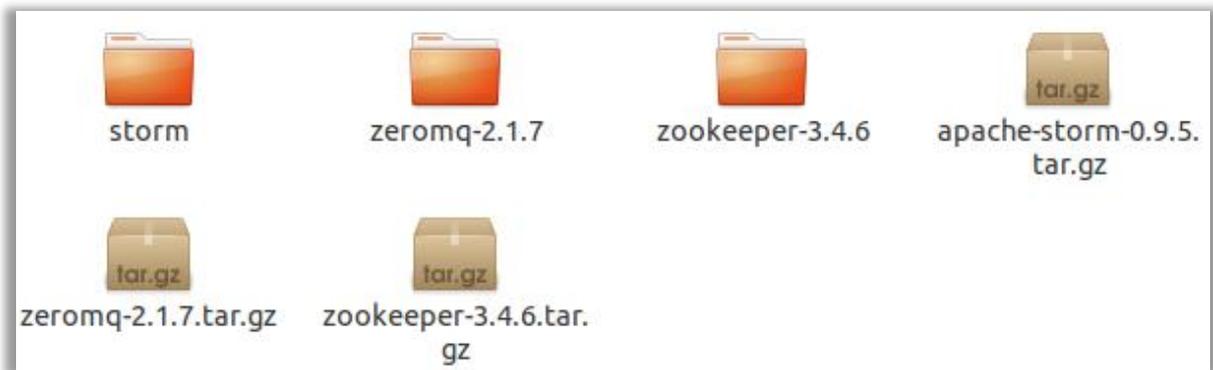


Figure 31. Nouveau répertoire Storm

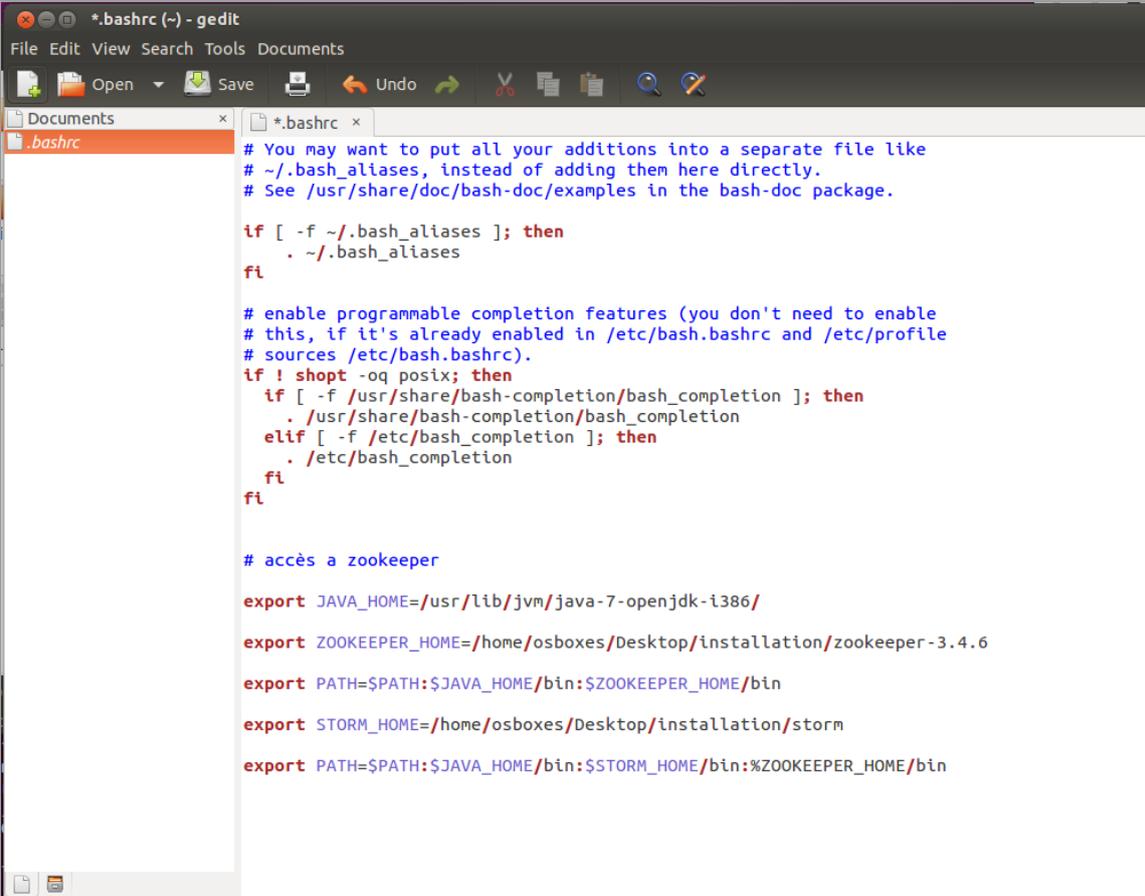
De manière usuelle, on place ce répertoire dans `/usr/local/bin/`.

On va ensuite ajouter quelques variables d'environnements.

Faire : **sudo gedit ~/.bashr**

```
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-i386/
export ZOOKEEPER_HOME=/home/osboxes/Desktop/installation/zookeeper-3.4.6
export PATH=$PATH:$JAVA_HOME/bin:$ZOOKEEPER_HOME/bin
export STORM_HOME=/home/osboxes/Desktop/installation/storm
export PATH=$PATH:$JAVA_HOME/bin:$STORM_HOME/bin:%ZOOKEEPER_HOME/bin
```

La nouvelle version du fichier est celle présentée sur la Figure 32.



```
*.bashrc (~) - gedit
File Edit View Search Tools Documents
Open Save Undo
Documents
*.bashrc x
.bashrc
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi

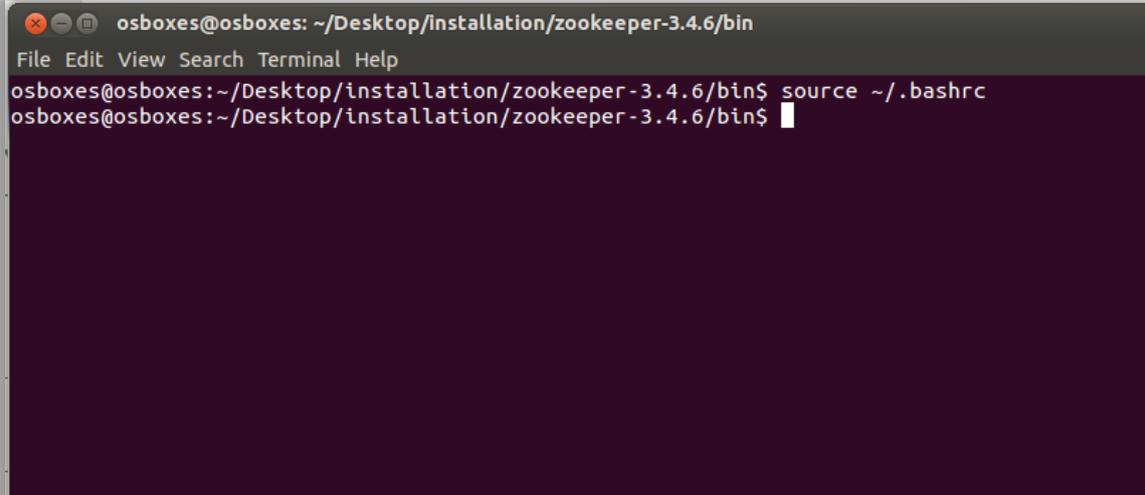
# accès a zookeeper

export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-i386/
export ZOOKEEPER_HOME=/home/osboxes/Desktop/installation/zookeeper-3.4.6
export PATH=$PATH:$JAVA_HOME/bin:$ZOOKEEPER_HOME/bin
export STORM_HOME=/home/osboxes/Desktop/installation/storm
export PATH=$PATH:$JAVA_HOME/bin:$STORM_HOME/bin:%ZOOKEEPER_HOME/bin

sh Tab Width: 8 Ln 113, Col 49 INS
```

Figure 32. Fichier bashrc modifié

Il faut penser à recharger le fichier bashrc par la commande (Figure 33) : **source ~/.bashrc**



```
osboxes@osboxes: ~/Desktop/installation/zookeeper-3.4.6/bin
File Edit View Search Terminal Help
osboxes@osboxes:~/Desktop/installation/zookeeper-3.4.6/bin$ source ~/.bashrc
osboxes@osboxes:~/Desktop/installation/zookeeper-3.4.6/bin$
```

Figure 33. Fichier bashrc rechargé

Via l'interface graphique, il faut se rendre dans le répertoire `/storm/conf` et éditer le fichier `storm.yaml` (Figure 34).

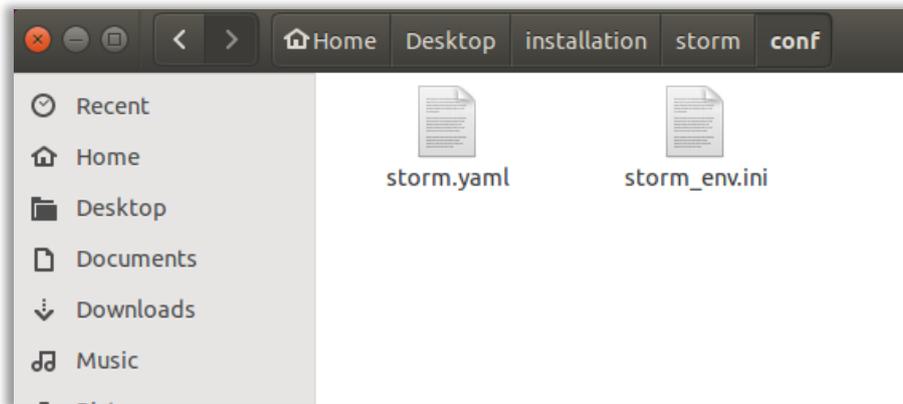


Figure 34. Le fichier de configuration de Storm

Avant de poursuivre, ceux ne connaissant pas la configuration réseau de leur machine, doivent consulter les informations de bases, dans un terminal via la commande `ifconfig`. Ils doivent obtenir un résultat proche de celui de la Figure 35.

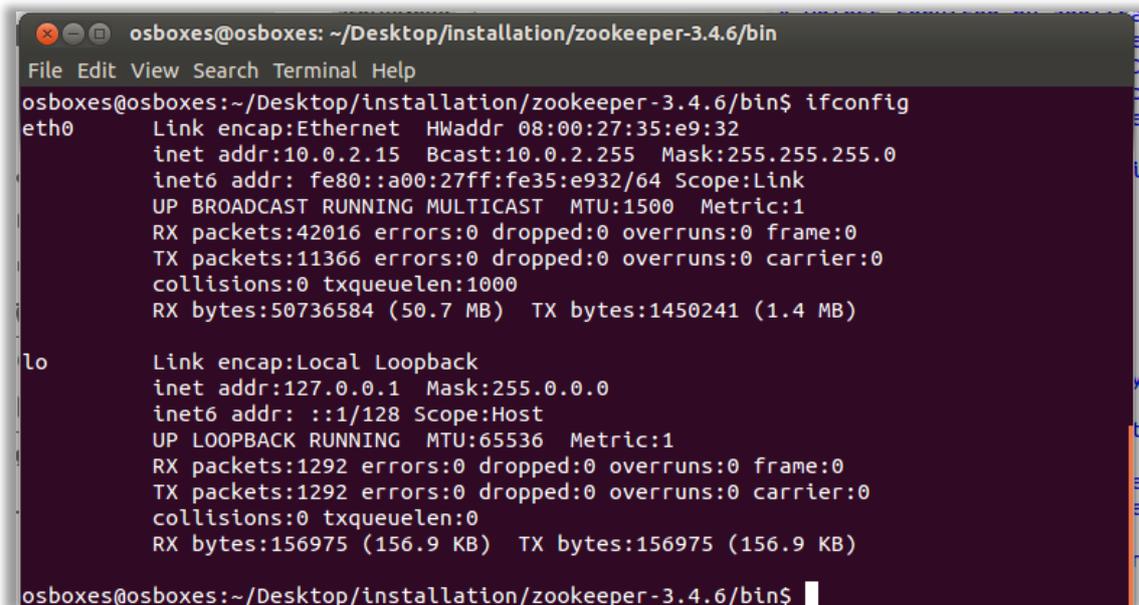


Figure 35. Le fichier de configuration réseau

Dans notre cas de figure les lignes à ajouter sont les suivantes :

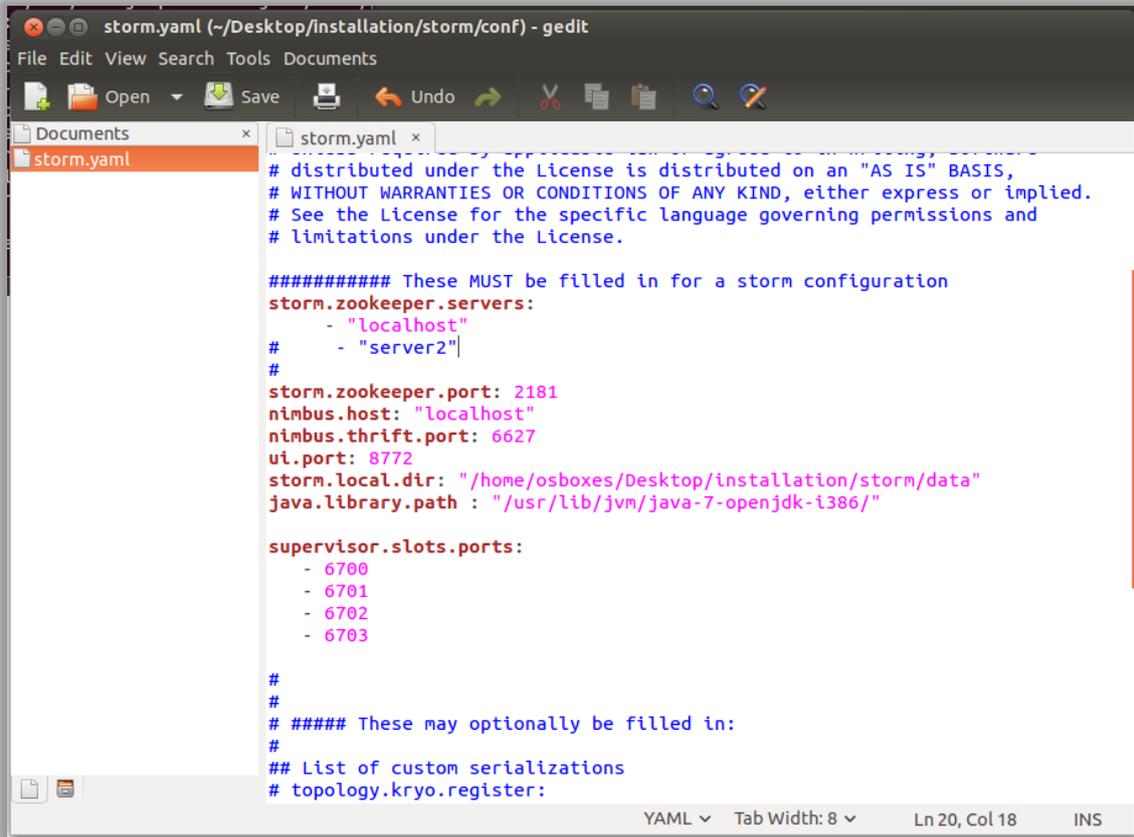
```
##### These MUST be filled in for a storm configuration
storm.zookeeper.servers:
    - "localhost"
#    - "server2"
#
storm.zookeeper.port: 2181
nimbus.host: "localhost"
nimbus.thrift.port: 6627
ui.port: 8772
```

```
storm.local.dir: "/home/osboxes/Desktop/installation/storm/data"
java.library.path : "/usr/lib/jvm/java-7-openjdk-i386/"

supervisor.slots.ports:
  - 6700
  - 6701
  - 6702
  - 6703
```

**Remarque :** nous avons utilisé ici localhost mais on aurait pu reprendre l'adresse ip de la machine avec la commande **inet addr**.

La nouvelle version est celle de la Figure 36



```
storm.yaml (~/Desktop/Installation/storm/conf) - gedit
File Edit View Search Tools Documents
Open Save Undo
Documents storm.yaml
# distributed under the license is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the license for the specific language governing permissions and
# limitations under the license.

##### These MUST be filled in for a storm configuration
storm.zookeeper.servers:
  - "localhost"
#   - "server2"
#
storm.zookeeper.port: 2181
nimbus.host: "localhost"
nimbus.thrift.port: 6627
ui.port: 8772
storm.local.dir: "/home/osboxes/Desktop/installation/storm/data"
java.library.path : "/usr/lib/jvm/java-7-openjdk-i386/"

supervisor.slots.ports:
  - 6700
  - 6701
  - 6702
  - 6703

#
#
# ##### These may optionally be filled in:
#
## List of custom serializations
# topology.kryo.register:
```

Figure 36. Le fichier storm.yaml après modification

Il faut créer un répertoire **data** dans le répertoire storm de sorte que le paramètre **storm.local.dir** fasse référence à un répertoire existant (Figure 37).

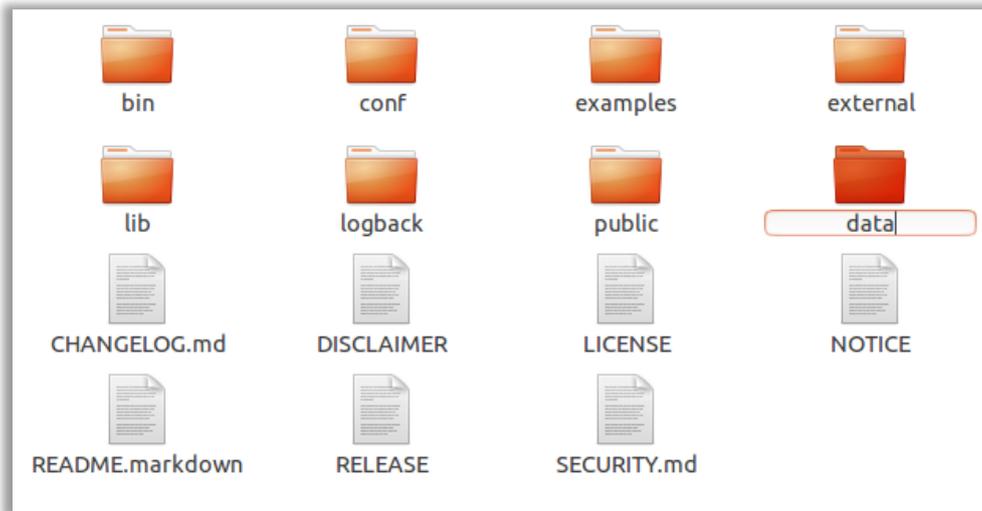


Figure 37. Création d'un répertoire data dans le répertoire storm

### Test de la configuration : démarrage de nimbus

Dans une console taper : storm nimbus

```
osboxes@osboxes: ~/Desktop/Installation/storm/bin
File Edit View Search Terminal Help
logback-core-1.0.13.jar:/home/osboxes/Desktop/Installation/storm/lib/carbonite-1.4.0.jar:/home/osboxes/Desktop/Installation/storm/lib/tools.cli-0.2.4.jar:/home/osboxes/Desktop/Installation/storm/lib/joda-time-2.0.jar:/home/osboxes/Desktop/Installation/storm/lib/closure-1.5.1.jar:/home/osboxes/Desktop/Installation/storm/lib/tools.logging-0.2.3.jar:/home/osboxes/Desktop/Installation/storm/lib/ring-core-1.1.5.jar:/home/osboxes/Desktop/Installation/storm/lib/disruptor-2.10.1.jar:/home/osboxes/Desktop/Installation/storm/lib/compojure-1.1.3.jar:/home/osboxes/Desktop/Installation/storm/lib/clj-time-0.4.1.jar:/home/osboxes/Desktop/Installation/storm/lib/slf4j-api-1.7.5.jar:/home/osboxes/Desktop/Installation/storm/lib/kryo-2.21.jar:/home/osboxes/Desktop/Installation/storm/lib/reflectasm-1.07-shaded.jar:/home/osboxes/Desktop/Installation/storm/lib/json-simple-1.1.jar:/home/osboxes/Desktop/Installation/storm/lib/jline-2.11.jar:/home/osboxes/Desktop/Installation/storm/lib/math.numeric-tower-0.0.1.jar:/home/osboxes/Desktop/Installation/storm/lib/jetty-6.1.26.jar:/home/osboxes/Desktop/Installation/storm/lib/commons-codec-1.6.jar:/home/osboxes/Desktop/Installation/storm/lib/hiccups-0.3.6.jar:/home/osboxes/Desktop/Installation/storm/lib/ring-servlet-0.3.11.jar:/home/osboxes/Desktop/Installation/storm/conf -Xmx1024m -Dlogfile.name=nimbus.log -Dlogback.configurationFile=/home/osboxes/Desktop/Installation/storm/logback/cluster.xml backtype=storm.daemon.nimbus
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar
```

Figure 38. Nimbus en fonctionnement

### Test de la configuration : démarrage du supervisor

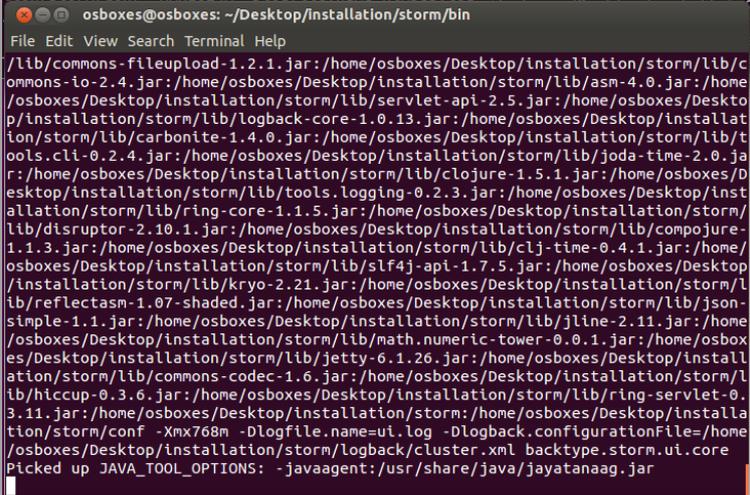
Dans une console taper : storm supervisor

```
osboxes@osboxes: ~/Desktop/Installation/storm/bin
File Edit View Search Terminal Help
ion/storm/lib/carbonite-1.4.0.jar:/home/osboxes/Desktop/Installation/storm/lib/tools.cli-0.2.4.jar:/home/osboxes/Desktop/Installation/storm/lib/joda-time-2.0.jar:/home/osboxes/Desktop/Installation/storm/lib/closure-1.5.1.jar:/home/osboxes/Desktop/Installation/storm/lib/tools.logging-0.2.3.jar:/home/osboxes/Desktop/Installation/storm/lib/ring-core-1.1.5.jar:/home/osboxes/Desktop/Installation/storm/lib/disruptor-2.10.1.jar:/home/osboxes/Desktop/Installation/storm/lib/compojure-1.1.3.jar:/home/osboxes/Desktop/Installation/storm/lib/clj-time-0.4.1.jar:/home/osboxes/Desktop/Installation/storm/lib/slf4j-api-1.7.5.jar:/home/osboxes/Desktop/Installation/storm/lib/kryo-2.21.jar:/home/osboxes/Desktop/Installation/storm/lib/reflectasm-1.07-shaded.jar:/home/osboxes/Desktop/Installation/storm/lib/json-simple-1.1.jar:/home/osboxes/Desktop/Installation/storm/lib/jline-2.11.jar:/home/osboxes/Desktop/Installation/storm/lib/math.numeric-tower-0.0.1.jar:/home/osboxes/Desktop/Installation/storm/lib/jetty-6.1.26.jar:/home/osboxes/Desktop/Installation/storm/lib/commons-codec-1.6.jar:/home/osboxes/Desktop/Installation/storm/lib/hiccups-0.3.6.jar:/home/osboxes/Desktop/Installation/storm/lib/ring-servlet-0.3.11.jar:/home/osboxes/Desktop/Installation/storm/conf -Xmx256m -Dlogfile.name=supervisor.log -Dlogback.configurationFile=/home/osboxes/Desktop/Installation/storm/logback/cluster.xml backtype=storm.daemon.supervisor
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar
```

Figure 39. Le superviseur en fonctionnement

## Test de la configuration : démarrage du ui

Dans une console taper : storm ui



```
osboxes@osboxes: ~/Desktop/Installation/storm/bin
File Edit View Search Terminal Help
./lib/commons-fileupload-1.2.1.jar:/home/osboxes/Desktop/Installation/storm/lib/commons-io-2.4.jar:/home/osboxes/Desktop/Installation/storm/lib/asm-4.0.jar:/home/osboxes/Desktop/Installation/storm/lib/servlet-api-2.5.jar:/home/osboxes/Desktop/Installation/storm/lib/logback-core-1.0.13.jar:/home/osboxes/Desktop/Installation/storm/lib/carbonite-1.4.0.jar:/home/osboxes/Desktop/Installation/storm/lib/tools.cli-0.2.4.jar:/home/osboxes/Desktop/Installation/storm/lib/joda-time-2.0.jar:/home/osboxes/Desktop/Installation/storm/lib/clojure-1.5.1.jar:/home/osboxes/Desktop/Installation/storm/lib/tools.logging-0.2.3.jar:/home/osboxes/Desktop/Installation/storm/lib/ring-core-1.1.5.jar:/home/osboxes/Desktop/Installation/storm/lib/disruptor-2.10.1.jar:/home/osboxes/Desktop/Installation/storm/lib/compojure-1.1.3.jar:/home/osboxes/Desktop/Installation/storm/lib/clj-time-0.4.1.jar:/home/osboxes/Desktop/Installation/storm/lib/slf4j-api-1.7.5.jar:/home/osboxes/Desktop/Installation/storm/lib/kryo-2.21.jar:/home/osboxes/Desktop/Installation/storm/lib/reflectasm-1.07-shaded.jar:/home/osboxes/Desktop/Installation/storm/lib/json-simple-1.1.1.jar:/home/osboxes/Desktop/Installation/storm/lib/jline-2.11.jar:/home/osboxes/Desktop/Installation/storm/lib/math.numeric-tower-0.0.1.jar:/home/osboxes/Desktop/Installation/storm/lib/jetty-6.1.26.jar:/home/osboxes/Desktop/Installation/storm/lib/commons-codec-1.6.jar:/home/osboxes/Desktop/Installation/storm/lib/hiccup-0.3.6.jar:/home/osboxes/Desktop/Installation/storm/lib/ring-servlet-0.3.11.jar:/home/osboxes/Desktop/Installation/storm:/home/osboxes/Desktop/Installation/storm/conf -Xmx768m -Dlogfile.name=ui.log -Dlogback.configurationFile=/home/osboxes/Desktop/Installation/storm/logback/cluster.xml backtype.storm.ui.core Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar
```

Figure 40. ui en fonctionnement

On peut ensuite vérifier le bon fonctionnement en se connectant sur :

<http://localhost:8772>

Ceci doit donner accès à une page web similaire à celle de la Figure 41.

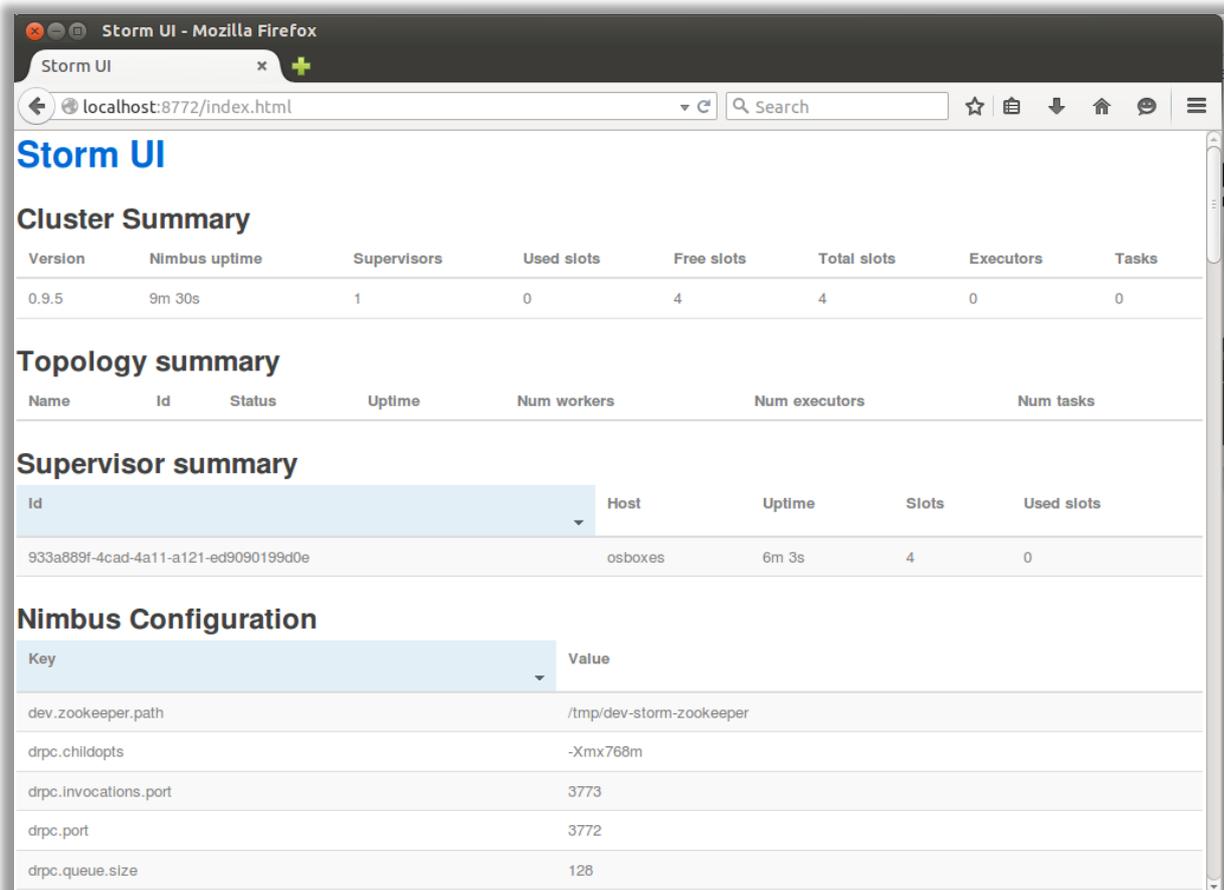
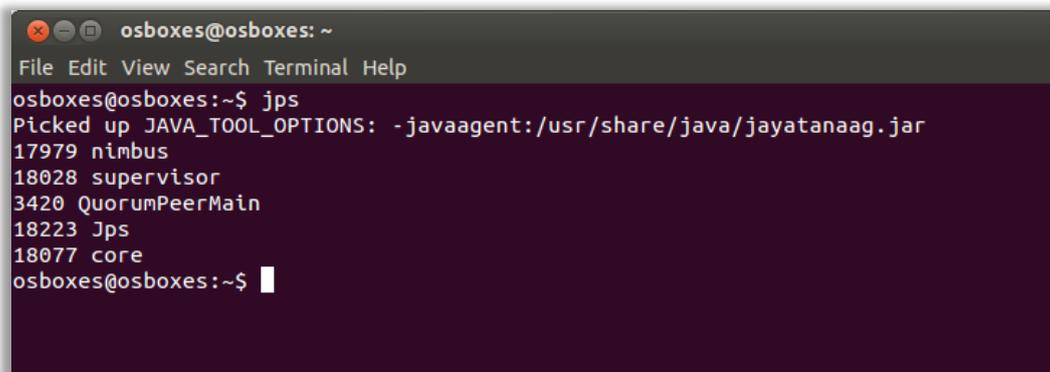


Figure 41. Accès à l'interface web

La deuxième vérification consiste à effectuer une commande `jps` dans un terminal. On doit alors obtenir un résultat identique à celui de la Figure 42 qui montre à la fois `nimbus` et `supervisor` dans la liste.



```
osboxes@osboxes: ~  
File Edit View Search Terminal Help  
osboxes@osboxes:~$ jps  
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar  
17979 nimbus  
18028 supervisor  
3420 QuorumPeerMain  
18223 Jps  
18077 core  
osboxes@osboxes:~$
```

Figure 42. Vérification par la commande `jps`

### **Étape 5. Installation de KAFKA**

La page de téléchargement est la suivante :

<http://kafka.apache.org/downloads.html>

La page est celle de la Figure 43.

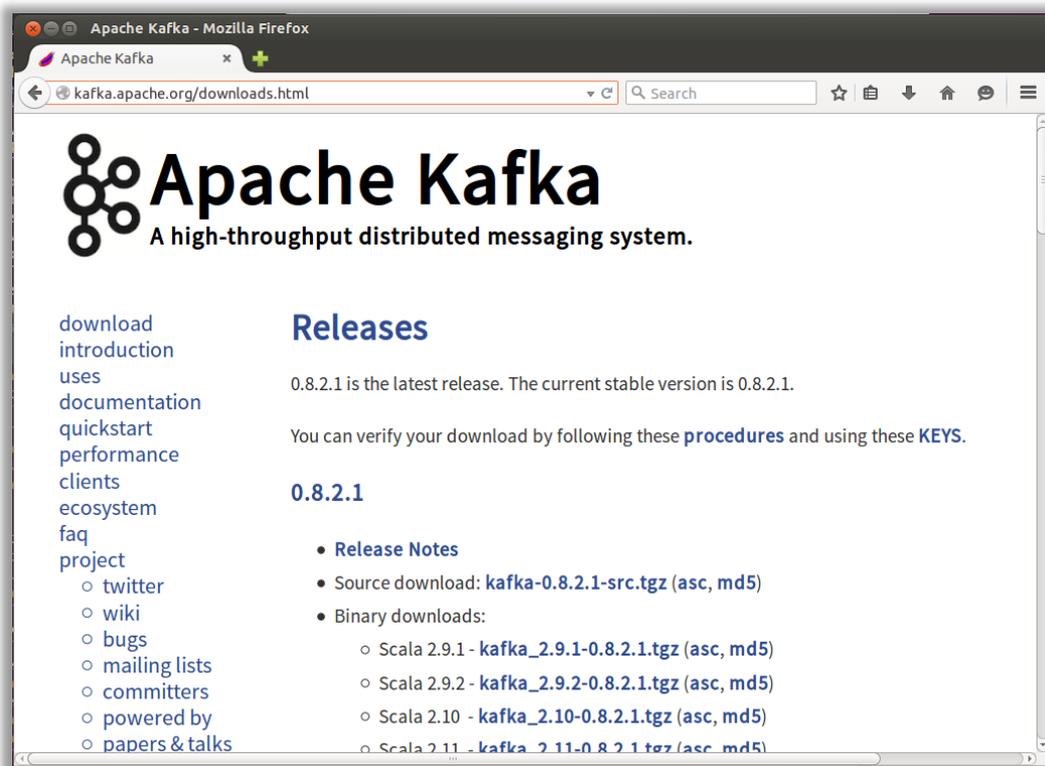


Figure 43. Section Download de Kafka

En aout 2015, la version courante est la version 2.11 que nous allons sortir de son archive dans le répertoire installation comme nous l'avons fait pour storm. Celui-ci doit alors ressembler au répertoire de la Figure 44.

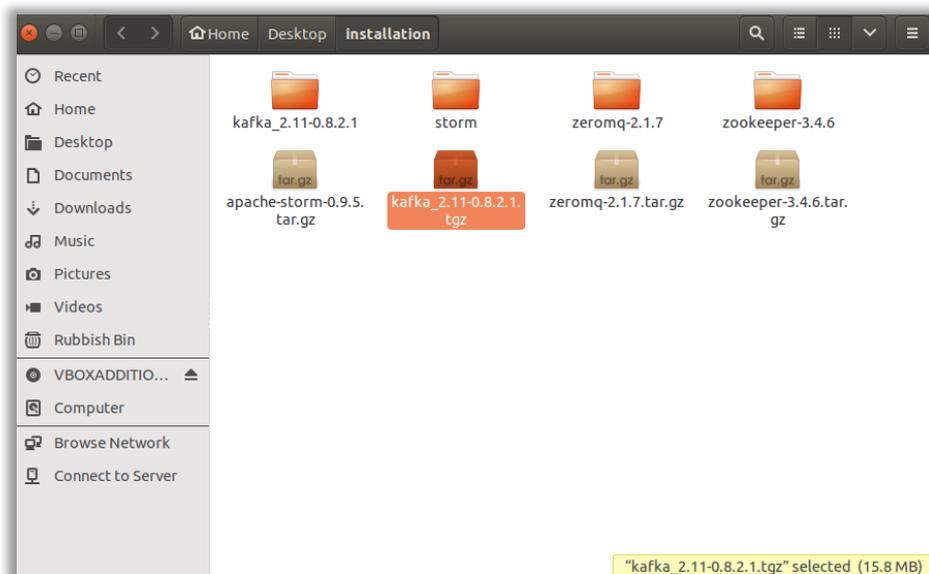


Figure 44. Kafka après téléchargement

Dans une console, on peut se rendre dans le répertoire **kafka** et taper **tree** pour obtenir un résultat similaire à celui de la figure 41.

```

osboxes@osboxes: ~/Desktop/installation/kafka_2.11-0.8.2.1
File Edit View Search Terminal Help
osboxes@osboxes:~/Desktop/installation/kafka_2.11-0.8.2.1$ tree
.
├── bin
│   ├── kafka-console-consumer.sh
│   ├── kafka-console-producer.sh
│   ├── kafka-consumer-offset-checker.sh
│   ├── kafka-consumer-perf-test.sh
│   ├── kafka-mirror-maker.sh
│   ├── kafka-preferred-replica-election.sh
│   ├── kafka-producer-perf-test.sh
│   ├── kafka-reassign-partitions.sh
│   ├── kafka-replay-log-producer.sh
│   ├── kafka-replica-verification.sh
│   ├── kafka-run-class.sh
│   ├── kafka-server-start.sh
│   ├── kafka-server-stop.sh
│   ├── kafka-simple-consumer-shell.sh
│   └── kafka-topics.sh
└── windows
    ├── kafka-console-consumer.bat
    ├── kafka-console-producer.bat
    ├── kafka-consumer-offset-checker.bat
    ├── kafka-consumer-perf-test.bat
    ├── kafka-mirror-maker.bat
    ├── kafka-preferred-replica-election.bat
    ├── kafka-producer-perf-test.bat
    └── kafka-reassign-partitions.bat

```

Figure 45. Résultat de la commande tree

Depuis le répertoire de kafka, la commande à taper dans le terminal est :

**./bin/kafka-server-start.sh config/server.properties**

Le résultat d'exécution doit être similaire à celui de la Figure 46.

```

osboxex@osboxex: ~/Desktop/installation/kafka_2.11-0.8.2.1
File Edit View Search Terminal Help
├── snappy-java-1.1.1.6.jar
├── zkclient-0.3.jar
├── zookeeper-3.4.6.jar
├── LICENSE
├── NOTICE
├── 4 directories, 63 files
osboxex@osboxex:~/Desktop/installation/kafka_2.11-0.8.2.1$ ./bin/kafka-server-start.sh config/server.properties
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar
[2015-08-12 15:42:48,148] INFO Verifying properties (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,212] INFO Property broker.id is overridden to 0 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,212] INFO Property log.cleaner.enable is overridden to false (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,217] INFO Property log.dirs is overridden to /tmp/kafka-logs (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,218] INFO Property log.retention.check.interval.ms is overridden to 300000 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,218] INFO Property log.retention.hours is overridden to 168 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,218] INFO Property log.segment.bytes is overridden to 1073741824 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,218] INFO Property num.io.threads is overridden to 8 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,218] INFO Property num.network.threads is overridden to 3 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,219] INFO Property num.partitions is overridden to 1 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,219] INFO Property num.recovery.threads.per.data.dir is overridden to 1 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,219] INFO Property port is overridden to 9092 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,219] INFO Property socket.receive.buffer.bytes is overridden to 102400 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,219] INFO Property socket.request.max.bytes is overridden to 104857600 (kafka.utils.VerifiableProperties)
[2015-08-12 15:42:48,219] INFO Property socket.send.buffer.bytes is overridden to 102400 (kafka.utils.VerifiableProperties)

```

Figure 46. Démarrage de kafka

Dans une console, on peut utiliser `jps` pour vérifier la présence de kafka dans la liste. Un résultat correct doit être similaire à celui de la Figure 47.

```

osboxex@osboxex: ~
File Edit View Search Terminal Help
osboxex@osboxex:~$ jps
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar
17979 nimbus
18028 supervisor
3420 QuorumPeerMain
18924 Jps
18793 Kafka
18077 core
osboxex@osboxex:~$

```

Figure 47. Vérification de la configuration

### **Etape 6. Pour se faciliter la tâche**

L'ensemble des commandes permettant de démarrer le cluster sont les suivantes :

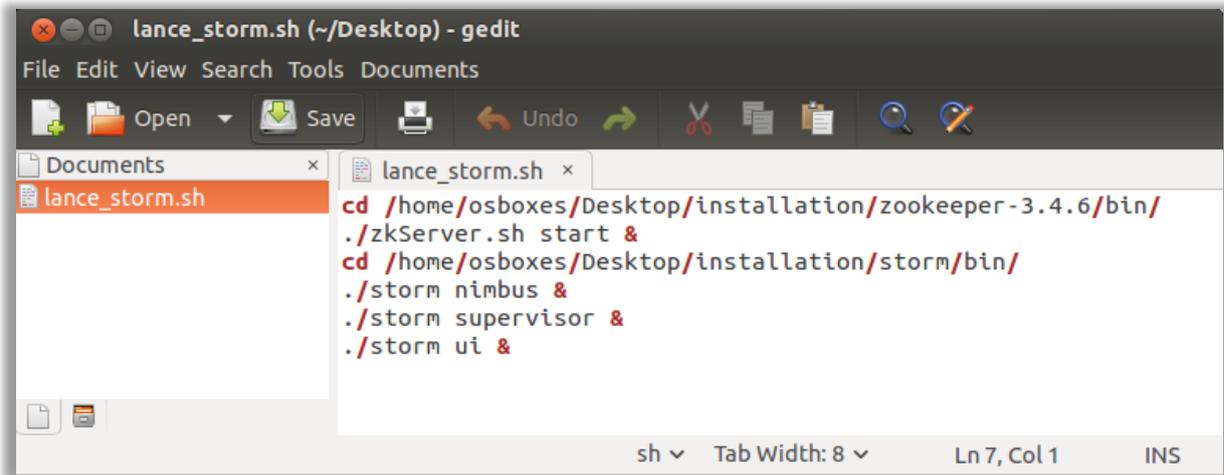
```

cd /home/osboxex/Desktop/installation/zookeeper-3.4.6/bin/
./zkServer.sh start &
cd /home/osboxex/Desktop/installation/storm/bin/

```

```
./storm nimbus &  
./storm supervisor &  
./storm ui &
```

Pour se simplifier la vie, on peut mettre ces commandes dans un fichier nommé lance\_storm.sh (Figure 48).



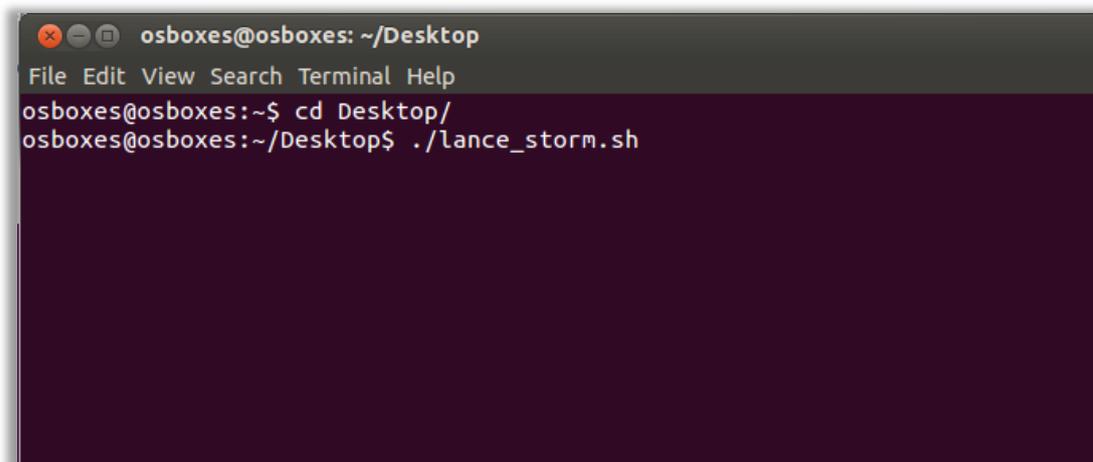
```
cd /home/osboxes/Desktop/installation/zookeeper-3.4.6/bin/  
./zkServer.sh start &  
cd /home/osboxes/Desktop/installation/storm/bin/  
./storm nimbus &  
./storm supervisor &  
./storm ui &
```

Figure 48. Création d'un fichier de lancement

On peut rendre ce fichier exécutable par exemple en faisant :

```
chmod 777 lance_storm.sh
```

On peut ensuite facilement démarrer le cluster Storm (Figure 49).



```
osboxes@osboxes:~$ cd Desktop/  
osboxes@osboxes:~/Desktop$ ./lance_storm.sh
```

Figure 49. Démarrage du cluster