

Création d'un environnement de développement

HYPERLEDGER

Ce tutoriel a été réalisé par Mathieu Indych.

Si vous désirez que ce tutoriel fonctionne merci de respecter scrupuleusement les étapes mentionnées !

Etape 1 : Installation de la machine virtuelle.

Utiliser la dernière version de Virtualbox téléchargeable ici :

<https://www.virtualbox.org/wiki/Downloads>

Etape 2 : Création de la machine virtuelle :

Attention : Utiliser UNIQUEMENT UBUNTU.

Créer une machine virtuelle classique suivant ce tuto sur Youtube :

<https://www.youtube.com/watch?v=QbmRXJKsys>

ISO UBUNTU téléchargeable ici :

<https://www.ubuntu.com/download/desktop>

(Bouton download du bas)

Etape 3 : Environnement de développement

Commandes à copier-coller une par une en utilisateur normal ET NON EN ROOT :

PARTIE I :

```
cd Documents/
sudo apt-get install curl
sudo apt-get install golang-go
export GOPATH=$HOME/go
export PATH=$PATH:$GOPATH/bin
sudo apt-get install nodejs
sudo apt-get install npm
sudo apt-get install python
sudo apt-get install docker

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -

sudo add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
sudo apt-get update
apt-cache policy docker-ce
sudo apt-get install -y docker-ce
sudo apt-get install docker-compose
sudo apt-get upgrade
```

PARTIE II :

```
sudo curl -sSL https://goo.gl/6wtTN5 | sudo bash -s 1.1.0
sudo chmod 777 -R fabric-samples
```

PARTIE III :

Génération du "First-Network":

```
cd fabric-samples/first-network
sudo ./byfn.sh generate
```

PARTIE IV :

Lancement du network:

```
sudo ./byfn.sh up
```

Cette étape est fatidique vous devez impérativement obtenir l'écran de fin suivant, si ce n'est pas le cas veuillez vérifier que vous avez bien exécuté les différentes commandes à l'aide de la commande :

```
history
```

Qui vous aidera à voir l'ensemble des commandes exécutées durant le tutoriel.

ECRAN DE FIN :

```
===== Chaincode is installed on peer1.org2 =====
Querying chaincode on peer1.org2...
CORE_PEER_TLS_ROOTCERT_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.example.com/peers/peer0.org2.example.com/tls/ca.crt
CORE_PEER_TLS_KEY_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/server.key
CORE_PEER_LOCALMSPID=Org2MSP
CORE_VM_ENDPOINT=unix:///host/var/run/docker.sock
CORE_PEER_TLS_CERT_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/server.crt
CORE_PEER_TLS_ENABLED=true
CORE_PEER_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.example.com/users/Admin@org2.example.com/msp
CORE_PEER_ID=cli
CORE_LOGGING_LEVEL=INFO
CORE_PEER_ADDRESS=peer1.org2.example.com:7051
===== Querying on peer1.org2 on channel 'mychannel'... =====
Attempting to Query peer1.org2 ...3 secs
+ peer chaincode query -C mychannel -n mycc -c '{"Args":["query","a"]}'
+ res=0
+ set +x

2019-03-23 15:24:29.555 UTC [chaincodeCmd] checkChaincodeCmdParams -> INFO 001 Using default escc
2019-03-23 15:24:29.555 UTC [chaincodeCmd] checkChaincodeCmdParams -> INFO 002 Using default vsc
Query Result: 90
2019-03-23 15:24:43.936 UTC [main] main -> INFO 003 Exiting....
===== Query on peer1.org2 on channel 'mychannel' is successful =====

===== All GOOD, BYFN execution completed =====

      END
mathieu@mathieu-VirtualBox:~/Documents/fabric-samples/first-network$
```

Si cet écran est différent et que vous ne bénéficiez pas de la mention fléchée veuillez me contacter au 0663638745 ou par email : mathieu.indych@esme.fr ou mathieuindych@gmail.com.

Temps moyen estimé pour l'exécution complète du tutoriel : 35 min suivant les connexions internet.

Bonne chance à tous.

Mathieu Indych, Cybersécurité