Pract 8 :

Su

go version

docker –version

docker-compose –version

node -v

npm -v

docker stop $(docker ps -a -q)

docker rm $(docker ps -a -q)

docker rmi $(docker images -q) -f

export GOPATH=$HOME/go

cd $GOPATH

mkdir src

cd src

mkdir github.com

cd github.com/

mkdir hyperledger

cd hyperledger/

ls

cd $GOPATH/src/github.com/hyperledger/

git clone <https://github.com/hyperledger/fabric-samples.git>

cd fabric-samples

git checkout release-1.4

cd $GOPATH/src/github.com/hyperledger/fabric-samples/first-network

chmod +x docker-compose-cli.yaml

gedit docker-compose-cli.yaml

cd ..

cd bin/

cp cryptogen ../first-network/

cd ..

cd first-network/

./byfn.sh generate

./byfn.sh up

docker exec -it cli bash

export CORE\_PEER\_MSCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOgranizations/org1.example.com/users/Admin@org1.example.com/msp

export CORE\_PEER\_ADDRESS=peer0.org1.example.com:7051

export CORE\_PEER-LOCALMSPID=”Org1MSP”

export CORE\_PEER\_TLS\_ROOTCERT\_FILE=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/peerOgranizations/org1.example.com/peers/peer0.org.example.com/tls/ca.crt

peer chaincode install -n evmcc -l golang -v 0 -p github.com/hyperledger/fabric-chaincode-evm/evmcc

peer chaincode instantiate -n evmcc -v 0 -c mychannel -c ‘{“Args”:[]}’ -o orderer.example.com:7050 –tls –cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem

exit

export FABPROXY\_CONFIG=${GOPATH}/src/github.com/hyperledger/fabric-chaincode-evm/examples/first-network-sdk-config.yaml

export FABPROXY\_USER=User1

export FABPROXY\_ORG=Org1

export FABPROXY\_CHANNEL=mychannel

export FABPROXY\_CCID=evmcc

export PORT=5000

cd fabric-chanincode-evm/

cd $GOPATH/src/github.com/hyperledger/fabric-chaincode-evm/

go mod init

go mod tidy -e

go mod vendor

go build -o fab3 ./fabproxy/cmd

you can then run the proxy ./fab3 this will start Fab3 at

<https://localhost:5001>