**LOG8430**

**Architecture logicielle et conception avancée**

**Automne 2018**

TP4

SOA, REST, Infonuagique, Architectures Multi-niveaux pour le traitement de mégadonnées

Présenté par :

Alexandre Clark (1803508)

David Tremblay (1748125)

Félix Agagnier (1795792)

02 décembre 2018

École Polytechnique de Montréal

Département de Génie Informatique et Logiciel



Le présent document est un guide d’installation, de déploiement et de fonctionnement de notre application REST développé pour ce travail pratique.

**Liens vers les livrables du travail pratique**

* Répertoire GitHub du serveur: <https://github.com/Agagnier35/log8430-tp3-server>
* Répertoire Github du client: <https://github.com/Agagnier35/log8430-tp3-client>
* Vidéo Youtube montrant le fonctionnement du système: <https://www.youtube.com/watch?v=140wlGTXGCY&feature=youtu.be>

**Guide d’installation**

**Installation de la machine virtuelle**

1. Démarrer le logiciel ***Oracle VM VirtualBox.***
2. Cliquersur l’icône ** en haut à gauche pour commencer la configuration d’une nouvelle VM. Donner un nom à la VM (sans importance) et laisser le type (Linux) et la version (Ubuntu 64-bit) par défaut puis cliquer sur *Next*.
3. Entrer 4096 dans la boîte d’entrée indiquant l’espace mémoire de la VM puis cliquer sur ***Next***5 fois en laissant les valeurs par défaut à chaque fois. La VM est maintenant créée et visible à gauche de l’interface.
4. Exécuter un clic droit sur cette VM et sélectionner l’option ***Network***. Sélectionner ***Adapter 2*** et cocher le champ ***Enable Network Adapter 2*** et sélectionner ***Host-only Adapter*** dans le menu déroulant se trouvant juste en dessous.
5. Télécharger le fichier .iso à l’adresse suivante: <http://cdimage.ubuntu.com/ubuntu/releases/18.04/release/>. Noter l’endroit où le fichier se trouve.
6. Sélectionner maintenant l’option ***Storage***. Cliquer sur l’icône  à droite du texte ***Controller: IDE*** puis cliquer sur ***Choose disk*** dans la fenêtre qui s’ouvre. Sélectionner le fichier .iso de l’étape 5 puis cliquer sur ***Ok***.
7. Avec la VM sélectionnée (en bleue), cliquer sur l’icône  pour démarrer la VM. La VM prend un certain temps à démarrer. Une fois prête, sélectionner les options par défaut 3 fois (jusqu’à ***Install Ubuntu***). Après, cette étape, noter l’adresse IP de ***eps08***puis sélectionner l’option par défaut 6 fois suivi de***Continue***. Remplir les informations demandées (nom, mot de passe, etc.) et terminer la configuration. Celle-ci peut prendre plusieurs minutes avant de se compléter.

**Configuration de la machine virtuelle**

Entrer les commandes suivantes dans la console de la VM :

1. sudo apt-get upgrade
2. sudo add-apt-repository 'deb http://archive.ubuntu.com/ubuntu xenial main universe'
3. sudo apt-get update
4. sudo add-apt-repository ppa:webupd8team/java
5. sudo apt update
6. sudo apt install oracle-java8-installer
7. java -version

**Installation de TOMCAT**

Entrer les commandes suivantes dans la console de la VM configurée:

1. sudo mkdir /opt/tomcat
2. cd /opt/tomcat
3. Aller à l’adresse suivante: <https://tomcat.apache.org/download-80.cgi> puis faire un clic droit sur le fichier tar.gz et copier l’adresse du lien (). L’adresse est dynamique et cette étape est donc nécessaire.
4. wget [coller l’adresse du point 3]
5. sudo tar xvzf apache-tomcat-8.5.35.tar.gz
6. sudo vim ~/.bashrc
7. Taper “**i**” pour modifier le fichier. Entrer les deux lignes suivantes à la fin puis taper “**:wq**” suivi de “**echap”** pour quitter:

**export JAVA\_HOME=/usr/lib/jvm/java-8-oracle**

**export CATALINA\_HOME=/opt/tomcat/apache-tomcat-8.5.35**

1. . ~/.bashrc
2. echo $JAVA\_HOME
3. sudo vim $CATALINA\_HOME/conf/tomcat-users.xml
4. Taper “**i**” pour éditer le fichier. Ajouter les lignes suivantes avant la dernière ligne du fichier:

**<role rolename="manager-gui"/>**

**<role rolename="admin-gui"/>**

**<role rolename="manager-script"/>**

**<user username="tomcat" password="s3cret" roles="manager-gui,admin-gui,manager-script"/**

1. sudo vim $CATALINA\_HOME/webapps/manager/META-INF/context.xml puis taper “**i**” et commenter la balise **<Valve>** vers la fin du fichier.
2. sudo vim $CATALINA\_HOME/bin/setenv.sh puis taper “**i**” et ajouter la ligne suivante au nouveau fichier créé:

**export JAVA\_OPTS="-Xms3G -Xmx3G -Dspark.host.address=[adresse IP d’esp08] -Dcassandra.host.address=[adresse IP d’esp08]"**

1. sudo vim $CATALINA\_HOME/webapps/manager/WEB\_INF/web.xml puis modifier les balises suivantes dans la balise **<multipart-config>**: **<max-file-size>5248000**

**<max-request-size>5248000**

1. sudo $CATALINA\_HOME/bin/startup.sh
2. Le server est maintenant démarré et se trouve à l’adresse IP d’eps08 de l’étape 7 de la configuration de la VM sur le port 8080 (xxx.xxx.xxx.xxx:8080).
3. Séléctionner  sur l’adresse du site et entrer l’utilisateur **tomcat** et le mot de passe **s3cret**. Comme défini dans le fichier tomcat-users.xml

**Installation de Cassandra**

Entrer les commandes suivantes:

1. echo "deb http://www.apache.org/dist/cassandra/debian 311x main" | sudo tee -a /etc/apt/sources.list.d/cassandra.sources.list
2. curl https://www.apache.org/dist/cassandra/KEYS | sudo apt-key add -
3. sudo apt-key adv --keyserver pool.sks-keyservers.net --recv-key A278B781FE4B2BDA
4. sudo apt-get update
5. sudo apt-install cassandra
6. sudo vim /etc/cassandra/cassandra.yaml. Taper “**i**” puis remplacer **rpc\_adress: localhost** par **rpc\_adress: [adresse IP d’esp08]**
7. sudo service cassandra start

**Installation de Spark**

Entrer les commandes suivantes:

1. sudo apt-get install scala
2. sudo mkdir /opt/spark/
3. cd /opt/spark/
4. Aller à l’adresse suivante: <https://spark.apache.org/downloads.html> puis choisir la release 2.3.2 à l’étape 1, laisser l’étape 2 par défaut et cliquer sur le lien de l’étape 3. Copier l’adresse du lien affiché sur la prochaine page. Cette étape est nécessaire puisque l’adresse de ce lien est dynamique.
5. sudo wget [coller l’adresse du lien du point 4]
6. sudo tar xvf spark-2.3.2-bin-hadoop2.7.tgz
7. sudo spark-2.3.2-bin-hadoop2.7/bin/spark-shell
8. sudo vim ~/.bashrc puis taper “**i**” et ajouter la ligne suivante à la fin du fichier:

**export $SPARK\_HOME=/opt/spark/spark-2.3.2-bin-hadoop2.7**

1. . ~/.bashrc
2. cd $SPARK\_HOME/conf
3. cp spark-env.sh.template spark-env.sh
4. sudo vim spark-env.sh puis taper “**i**” et ajouter les lignes suivantes à la fin du fichier:

**SPARK\_MASTER\_HOST=[adresse ip d’esp08]**

**SPARK\_LOCAL\_IP=[adresse IP d’esp08]**

**SPARK\_MASTER\_PORT=7077**

**SPARK\_MASTER\_WEBUI\_PORT=7078**

**SPARK\_WORKER\_WEBUI\_PORT=7079**

**SPARK\_WORKER\_INSTANCES=2**

1. sudo bash $SPARK\_HOME/sbin/start\_master.sh
2. sudo bash $SPARK\_HOME/sbin/start\_slave.sh spark://[adresse IP d’esp08]:7077

**Guide de déploiement**

Ouvrir un terminal et entrer les commandes suivantes:

1. cd [endroit ou notre dossier du serveur se trouve]
2. mvn clean package
3. Se connecter avec l’utilisateur **tomcat** et le mot de passe **s3cret** à l’adresse [eps08:8080/manager/html] puis sélectionner le fichier tp3.war dans le dossier du serveur (tp3-restApi/target). Finalement, cliquer sur ***Deploy****.*
4. *cd [endroit ou notre dossier du client se trouve]*
5. mvn clean package
6. cd [dossier ou le client se trouve]/target/
7. java -jar tp3-1.0-SNAPSHOT-jar-with-dependencies.jar
8. Lorsque la console demande l’URL du service, entrer: "http://[eps08]:8080/tp3/rest"
9. L’application est maintenant prête à être utilisée !

**Guide d'utilisation**

Un lien vers un vidéo Youtube démonstratif se trouve au début de ce document pour une démonstration.

Le client peut entrer 4 commandes:

* exit: Pour quitter l’application
* help: Pour obtenir une courte description des commandes qui peuvent être utilisées
* get: Retourne le produit ayant été le plus acheté dans les factures présentes dans la base de données sous le format suivant: **[statut de la requête (200 si tout va bien)]** Name: **[nom du produit]**, price: **[prix du produit]**, bought**: [nombre de fois achetées au total]** times
* add: Pour ajouter une facture dans la base de données. Une fois la commande entrée, l’application demande le lien vers un fichier csv à ajouter (une facture). Les factures ont trois colonnes avec les entêtes **productName**, **price** et **quantity**. Deux exemples de factures se trouvent dans le répertoire Github de notre projet (lien au début du document). Les produits dans le fichier csv se retrouveront dans notre base de données par la suite. Les produits des factures (avec leur quantité) se retrouvent dans la table invoices.invoiceitem et une liste des produits (sans quantité ni doublon) se retrouve dans invoices.product. Le status code 201 signifie que l'opération s'est effectué avec succès.