**LabHyper - Build and deploy a blockchain Fabcar network with Hyperledger Fabric**

[**1.**     **Giới thiệu**](http://cnttbkdn.com:7777/mod/resource/tknetlab/LabHyper%20-%20Build%20a%20Fabcar%20Blockchain%20Network%20with%20Hyperledger%20Fabric.htm#_Toc102926998)

[**2.**     **Cài đặt máy ảo VMHyperledger**](http://cnttbkdn.com:7777/mod/resource/tknetlab/LabHyper%20-%20Build%20a%20Fabcar%20Blockchain%20Network%20with%20Hyperledger%20Fabric.htm#_Toc102926999)

[**3.**     **Triển khai hạ tầng mạng First Network  và cài đặt Fabcar chaincode lên tất cả các nút trong kênh**](http://cnttbkdn.com:7777/mod/resource/tknetlab/LabHyper%20-%20Build%20a%20Fabcar%20Blockchain%20Network%20with%20Hyperledger%20Fabric.htm#_Toc102927000)

[**4.**     **Truy vấn Fabcar chaincode bằng lệnh (CLI)**](http://cnttbkdn.com:7777/mod/resource/tknetlab/LabHyper%20-%20Build%20a%20Fabcar%20Blockchain%20Network%20with%20Hyperledger%20Fabric.htm#_Toc102927001)

[4.1         Cách 1: Dùng lệnh gộp](http://cnttbkdn.com:7777/mod/resource/tknetlab/LabHyper%20-%20Build%20a%20Fabcar%20Blockchain%20Network%20with%20Hyperledger%20Fabric.htm#_Toc102927002)

[4.2         Cách 2: Sử dụng docker exec -it cli bash](http://cnttbkdn.com:7777/mod/resource/tknetlab/LabHyper%20-%20Build%20a%20Fabcar%20Blockchain%20Network%20with%20Hyperledger%20Fabric.htm#_Toc102927003)

[**5.**     **Dừng mạng Fabric**](http://cnttbkdn.com:7777/mod/resource/tknetlab/LabHyper%20-%20Build%20a%20Fabcar%20Blockchain%20Network%20with%20Hyperledger%20Fabric.htm#_Toc102927004)

[**6.**     **Kết luận**](http://cnttbkdn.com:7777/mod/resource/tknetlab/LabHyper%20-%20Build%20a%20Fabcar%20Blockchain%20Network%20with%20Hyperledger%20Fabric.htm#_Toc102927005)

[**7.**     **Tham khảo**](http://cnttbkdn.com:7777/mod/resource/tknetlab/LabHyper%20-%20Build%20a%20Fabcar%20Blockchain%20Network%20with%20Hyperledger%20Fabric.htm#_Toc102927006)

**1.    Giới thiệu**

§  Nội dung bài lab nhằm triển khai hệ thống mạng blockchain fabcar

§  Fabcar là một cơ sở dữ liệu của các bản ghi về ô tô được lưu trữ trong sổ cái của một mạng Fabric. Tổ chức lưu trữ dữ liệu của fabcar trên blockchain tương tự trên cơ sở dữ liệu truyền thống: sử dụng bảng, được lập chỉ mục với Mã ô tô (CarID) và Thông tin nhà sản xuất (Model), Màu (Color) và Chủ sở hữu (Owner) cũng từng chiếc xe.

§  Fabcar chaincode là nơi chứa logic kinh doanh và sẽ thực thi khi được gọi. Sổ cái chỉ được cập nhật khi chaincode được gọi.

§  Fabcar chaincode có thể chạy trên bất kỳ mạng Fabric nào. Trong phần này triển khai Fabcar chạy trên first-network hoặc test-network..

§  Xem thông tin Fabcar chaincode: fabric-samples-v1/chaincode/fabcar

§  Chaincode có thể được viết bằng nhiều ngôn ngữ. Phần này triển khai chaincode được viết bằng Golang. Chaincode follow theo một mẫu cụ thể được xác định trong Hyperledger Fabric. Đó là cấu trúc dữ liệu, các hàm Init() và Invoke(), cộng với một số hàm sẽ được gọi bởi Invoke().

Xem thêm: https://viblo.asia/p/mot-ung-dung-tren-hyperledger-fabric-se-trong-nhu-the-nao-ORNZqGxN50n

**2.    Bước 1. Cài đặt và chạy máy ảo VMHyperledger**

§  Tải máy ảo VMHyperledger02 tại LINK06

§  Giải nén file tải về tại thư mục gốc ổ đĩa.

§  Cài đặt VMware Workstation. Chạy VMWare.

§  Từ cửa sổ VMWare, chọn menu File/Open, đến thư mục VMHyperledger02, mở file \*.vmx và khởi động máy ảo.

§  Sử dụng công cụ SecureCRT hoặc công cụ khác truy cập từ xa đến máy ảo theo địa chỉ IP: 192.168.1.10, port 22, username: ubuntu, pass: 123456

**3.    Bước 2. Triển khai hạ tầng mạng First Network  và cài đặt Fabcar chaincode lên tất cả các nút trong kênh**

* Triển khai hạ tầng mạng First Network

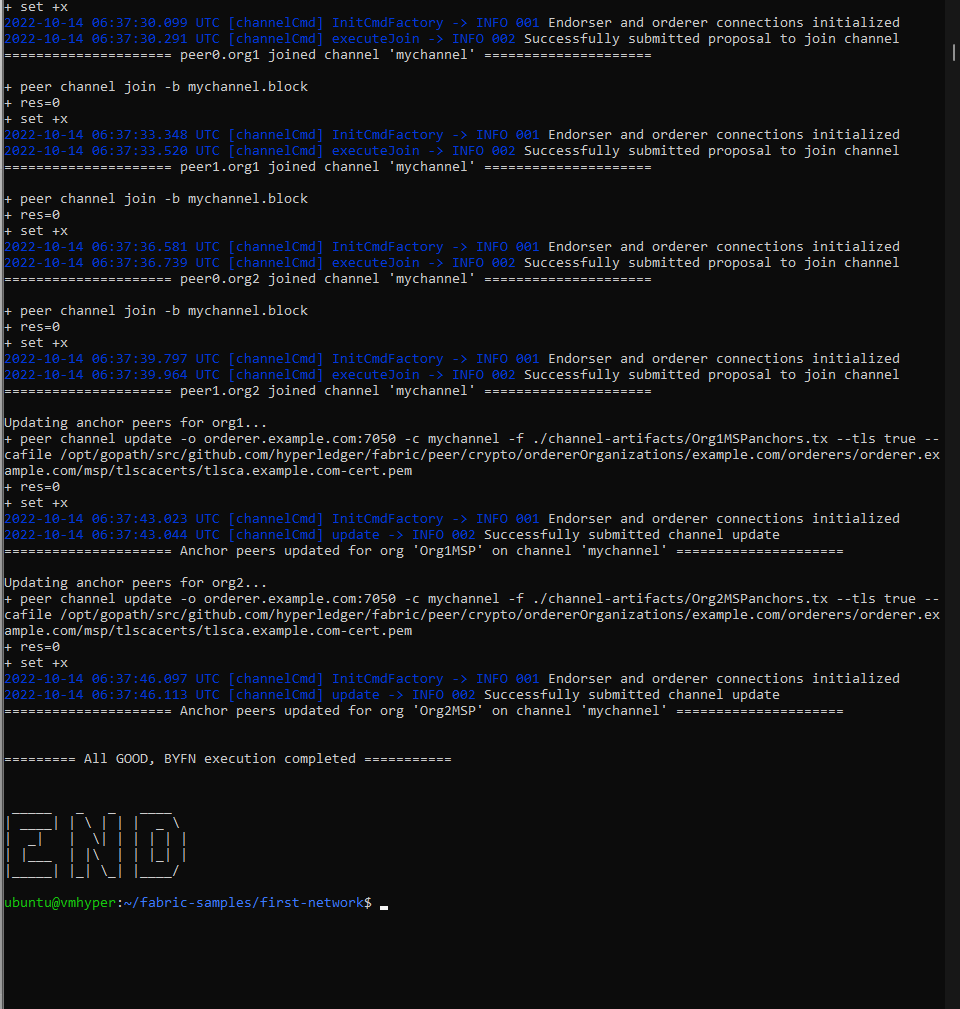
**ubuntu@vmhyper:~$ ./5.reboot.sh**

**ubuntu@vmhyper:~$ sudo chmod 666 /var/run/docker.sock**

**ubuntu@vmhyper:~$ cd $HOME/fabric-samples-v1-v1/first-network**

**ubuntu@vmhyper:~/fabric-samples-v1-v1/first-network$ echo y | ./byfn.sh down**

**ubuntu@vmhyper:~/fabric-samples-v1-v1/irst-network$ echo y | ./byfn.sh up -a -n -s couchdb**

****

§  Fabcar chaincode được cài đặt trên các nút tham gia kênh thông qua cli container với các biến môi trường được thiết lập.

§  Phần này sẽ cài đặt chaincode trên 4 nút.

**ubuntu@vmhyper:~$ sudo chmod 666 /var/run/docker.sock**

**ubuntu@vmhyper:~$ cd $HOME/fabric-samples-v1/fabcar**

**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$ export GOPATH=$HOME/go**

**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$ export PATH=$PATH:$GOPATH/bin**

**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$ ls**

apiserver   javascript-low-level  package-lock.json  typescript

javascript  node\_modules          startFabric.sh

**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$ ./startFabric.sh**

Kết quả trạng thái sau khi *startFabric.sh* được thực thi.

§  Tất cả các nút tham gia *mychannel*

§  *Fabcar chaincode* được cài đặt trên tất cả các nút tham gia kênh

§  *Fabcar chaincode* được khởi tạo trên mychannel

**Description: Description: Description: Description: Description: Description: Description: Graphical user interface

Description automatically generated**

**Text

Description automatically generated**

**Text

Description automatically generated**

**Text

Description automatically generated**

**4.    Bước 3. Truy vấn Fabcar chaincode bằng lệnh (CLI)**

§  Các biến môi trường mặc định trên CLI container trỏ đến **peer0.org1**. Khi ta muốn truy cập các nút khác, cần phải thay đổi các tham số liên quan.

§  Các bước thực hiện:

₋      Truy vấn tất cả bản ghi ô tô trên sổ cái

₋      Truy vấn xe theo CarID

₋      Thêm mới một bản ghi vào sổ cái

₋      Truy vấn tất cả bản ghi ô tô trên sổ cái để xem bản ghi vừa được thêm mới

₋      Thay đổi chủ nhân của một xe ô tô bất kỳ nào đó

₋      Truy vấn lại sổ cái và xem sự thay đổi

**4.1         Cách 1: Dùng lệnh gộp**

***1) Truy vấn tất cả bản ghi ô tô trên sổ cái (từ peer1.org1)***

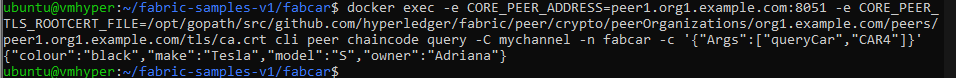
**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$ docker exec cli peer chaincode query -C mychannel -n fabcar -c '{"Args":["queryAllCars"]}'**

**Text

Description automatically generated**

***2) Truy vấn xe theo CarID (từ peer1.org1)***

**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$ docker exec -e CORE\_PEER\_ADDRESS=peer1.org1.example.com:8051 -e CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/peers/peer1.org1.example.com/tls/ca.crt cli peer chaincode query -C mychannel -n fabcar -c '{"Args":["queryCar","CAR4"]}'**

****

***3) Thêm mới một bản ghi vào sổ cái (from peer1.org1)***

**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$ docker exec -e CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.example.com/users/Admin@org2.example.com/msp -e CORE\_PEER\_ADDRESS=peer1.org2.example.com:10051 -e CORE\_PEER\_LOCALMSPID="Org2MSP" -e 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 cli peer chaincode query -C mychannel -n fabcar -c '{"Args": ["queryAllCars"]}'**

₋      Biến môi trường trỏ tới peer1.org1

₋      Vì chính sách chứng thực yêu cầu hai tổ chức đều xác nhận, gửi lệnh này đến peer0.org1 và peer0.org2 kèm với chứng chỉ TLS của chúng (tùy chọn tlsRootCertFiles).

Graphical user interface, text

Description automatically generated

***4) Truy vấn tất cả bản ghi ô tô trên sổ cái để xem bản ghi vừa được thêm mới (từ peer1.org2)***

**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$**

**docker exec -e CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.example.com/users/Admin@org2.example.com/msp -e CORE\_PEER\_ADDRESS=peer1. org2.example.com:10051 -e CORE\_PEER\_LOCALMSPID = "Org2MSP" –e 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 cli peer chaincode query -C mychannel -n fabcar -c '{"Args": ["queryAllCars"]}'**

**Graphical user interface, text

Description automatically generated**

***5) Thay đổi chủ sở hữu của một xe ô tô bất kỳ nào đó (từ peer0.org1)***

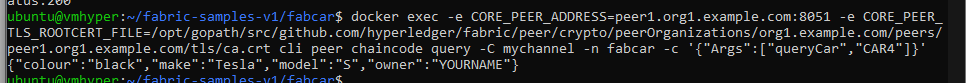
**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$ docker exec cli peer chaincode invoke -o orderer.example.com:7050 --tls true --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem -C mychannel -n fabcar --peerAddresses peer0.org1.example.com:7051 --tlsRootCertFiles /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/ca.crt --peerAddresses peer0.org2.example.com:9051 --tlsRootCertFiles /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.example.com/peers/peer0.org2.example.com/tls/ca.crt -c '{"Args":["changeCarOwner", "CAR4", "YOURNAME"]}'**

**Text

Description automatically generated**

***6) Truy vấn lại sổ cái và xem sự thay đổi***

**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$ docker exec -e CORE\_PEER\_ADDRESS=peer1.org1.example.com:8051 -e CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/peers/peer1.org1.example.com/tls/ca.crt cli peer chaincode query -C mychannel -n fabcar -c '{"Args":["queryCar","CAR4"]}'**

****

**4.2         Cách 2: Sử dụng docker exec -it cli bash**

***1) Truy vấn tất cả bản ghi ô tô trên sổ cái (từ peer0.org1)***

**ubuntu@vmhyper:~/fabric-samples-v1/fabcar$ docker exec -it cli bash**

**bash-5.1# peer chaincode query -C mychannel -n fabcar -c '{"Args":["queryAllCars"]}'**

**Graphical user interface, text

Description automatically generated**

***2) Truy vấn xe theo CarID (từ peer1.org1)***

**bash-5.1# CORE\_PEER\_ADDRESS=peer1.org1.example.com:8051**

**bash-5.1# CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/peers/peer1.org1.example.com/tls/ca.crt**

**bash-5.1# peer chaincode query -C mychannel -n fabcar -c '{"Args": ["queryCar", "CAR4"]}'**

****

***3) Thêm mới một bản ghi vào sổ cái (from peer0.org2)***

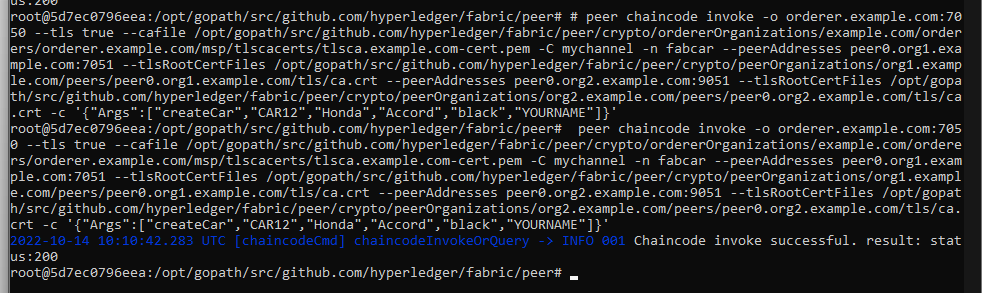
**bash-5.1# CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.example.com/users/Admin@org2.example.com/msp**

**bash-5.1# CORE\_PEER\_ADDRESS=peer0.org2.example.com:9051**

**bash-5.1# CORE\_PEER\_LOCALMSPID="Org2MSP"**

**bash-5.1# 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**

**bash-5.1# peer chaincode invoke -o orderer.example.com:7050 --tls true --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem -C mychannel -n fabcar --peerAddresses peer0.org1.example.com:7051 --tlsRootCertFiles /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/ca.crt --peerAddresses peer0.org2.example.com:9051 --tlsRootCertFiles /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.example.com/peers/peer0.org2.example.com/tls/ca.crt -c '{"Args":["createCar","CAR12","Honda","Accord","black","YOURNAME"]}'**

****

***4) Truy vấn tất cả bản ghi ô tô trên sổ cái để xem bản ghi vừa được thêm mới (từ peer1.org2)***

**bash-5.1# CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.example.com/users/Admin@org2.example.com/msp**

**bash-5.1# CORE\_PEER\_ADDRESS=peer1.org2.example.com:10051**

**bash-5.1# CORE\_PEER\_LOCALMSPID="Org2MSP"**

**bash-5.1# CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.example.com/peers/peer1.org2.example.com/tls/ca.crt**

**peer chaincode query -C mychannel -n fabcar -c '{"Args": ["queryAllCars"]}'**

**Text

Description automatically generated**

***5) Thay đổi chủ sở hữu của một xe ô tô bất kỳ nào đó (từ peer0.org1)***

**bash-5.1# CORE\_PEER\_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/users/Admin@org1.example.com/msp**

**bash-5.1# CORE\_PEER\_ADDRESS=peer0.org1.example.com:7051**

**bash-5.1# CORE\_PEER\_LOCALMSPID="Org1MSP"**

**bash-5.1# CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/ca.crt**

**bash-5.1# peer chaincode invoke -o orderer.example.com:7050 --tls true --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem -C mychannel -n fabcar --peerAddresses peer0.org1.example.com:7051 --tlsRootCertFiles /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/ca.crt --peerAddresses peer0.org2.example.com:9051 --tlsRootCertFiles /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org2.example.com/peers/peer0.org2.example.com/tls/ca.crt -c '{"Args":["changeCarOwner", "CAR12", "YOURNAME"]}'**

**A picture containing background pattern

Description automatically generated**

***6) Truy vấn lại sổ cái và xem sự thay đổi***

**bash-5.1# CORE\_PEER\_ADDRESS=peer1.org1.example.com:8051**

**bash-5.1# CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOrganizations/org1.example.com/peers/peer1.org1.example.com/tls/ca.crt**

**peer chaincode query -C mychannel -n fabcar -c '{"Args":["queryCar","CAR4"]}'**

**A screen shot of a computer

Description automatically generated with low confidence**

Trong thực tế, ta sẽ sử dụng ứng dụng client, thông qua SDK để thực hiện các thao tác với chaincode thay vì gõ lệnh trên terminal.

**5.    Bước 4. Dừng mạng Fabric**

**ubuntu@vmhyper~ $ cd $HOME/fabric-samples-v1/fabcar**

**ubuntu@vmhyper:~/fabric-samples-v1/first-network$ ./networkDown.sh**

**6.    Kết luận**

1)     Chạy mạng first-network hoặc test-network

2)     Cài đặt fabcar chaincode trên mạng

3)     Thực hiện các lệnh (CLI) để xem/sửa/truy vấn các bản ghi trên fabcar network

**7.    Tham khảo**

[1]     https://developpaper.com/develop-your-first-blockchain-application-with-hyperledger-fabric/

[2]     https://hyperledger-fabric.readthedocs.io/en/release-1.2/write\_first\_app.html

[3]     https://www.youtube.com/watch?v=TEHWOKVN9JA

----------------------------------------------------------