

BLC C4 - Drug Counterfeiting Problem

Instruction DOC

Saurav Savarn

1. Project Directory Structure

The terminal window shows the command `spartans@spartans-Standard-PC-Q35-ICH9-2009: ~/workspace/pharma-net$ tree -D` being run. The output displays the directory structure:

```
[Jul 18 20:58] .  
└── [Jul 18 10:30] application  
    ├── [Jul 18 12:10] 10_updateShipment.js  
    ├── [Jul 18 12:10] 11_getShipment.js  
    ├── [Jul 16 12:29] 12_history.js  
    ├── [Jul 16 12:31] 13_currentState.js  
    ├── [Jul 16 12:36] 1_addToWallet_manufacturer.js  
    ├── [Jul 16 12:37] 2_addToWallet_distributor.js  
    ├── [Jul 16 12:37] 3_addToWallet_transporter.js  
    ├── [Jul 16 12:37] 4_addToWallet_retailer.js  
    ├── [Jul 16 12:38] 5_addToWallet_consumer.js  
    ├── [Jul 16 15:16] 6_register_companies.js  
    ├── [Jul 16 12:14] 7_addDrug.js  
    ├── [Jul 16 12:18] 8_createdPO.js  
    ├── [Jul 16 16:51] 9_createdShipment.js  
    ├── [Jul 16 12:42] connection-profile-consumer.yaml  
    ├── [Jul 16 12:42] connection-profile-distributor.yaml  
    ├── [Jul 16 12:42] connection-profile-manufacturer.yaml  
    ├── [Jul 16 12:42] connection-profile-retailer.yaml  
    ├── [Jul 16 12:42] connection-profile-transporter.yaml  
    ├── [Jul 16 16:11] contractHelper.js  
    ├── [Jul 18 18:48] createPharmacyDetails.js  
    ├── [Jul 18 18:48] index.js  
    ├── [Jul 16 08:32] package.json  
    └── [Jul 18 08:25] package-lock.json  
[Jul 18 20:51] chaincode  
├── [Jul 18 12:16] contract.js  
├── [Jul 18 12:16] index.js  
└── [Jul 15 22:23] package.json  
[Jul 18 18:49] network  
└── [Nov 22 2021] bta  
    ├── [Jul 15 13:37] configtxgen  
    ├── [Jul 15 13:37] configtxlator  
    ├── [Jul 15 13:37] cryptogen  
    ├── [Jul 15 13:37] dockerca  
    ├── [Jul 15 13:37] fabric-ca-client  
    ├── [Jul 15 13:37] idemixgen  
    ├── [Jul 15 13:37] orderer  
    └── [Jul 15 13:37] peer  
    ├── [Jul 18 10:28] artifacts  
    ├── [Jul 15 13:37] configtx.yaml  
    ├── [Jul 15 13:37] crypto-config.yaml  
    └── [Nov 22 2021] docker-base  
        ├── [Jul 15 13:37] docker-compose-base.yaml  
        ├── [Jul 15 13:37] docker-compose-peer.yaml  
        ├── [Jul 15 13:37] docker-compose-orderer.yaml  
        ├── [Jul 15 13:37] docker-compose-tls.yaml  
        ├── [Jul 15 13:37] docker-compose.yaml  
        └── [Jul 18 10:28] fabricNetwork.sh  
    └── [Nov 22 2021] fabric-samples  
        ├── [Jul 15 13:37] bootstrap.sh  
        ├── [Jul 18 10:28] installChaincode.sh  
        ├── [Jul 15 13:37] updateChaincode.sh  
        └── [Jul 15 13:37] utils.sh  
[Jul 15 00:58] Network Information  
└── [Jul 16 00:58] Project Structure  
    ├── [Oct 5 2020] History.postman_collection.json  
    ├── [Oct 5 2020] Initiation.postman_collection.json  
    └── [Oct 5 2020] Supply Chain.postman_collection.json  
directories, 51 files  
spartans@spartans-Standard-PC-Q35-ICH9-2009: ~/workspace/pharma-net$ ===
```

(fig.1)

As per the above image **(fig.1)**, the top-level directories of project “pharma-net” are:

- **application** — This directory contains different connection profiles for each stakeholder of the organisation (which are manufacturer, distributor, retailer, transporter & consumer and the various functionalities enjoyed by them) , wallet to store the identity of the Admin user of each organisation, node modules corresponding to every function defined in the smart contract and a node server for this application using the ExpressJS library and expose these node modules for application as server-side endpoints.

NOTE: this application is what the first interaction points for the request coming in from Postman and delegates the requests/responses to & from hyperledger fabric network at the backend.

- **chaincode** — This directory contains the smart contracts which is nothing but the **contract.js** file here containing the logic to be supported by the Hyperledger Fabric network for this drug-counterfeiting solution.
- **network** — This directory contains the fabric network setup(s) and the associated files required to raise/install/down/etc. operations for the required hyperledger fabric network as defined by the Network Architecture (click to view the Network Architecture).
- **test** — This directory contains the json files (i.e. json collections) for postman required in order to test the network raised and application deployed.

2. Command to give executable permission to the shell script files.

Below sets of command are required to proceed first. These commands will grant the executable rights/permissions to the shell script and thus can be run further without sudo (i.e. as root user). The execute permissions should be granted to the below files as :

a. Browse first to the project network director as:

```
# cd pharma-net/network/
```

b. Then, provide below commands from terminal one-by-one as:

- i. sudo chmod +x ./fabricNetwork.sh
- ii. sudo chmod +x bin/configtxgen
- iii. sudo chmod +x bin/configtxlator
- iv. sudo chmod +x bin/cryptogen
- v. sudo chmod +x bin/discover
- vi. sudo chmod +x bin/fabric-ca-client
- vii. sudo chmod +x bin/idemixgen
- viii. sudo chmod +x bin/orderer
- ix. sudo chmod +x bin/peer
- x. sudo chmod +x scripts/bootstrap.sh
- xi. sudo chmod +x scripts/installChaincode.sh
- xii. sudo chmod +x scripts/updateChaincode.sh
- xiii. sudo chmod +x scripts/utils.sh

NOTE: refer the attached snapshot below

```
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net$ pwd
/home/spartans/workspace/pharma-net
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net$ cd network/
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ pwd
/home/spartans/workspace/pharma-net/network
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x ./fabricNetwork.sh
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x bin/
configtxgen configtxlator cryptogen discover fabric-ca-client idemixgen orderer peer
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x bin/configtxgen
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x bin/configtxlator
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x bin/cryptogen
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x bin/discover
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x bin/fabric-ca-client
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x bin/idemixgen
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x bin/orderer
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x bin/peer
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x scripts/
bootstrap.sh installChaincode.sh updateChaincode.sh utils.sh
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x scripts/bootstrap.sh
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x scripts/installChaincode.sh
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x scripts/updateChaincode.sh
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ sudo chmod +x scripts/utils.sh
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
```

2. Command to start the hyperledger fabric network

Below sets of command are required to generate crypto material, channel artifacts and network start. Please follow the sequence of steps to complete the network installation as required:

c. **Browse to the project directory as:**

```
# cd pharma-net/network/
```

NOTE: refer the attached snapshot below

```
spartans@spartans-Standard-PC-Q35-IC... x spartans@spartans-Standard-PC-Q35-IC... x spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace$ pwd
/home/spartans/workspace
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace$ cd pharma-net/network/
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 
```

b. Generate crypto-material & channel-artifacts for the network

./fabricNetwork.sh generate

NOTE: refer the attached snapshot below

```
spartans@spartans-Standard-PC-Q35-ICH9-2005:~/workspace/pharma-net/network$ ./fabricNetwork.sh generate
Generating certs and genesis block for channel 'pharmacchannel' with CLI timeout of '100' seconds and CLI delay of '5' seconds and chaincode version '1.0'
Continue? [Y/n] Y
proceeding ...
/home/spartans/workspace/pharma-net/network/bin/cryptogen
#####
##### Generate certificates using cryptogen tool #####
#####
+ cryptogen generate -config=./crypto-config.yaml
manufacturer.pharma-supply-network.com
distributor.pharma-supply-network.com
retailer.pharma-supply-network.com
consumer.pharma-supply-network.com
transporter.pharma-supply-network.com
+ res=0
+ set +x

/home/spartans/workspace/pharma-net/network/bin/configtxgen
#####
##### Generating Orderer Genesis block #####
#####
+ configtxgen -profile OrdererGenesis -channelID upgrad-sys-channel -outputBlock ./channel-artifacts/genesis.block
2023-07-18 21:30:08.618 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2023-07-18 21:30:08.751 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 002 orderer type: solo
2023-07-18 21:30:08.752 IST [common.tools.configtxgen.localconfig] Load -> INFO 003 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:08.866 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 004 orderer type: solo
2023-07-18 21:30:08.866 IST [common.tools.configtxgen.localconfig] LoadTopLevel -> INFO 005 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:08.891 IST [common.tools.configtxgen] doOutputBlock -> INFO 006 Generating genesis block
2023-07-18 21:30:08.893 IST [common.tools.configtxgen] doOutputBlock -> INFO 007 Writing genesis block
+ res=0
+ set +x

#####
## Generating channel configuration transaction 'channel.tx' ##
#####
+ configtxgen -profile PharmaChannel -outputCreateChannelTx ./channel-artifacts/channel.tx -channelID pharmacchannel
2023-07-18 21:30:09.051 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2023-07-18 21:30:09.221 IST [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:09.349 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 003 orderer type: solo
2023-07-18 21:30:09.349 IST [common.tools.configtxgen.localconfig] LoadTopLevel -> INFO 004 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:09.349 IST [common.tools.configtxgen] doOutputChannelCreateTx -> INFO 005 Generating new channel configtx
2023-07-18 21:30:09.350 IST [common.tools.configtxgen.encoder] NewChannelGroup -> WARN 006 Default policy emission is deprecated, please include policy specifications for the channel group in configtx.yaml
2023-07-18 21:30:09.371 IST [common.tools.configtxgen.encoder] NewChannelGroup -> WARN 007 Default policy emission is deprecated, please include policy specifications for the channel group in configtx.yaml
2023-07-18 21:30:09.392 IST [common.tools.configtxgen] doOutputChannelCreateTx -> INFO 008 Writing new channel tx
+ res=0
+ set +x

#####
## Generating anchor peer update for manufacturerMSP #####
#####
+ configtxgen -profile PharmaChannel -outputAnchorPeersUpdate ./channel-artifacts/manufacturerMSPanchors.tx -channelID pharmacchannel -asOrg manufacturerMSP
2023-07-18 21:30:09.585 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2023-07-18 21:30:09.705 IST [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:09.818 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 003 orderer type: solo
2023-07-18 21:30:09.818 IST [common.tools.configtxgen.localconfig] LoadTopLevel -> INFO 004 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:09.818 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 005 Generating anchor peer update
2023-07-18 21:30:09.819 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 006 Writing anchor peer update
+ res=0
+ set +x

#####
## Generating anchor peer update for distributorMSP #####
#####
+ configtxgen -profile PharmaChannel -outputAnchorPeersUpdate ./channel-artifacts/distributorMSPanchors.tx -channelID pharmacchannel -asOrg distributorMSP
2023-07-18 21:30:09.951 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2023-07-18 21:30:10.113 IST [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:10.228 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 003 orderer type: solo
2023-07-18 21:30:10.228 IST [common.tools.configtxgen.localconfig] LoadTopLevel -> INFO 004 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:10.228 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 005 Generating anchor peer update
2023-07-18 21:30:10.231 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 006 Writing anchor peer update
+ res=0
+ set +x

#####
## Generating anchor peer update for retailerMSP #####
#####
+ configtxgen -profile PharmaChannel -outputAnchorPeersUpdate ./channel-artifacts/retailerMSPanchors.tx -channelID pharmacchannel -asOrg retailerMSP
2023-07-18 21:30:10.411 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2023-07-18 21:30:10.530 IST [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:10.641 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 003 orderer type: solo
2023-07-18 21:30:10.641 IST [common.tools.configtxgen.localconfig] LoadTopLevel -> INFO 004 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:10.641 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 005 Generating anchor peer update
2023-07-18 21:30:10.642 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 006 Writing anchor peer update
+ res=0
+ set +x
```

```
#####
##### Generating anchor peer update for transporterMSP #####
#####
+ configtxgen -profile PharmaChannel -outputAnchorPeersUpdate ./channel-artifacts/transporterMPanchors.tx -channelID pharmachannel -asOrg transporterMSP
2023-07-18 21:30:10.826 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2023-07-18 21:30:10.945 IST [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:11.059 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 003 orderer type: solo
2023-07-18 21:30:11.059 IST [common.tools.configtxgen.localconfig] LoadToplevel -> INFO 004 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:11.059 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 005 Generating anchor peer update
2023-07-18 21:30:11.060 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 006 Writing anchor peer update
+ res=0
+ set +x

#####
##### Generating anchor peer update for consumerMSP #####
#####
+ configtxgen -profile PharmaChannel -outputAnchorPeersUpdate ./channel-artifacts/consumerMPanchors.tx -channelID pharmachannel -asOrg consumerMSP
2023-07-18 21:30:11.235 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2023-07-18 21:30:11.353 IST [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:11.468 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 003 orderer type: solo
2023-07-18 21:30:11.468 IST [common.tools.configtxgen.localconfig] LoadToplevel -> INFO 004 Loaded configuration: /home/spartans/workspace/pharma-net/network/configtx.yaml
2023-07-18 21:30:11.468 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 005 Generating anchor peer update
2023-07-18 21:30:11.469 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 006 Writing anchor peer update
+ res=0
+ set +x

spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$
```

C. Create docker container, create and join channel, update anchor peer

./fabricNetwork.sh up

NOTE: refer the attached snapshot below

[!] Running 30/30					
		NAME	COMMAND	CREATED	STATUS
d79756e24b56	hyperledger/fabric-tools:1.4.9	cli	"/bin/bash"	23 seconds ago	Up Less than a second
8d0c1cc5d037	hyperledger/fabric-orderer:1.4.3	orderer.pharma-supply-network.com	"orderer"	23 seconds ago	Up 8 seconds
, :::7050->7050/tcp		ca.manufacturer.pharma-supply-network.com	"sh -c 'fabric-ca-se...' "	24 seconds ago	Exited (1) 9 seconds ago
1e1bbd6a5bfe	hyperledger/fabric-ca:1.4.9	ca.retailer.pharma-supply-network.com	"peer node start --p..."	24 seconds ago	Up 9 seconds
3125c09099be	hyperledger/fabric-peer:1.4.9	peer1.retailer.pharma-supply-network.com	"sh -c 'fabric-ca-se...' "	24 seconds ago	Exited (1) 5 seconds ago
cp, :::16051->16051/tcp, 0.0.0.0:16053->16053/tcp, ::::16053->16053/tcp		ca.transporter.pharma-supply-network.com	"peer node start --p..."	24 seconds ago	Up 3 seconds
c47a3315e832	hyperledger/fabric-ca:1.4.9	peer1.transporter.pharma-supply-network.com	"peer node start --p..."	24 seconds ago	Up 9 seconds
cbd24ca0dd07	hyperledger/fabric-peer:1.4.9	peer0.manufacturer.pharma-supply-network.com	"peer node start --p..."	24 seconds ago	Up 13 seconds
cp, :::11051->11051/tcp, 0.0.0.0:11053->11053/tcp, ::::11053->11053/tcp		peer0.retailer.pharma-supply-network.com	"sh -c 'fabric-ca-se...' "	24 seconds ago	Exited (1) 11 seconds ago
bd3419bd7e53	hyperledger/fabric-ccenv:1.4.9	chaincode	"peer node start --p..."	24 seconds ago	Up 4 seconds
5a35b66a5ebf	hyperledger/fabric-peer:1.4.9	peer0.transporter.pharma-supply-network.com	"peer node start --p..."	24 seconds ago	Up 2 seconds
cp, :::17051->17051/tcp, 0.0.0.0:17053->17053/tcp, ::::17053->17053/tcp		ca.transporter.pharma-supply-network.com	"sh -c 'fabric-ca-se...' "	24 seconds ago	Up 12 seconds
aea30f01e8b8	hyperledger/fabric-ca:1.4.9	peer0.retailer.pharma-supply-network.com	"peer node start --p..."	24 seconds ago	Up 12 seconds
a1d56a2623e6	hyperledger/fabric-peer:1.4.9	peer0.transporter.pharma-supply-network.com	"peer node start --p..."	24 seconds ago	Up 12 seconds
cp, :::15051->15051/tcp, 0.0.0.0:15053->15053/tcp, ::::15053->15053/tcp		peer0.retailer.pharma-supply-network.com	"peer node start --p..."	24 seconds ago	Up 12 seconds
22c715e9c65	hyperledger/fabric-peer:1.4.9	chaincode	"peer node start --p..."	24 seconds ago	Up 12 seconds

```

:::8051->8051/tcp, 0.0.0.0:8053->8053/tcp, :::8053->8053/tcp          peer1.consumer.pharma-supply-network.com
e28aaec7ed1 hyperledger/fabric-peer:1.4.9                                     peer1.manufacturer.pharma-supply-network.com
cd, :::12051->12051/tcp, 0.0.0.0:12053->12053/tcp, :::12053->12053/tcp      ca.distributor.pharma-supply-network.com
259024d86d31 hyperledger/fabric-ca:1.4.9
2b5d0ec1d99 hyperledger/fabric-peer:1.4.9                                     peer0.distributor.pharma-supply-network.com
cp, :::13051->13051/tcp, 0.0.0.0:13053->13053/tcp, :::13053->13053/tcp      peer1.transporter.pharma-supply-network.com
2ecc5f6b845 hyperledger/fabric-peer:1.4.9                                     ca.consumer.pharma-supply-network.com
cp, :::18051->18051/tcp, 0.0.0.0:18053->18053/tcp, :::18053->18053/tcp      peer1.consumer.pharma-supply-network.com
5449ae829efb hyperledger/fabric-ca:1.4.9
Sac9080274c1 hyperledger/fabric-peer:1.4.9                                     peer0.consumer.pharma-supply-network.com
, :::7051->7051/tcp, 0.0.0.0:7053->7053/tcp, :::7053->7053/tcp      peer1.distributor.pharma-supply-network.com
e6167fcc273a hyperledger/fabric-peer:1.4.9                                     dev-peer0.consumer.pharma-supply-network.com-pharmanet:1.0-e827da1aff808882bc3ad999e0ff151a9807862195ceca9d5b687105fb87c7
2ecc5f6b845 hyperledger/fabric-peer:1.4.9                                     dev-peer0.consumer.pharma-supply-network.com-pharmanet:1.0-e827da1aff808882bc3ad999e0ff151a9807862195ceca9d5b687105fb87c7
34857e33c5d1 dev-peer0.consumer.pharma-supply-network.com-pharmanet:1.0-e827da1aff808882bc3ad999e0ff151a9807862195ceca9d5b687105fb87c7
a6ec1847cb80 dev-peer0.retailer.pharma-supply-network.com-pharmanet:1.0-ce981c948c4ba2a63801639c899e4801e1de9b4a7ef25f4e2d32633daed2
a5dc2c6b482a0 dev-peer0.transporter.pharma-supply-network.com-pharmanet:1.0-ce981c948c4ba2a63801639c899e4801e1de9b4a7ef25f4e2d32633daed2
30074062d925 dev-peer0.distributor.pharma-supply-network.com-pharmanet:1.0-ce981c948c4ba2a63801639c899e4801e1de9b4a7ef25f4e2d32633daed2
8116b588f275 dev-peer0.manufacturer.pharma-supply-network.com-pharmanet:1.0-7944c12914fbfe1171b486360051d3da6b6cc3f4b513359f3a8e2f7164f7fd5f7
b446da514ea hyperledger/fabric-ccenv:1.4                                     dev-peer0.manufacturer.pharma-supply-network.com-pharmanet:1.0-7944c12914fbfe1171b486360051d3da6b6cc3f4b513359f3a8e2f7164f7fd5f7
Sleeping 10s to allow cluster to complete booting
                                                adoring_gauss

Setting Up Hyperledger Fabric Network

Channel name : pharmachannel
Creating channel...
+ peer channel create -o orderer.pharma-supply-network.com:7050 -c pharmachannel -f ./channel-artifacts/channel.tx
+ res=0
+ set +x
2023-07-19 02:03:18.760 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:03:19.094 UTC [cli_common] ReadBlock -> INFO 002 Received block: 0
===== Channel 'pharmachannel' created =====

Having all peers join the channel...
+ peer channel join -b pharmachannel.block
+ res=0
+ set +x
2023-07-19 02:03:19.339 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:03:19.534 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer0.consumer.pharma-supply-network.com joined channel 'pharmachannel' =====

+ peer channel join -b pharmachannel.block
+ res=0
+ set +x
2023-07-19 02:03:24.889 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:03:25.053 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer0.consumer.pharma-supply-network.com joined channel 'pharmachannel' =====

+ peer channel join -b pharmachannel.block
+ res=0
+ set +x
2023-07-19 02:03:30.419 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:03:30.576 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer0.distributor.pharma-supply-network.com joined channel 'pharmachannel' =====

+ peer channel join -b pharmachannel.block
+ res=0
+ set +x
2023-07-19 02:03:35.945 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:03:36.094 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer0.distributor.pharma-supply-network.com joined channel 'pharmachannel' =====

+ peer channel join -b pharmachannel.block
+ res=0
+ set +x
2023-07-19 02:03:41.445 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:03:41.601 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer0.manufacturer.pharma-supply-network.com joined channel 'pharmachannel' =====

+ peer channel join -b pharmachannel.block
+ res=0
+ set +x
2023-07-19 02:03:46.936 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:03:47.088 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer0.manufacturer.pharma-supply-network.com joined channel 'pharmachannel' =====

+ peer channel join -b pharmachannel.block
+ res=0
+ set +x
2023-07-19 02:03:52.438 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:03:52.599 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer0.transporter.pharma-supply-network.com joined channel 'pharmachannel' =====

+ peer channel join -b pharmachannel.block
+ res=0
+ set +x
2023-07-19 02:03:57.952 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:03:58.105 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer1.retailer.pharma-supply-network.com joined channel 'pharmachannel' =====

+ peer channel join -b pharmachannel.block
+ res=0
+ set +x
2023-07-19 02:04:03.465 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:04:03.621 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer0.transporter.pharma-supply-network.com joined channel 'pharmachannel' =====

+ peer channel join -b pharmachannel.block
+ res=0
+ set +x
2023-07-19 02:04:08.953 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:04:09.183 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer1.transporter.pharma-supply-network.com joined channel 'pharmachannel' =====

Updating anchor peers for CONSUMER...
+ peer channel update -o orderer.pharma-supply-network.com:7050 -c pharmachannel -f ./channel-artifacts/consumerMSPanchor.tx
+ res=0
+ set +x
2023-07-19 02:04:14.452 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:04:14.551 UTC [channelCmd] update -> INFO 002 Successfully submitted channel update
===== Anchor peers updated for org 'consumerMSP' on channel 'pharmachannel' =====

```

```

Updating anchor peers for MANUFACTURER...
+ peer channel update -o orderer.pharma-supply-network.com:7050 -c pharmachannel -f ./channel-artifacts/manufacturerMSPanchors.tx
+ res=0
+ set +x
2023-07-19 02:04:19.887 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:04:19.981 UTC [channelCmd] update -> INFO 001 Successfully submitted channel update
=====
Anchor peers updated for org 'manufacturerMSP' on channel 'pharmachannel' =====

Updating anchor peers for DISTRIBUTOR...
+ peer channel update -o orderer.pharma-supply-network.com:7050 -c pharmachannel -f ./channel-artifacts/distributorMSPanchors.tx
+ res=0
+ set +x
2023-07-19 02:04:25.373 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:04:25.463 UTC [channelCmd] update -> INFO 001 Successfully submitted channel update
=====
Anchor peers updated for org 'distributorMSP' on channel 'pharmachannel' =====

Updating anchor peers for RETAILER...
+ peer channel update -o orderer.pharma-supply-network.com:7050 -c pharmachannel -f ./channel-artifacts/retailerMSPanchors.tx
+ res=0
+ set +x
2023-07-19 02:04:30.839 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2023-07-19 02:04:30.936 UTC [channelCmd] update -> INFO 001 Successfully submitted channel update
=====
Anchor peers updated for org 'retailerMSP' on channel 'pharmachannel' =====

=====
All GOOD, Hyperledger Fabric Pharma Network Is Now Up and Running! =====

```

END

spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network\$

3. Command to connect to the chaincode container and run the chaincode.

NOTE: the below given command can be used to exec into the chaincode and run it.

This is not necessary to be executed as we can skip this process and directly go into the Point no. (4). All these process of chaincode instantiate & run is taken care off already in the Point no. (4) as automated install & instantiate process.

I. Command to exec into container shell

docker exec -it chaincode /bin/bash

NOTE:

- the present working directory is the project directory (**\$PWD**) which is “pharma-net”.**
- With above command, we will be accessing the Container Shell using docker exec command**

Refer the attached snapshot below

```
spartans@spartans-Standard-PC-035-ICH9-2009:~/workspace/pharma-net$ docker exec -it chaincode /bin/bash
root@bd3419bde753:/opt/gopath/src/chaincode# ls
contract.js index.js package.json
root@bd3419bde753:/opt/gopath/src/chaincode#
root@bd3419bde753:/opt/gopath/src/chaincode#
root@bd3419bde753:/opt/gopath/src/chaincode#
root@bd3419bde753:/opt/gopath/src/chaincode# npm install
npm WARN deprecated grpc@1.17.0: This library will not receive further updates other than security fixes. We recommend using @grpc/grpc-js instead.
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
npm WARN deprecated node-pre-gyp@0.12.0: Please upgrade to @mapbox/node-pre-gyp: the non-scoped node-pre-gyp package is deprecated and only the @mapbox scoped package will receive updates in the future
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated uidv3@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.
```

II. Command to install the dependencies as defined in the package.json.

NOTE: to check package.json, refer path as “pharma-net/chaincode”

```
root@bd3419bde753:/opt/gopath/src/chaincode# npm install
npm WARN deprecated grpc@1.17.0: This library will not receive further updates other than security fixes. We recommend using @grpc/grpc-js instead.
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
npm WARN deprecated node-pre-gyp@0.12.0: Please upgrade to @mapbox/node-pre-gyp: the non-scoped node-pre-gyp package is deprecated and only the @mapbox scoped package will receive updates in the future
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated uidv3@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.
```

III. Command to run the chaincode.

npm run start-manufacturer

Refer the attached snapshot below

```
root@bd3419bde753:/opt/gopath/src/chaincode# npm run start-manufacturer
> pharmanet@1.0.0 start-manufacturer /opt/gopath/src/chaincode
> fabric-chaincode-node start --peer.address peer0.manufacturer.pharma-supply-network.com:11053 --chaincode-id-name pharmanet:1.1
2023-07-19T02:29:27.628Z info [shim:contracts-spi/bootstrap.js] No metadata file supplied in contract, introspection will generate all the data
2023-07-19T02:29:27.638Z info [shim:contracts-spi/chaincodefromcontract.js] Using serializers {"transaction":"jsonSerializer","serializers":{}}
2023-07-19T02:29:27.862Z info [shim:lib/chaincode.js] Registering with peer peer0.manufacturer.pharma-supply-network.com:11053 as chaincode "pharmanet:1.1"
Command succeeded
2023-07-19T02:29:27.937Z info [shim:lib/handler.js] Successfully registered with peer node. State transferred to "established"
2023-07-19T02:29:27.942Z info [shim:lib/handler.js] Successfully established communication with peer node. State transferred to "ready"
```

III. Command to gracefully exit the container shell, without stopping it.

Press “Ctrl + p” & followed by “Ctrl + q” over keyboard

IV. Similarly we can run the chaincode using below command as well. Note all these are defined inside package.json file only. Refer the below snapshot to see the definition inside package.json's "scripts" section.

```
"scripts": {
  "lint": "eslint .",
  "pretest": "npm run lint",
  "test": "nyc mocha test --recursive",
  "start-manufacturer": "fabric-chaincode-node start --peer.address peer0.manufacturer.pharma-supply-network.com:11053 --chaincode-id-name pharmanet:1.1",
  "start-distributor": "fabric-chaincode-node start --peer.address peer0.distributor.pharma-supply-network.com:9053 --chaincode-id-name pharmanet:1.1",
  "start-retailer": "fabric-chaincode-node start --peer.address peer0.retailer.pharma-supply-network.com:15053 --chaincode-id-name pharmanet:1.1",
  "start-transport": "fabric-chaincode-node start --peer.address peer0.transporter.pharma-supply-network.com:17053 --chaincode-id-name pharmanet:1.1",
  "start": "fabric-chaincode-node start",
  "mocha": "mocha test --recursive"
},
```

NOTE:

- a. As per above snapshot attached, we can run the chaincode likewise below commands as well

#> npm run start-distributor

Or

#> npm run start-retailer

Or

#> npm run start-transport

- b. Also take a note that every time we start the container this way, we required to gracefully exit the container using "ctrl + p" followed by "ctrl + q", otherwise the container will stop.



TIP: refer below snippet from the package.json inside /parma-net/chaincode directory.

```
package.json - parma-net - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
PHARMA-NET
  > application
  > chaincode
    > node_modules
    > contract.js
    > index.js
    > package-lock.json
    > package.json
  > chaincode-advanced
  > network
  > test
  > Network Information
  > Project Structure
package.json
  1  {
  2    "name": "pharmanet",
  3    "version": "1.0.0",
  4    "description": "Pharma Network : A Distributed Drug-Counterfeit Network",
  5    "main": "index.js",
  6    "engines": {
  7      "node": ">=8",
  8      "npm": ">=5"
  9    }
  10   "scripts": {
  11     "lint": "eslint .",
  12     "pretest": "npm run lint",
  13     "test": "nyc mocha test --recursive",
  14     "start-manufacturer": "fabric-chaincode-node start --peer.address peer0.manufacturer.pharma-supply-network.com:11053 --chaincode-id-name pharmanet:1.1",
  15     "start-distributor": "fabric-chaincode-node start --peer.address peer0.distributor.pharma-supply-network.com:9053 --chaincode-id-name pharmanet:1.1",
  16     "start-retailer": "fabric-chaincode-node start --peer.address peer0.retailer.pharma-supply-network.com:15053 --chaincode-id-name pharmanet:1.1",
  17     "start-transport": "fabric-chaincode-node start --peer.address peer0.transporter.pharma-supply-network.com:17053 --chaincode-id-name pharmanet:1.1",
  18     "start": "fabric-chaincode-node start",
  19     "mocha": "mocha test --recursive"
  20   }
  21   "keywords": [
  22     "Blockchain",
  23     "Hyperledger",
  24     "Fabric",
  25     "Distributed",
  26     "Application",
  27     "NodeJS",
  28     "Drug Counterfeit"
  29   ],
  30   "engineStrict": true,
  31   "author": "Saurav Savarn",
  32   "license": "ISC",
  33   "dependencies": {
  34     "fabric-contract-api": "1.4.2",
  35     "fabric-shim": "1.4.2",
  36   }
}
OUTLINE
TIMELINE
```

4. Command to install and instantiate chaincode.

Use the below command to install and instantiate chaincode at one go:

a. **Ensure you are into project directory as below:**

```
# cd pharma-net/network/
```

b. **Issue below commands to install and instantiate the chaincode as**

```
# ./fabricNetwork.sh install
```

NOTE: refer the attached snapshot below

```
spartans@spartans-Standard-PC-035-ICH9-2009:~/workspace/pharma-net/network$ ./FabricNetwork.sh install
Installing chaincode for channel 'pharmachannel' with CLI timeout of '100' seconds and CLI delay of '5' seconds and chaincode version '1.0'
Continue? [Y/n] Y
proceeding ...
LOCAL_VERSION=1.4.2
DOCKER_IMAGE_VERSION=v2.5.3
=====
WARNING =====
Local fabric binaries and docker images are
out of sync. This may cause problems.
=====

START

Deploying Chaincode pharmenet On Pharma Network
Channel name : pharmachannel
Installing chaincode on peer0.manufacturer.pharma-supply-network.com ...
+ peer chaincode install -n pharmenet -v 1.0 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
+ res=0
+ set +x
2023-07-19 03:13:52.339 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 001 Using default escc
2023-07-19 03:13:52.340 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 002 Using default vscc
2023-07-19 03:13:53.040 UTC [chaincodeCmd] install > INFO 003 Installed remotely response:<status:200 payload:"OK" >
=====
Chaincode is installed on peer0.manufacturer =====

Installing chaincode on peer0.distributor.pharma-supply-network.com ...
+ peer chaincode install -n pharmenet -v 1.0 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
+ res=0
+ set +x
2023-07-19 03:13:53.476 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 001 Using default escc
2023-07-19 03:13:53.476 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 002 Using default vscc
2023-07-19 03:13:54.093 UTC [chaincodeCmd] install > INFO 003 Installed remotely response:<status:200 payload:"OK" >
=====
Chaincode is installed on peer0.distributor =====

Installing chaincode on peer0.retailer.pharma-supply-network.com ...
+ peer chaincode install -n pharmenet -v 1.0 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
+ res=0
+ set +x
2023-07-19 03:13:54.499 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 001 Using default escc
2023-07-19 03:13:54.500 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 002 Using default vscc
2023-07-19 03:13:55.054 UTC [chaincodeCmd] install > INFO 003 Installed remotely response:<status:200 payload:"OK" >
=====
Chaincode is installed on peer0.retailer =====

Installing chaincode on peer0.transporter.pharma-supply-network.com ...
+ peer chaincode install -n pharmenet -v 1.0 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
+ res=0
+ set +x
2023-07-19 03:13:55.456 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 001 Using default escc
2023-07-19 03:13:55.457 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 002 Using default vscc
2023-07-19 03:13:56.220 UTC [chaincodeCmd] install > INFO 003 Installed remotely response:<status:200 payload:"OK" >
=====
Chaincode is installed on peer0.transporter =====
```

```

Installing chaincode on peer0.consumer.pharma-supply-network.com ...
+ peer chaincode install -n pharmanet -v 1.0 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
+ res=0
+ set +x
2023-07-19 03:13:56.680 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 001 Using default escc
2023-07-19 03:13:56.683 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 002 Using default vscc
2023-07-19 03:13:57.446 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 003 Installed remotely response:<status:200 payload:"OK" >
=====
Chaincode is installed on peer0.consumer =====

Instantiating chaincode on channel using peer0.manufacturer.pharma-supply-network.com ...
+ peer chaincode instantiate -o orderer.pharma-supply-network.com:7050 -C pharnachannel -n pharmanet -l node -v 1.0 -c '{"Args":["org.pharma-network.pharmanet:instantiate"]}' -P 'OR ('\\''manufacturerMSP.member'\\'',\\''distributorMSP.member'\\'',\\''retailerMSP.member'\\'',\\''transporterMSP.member'\\'',\\''consumerMSP.member'\\'')'
+ res=0
+ set +x
2023-07-19 03:13:57.689 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 001 Using default escc
2023-07-19 03:13:57.690 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 002 Using default vscc
=====
Chaincode is instantiated on peer0.manufacturer on channel 'pharnachannel' =====

Instantiating chaincode on channel using peer0.distributor.pharma-supply-network.com ...
+ peer chaincode instantiate -o orderer.pharma-supply-network.com:7050 -C pharnachannel -n pharmanet -l node -v 1.0 -c '{"Args":["org.pharma-network.pharmanet:instantiate"]}' -P 'OR ('\\''manufacturerMSP.member'\\'',\\''distributorMSP.member'\\'',\\''retailerMSP.member'\\'',\\''transporterMSP.member'\\'',\\''consumerMSP.member'\\'')'
+ res=0
+ set +x
2023-07-19 03:14:06.993 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 001 Using default escc
2023-07-19 03:14:06.993 UTC [chaincodeCmd] checkChaincodeCmdParams > INFO 002 Using default vscc
=====
Chaincode is instantiated on peer0.distributor on channel 'pharnachannel' =====

=====
All GOOD, Chaincode PHARANET Is Now Installed & Instantiated On Pharmanet Network =====

```

END

spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network\$ █

NOTE: Once the chaincode install & instantiate command completes, if we check the docker running containers & docker volumes mapped to those running containers, using command “docker ps” & “docker volume ls” respectively, we will see as per below snapshot.

```

spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ docker volume ls
DRIVER VOLUME NAME
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
RTS
b5bf42910455 dev-peer0.distributor.pharma-supply-network.com-pharmanet-1.0-74764253d704b5a086c4fdeb3b762c3ce918cf68b9f708ac9798ca5e7df6c20b "bin/sh -c 'cd /usr..." 6 minutes ago Up 6 minutes
dev-peer0.distributor.pharma-supply-network.com-pharmanet-1.0
7346777fa65a dev-peer0.manufacturer.pharma-supply-network.com-pharmanet-1.0-7944c12914fbe1171b486360051d3daeb0cc3f4b513359f3a8e2f/c04f7fd5f7 "/bin/sh -c 'cd /usr..." 7 minutes ago Up 7 minutes
dev-peer0.manufacturer.pharma-supply-network.com-pharmanet-1.0
90bab33b906 hyperledger/fabric-tools:1.4.9
cli
f3e650793dd4 hyperledger/fabric-peer:1.4.9
0.0.0.17051->17051/tcp, ::17051->17053/tcp, ::17053->17053/tcp
d0242aa6f911 hyperledger/fabric-peer:1.4.9
0.0.0.18051->18051/tcp, ::18051->18051/tcp, 0.0.0.0:18053->18053/tcp, ::18053->18053/tcp
b6c513beecfc hyperledger/fabric-cncenv:1.4.9
c0262b5190a9 hyperledger/fabric-peer:1.4.9
0.0.0.15051->15051/tcp, ::15051->15053/tcp, 0.0.0.0:15053->15053/tcp, ::15053->15053/tcp
6ba775d2bf0f hyperledger/fabric-peer:1.4.9
0.0.0.14051->14051/tcp, ::14051->14053/tcp, 0.0.0.0:14053->14053/tcp
af1f819804d hyperledger/fabric-peer:1.4.9
0.0.0.12051->12051/tcp, ::12051->12051/tcp, 0.0.0.0:12053->12053/tcp, ::12053->12053/tcp
995a244ecb9 hyperledger/fabric-peer:1.4.9
0.0.0.16051->16051/tcp, ::16051->16051/tcp, 0.0.0.0:16053->16053/tcp, ::16053->16053/tcp
7564ffbe3f1 hyperledger/fabric-peer:1.4.9
0.0.0.13051->13051/tcp, ::13051->13053/tcp, 0.0.0.0:13053->13053/tcp
7eb65eb1fd75 hyperledger/fabric-peer:1.4.9
0.0.0.11051->11051/tcp, ::11051->11053/tcp, 0.0.0.0:11053->11053/tcp
6a9e64a31c8b hyperledger/fabric-peer:1.4.9
0.0.0.7051->7051/tcp, ::7051->7053/tcp, 0.0.0.0:7053->7053/tcp
d47d523817bd hyperledger/fabric-orderer:1.4.3
0.0.0.7050->7050/tcp, ::7050->7050/tcp
37951d36782 hyperledger/fabric-peer:1.4.9
0.0.0.8051->8051/tcp, ::8051->8053/tcp, 0.0.0.0:8053->8053/tcp
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ docker volume ls
DRIVER VOLUME NAME
local 035b6ea382f253a094c2dddb95a77a603a297880ab69e97807958f78f39b22e
local network_orderer.pharma-supply-network.com
local network_peer0.consumer.pharma-supply-network.com
local network_peer0.distributor.pharma-supply-network.com
local network_peer0.manufacturer.pharma-supply-network.com
local network_peer0.retailer.pharma-supply-network.com
local network_peer0.transporter.pharma-supply-network.com
local network_peer1.consumer.pharma-supply-network.com
local network_peer1.distributor.pharma-supply-network.com
local network_peer1.manufacturer.pharma-supply-network.com
local network_peer1.retailer.pharma-supply-network.com
local network_peer1.transporter.pharma-supply-network.com
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ █

```

5. Commands to start Node application

To start the backend nodejs application sequence of steps is required to be followed. Also it is need to ensure the directory we are in. Please follow below steps to start the nodejs application:

a. **Ensure you are into project's application directory as below:**

```
# cd pharma-net/application/
```

NOTE: refer the attached snapshot below. As per the snapshot attached here, if we list the directory using command “ls”, we can see list of files inside the “application” directory.

```
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace$ cd pharma-net/application/
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$ ls
10_updateShipment.js      2_addToWallet_distributor.js    7_addDrug.js          connection-profile-manufacturer.yaml index.js
11_retailDrug.js          3_addToWallet_transporter.js  8_createPO.js        connection-profile-retailer.yaml package.json
12_history.js              4_addToWallet_retailer.js       9_createsshipment.js  connection-profile-transporter.yaml package-lock.json
13_currentState.js         5_addToWallet_consumer.js     connection-profile-consumer.yaml contractHelper.js
1_addToWallet_manufacturer.js 6_register_companies.js   connection-profile-distributor.yaml getCompanyDetails.js
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$
```

b. **Install the node-modules which are required, using the below command:**

```
# npm install
```

NOTE: refer the attached snapshot below to see the process of node dependencies install.

```
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$ npm install
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated cloudant-follow@0.17.0: This package is no longer maintained.
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
npm WARN deprecated core-js@2.6.12: core-js@<3.23.3 is no longer maintained and not recommended for usage due to the number of issues. Because of the V8 engine whims, feature detection in old core-js versions could cause a slowdown up to 100x even if nothing is polyfilled. Some versions have web compatibility issues. Please, upgrade your dependencies to the actual version of core-js.
npm WARN deprecated grpc@1.24.11: This library will not receive further updates other than security fixes. We recommend using @grpc/grpc-js instead.

added 428 packages, and audited 429 packages in 3m

26 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (7 moderate, 1 high)

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$
```

c. **Update the keystores to make sure Identity can be successfully stored in the Wallet, using the below command:**

npm run identityAddAll

NOTE: refer the attached snapshot below to see the process of Identity installations to the Wallet.

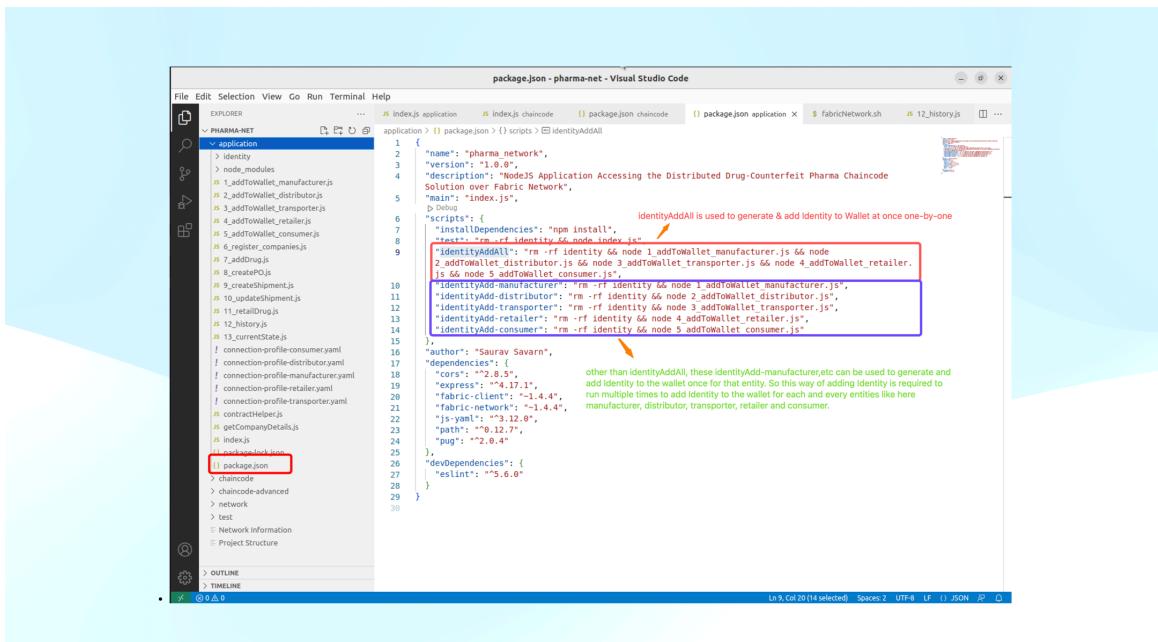
This is very much important otherwise the request will not be identified as valid request and will be rejected by the MSP.

```
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$ npm run identityAddAll
> pharma_network@1.0.0 identityAddAll
> rm -rf identity && node 1_addToWallet_manufacturer.js && node 2_addToWallet_distributor.js && node 3_addToWallet_transporter.js && node 4_addToWallet_retailer.js && node 5_addToWallet_consumer.js

Manufacturer identity added to wallet.
Distributor identity added to wallet.
Transporter identity added to wallet.
retailer identity added to wallet.
consumer identity added to wallet.
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$
```



TIP: refer below snippet from the package.json inside /parma-net/application directory. This is the package.json files for the application requirement. Please check the inline text added to the snapshot while will detailed about the use of each type of identityAdd** as defined within scripts in package.json.



- d. **Start the nodejs application using below command:**

```
# node .
```

NOTE: refer the attached snapshot below to see the the output of command “node .”, which is going to start the nodejs application. This will print “**Distributed Drug-Counterfeit Pharma Application listening on port 3000!**”, which indicates that application have been started successfully.

```
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$ pwd
/home/spartans/workspace/pharma-net/application
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$ 
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/application$ node .
Distributed Drug-Counterfeit Pharma Application listening on port 3000!
```

6. Using postman to run test collection.

Below steps to be followed to start testing the application.

- a. **Launch the Postman Application**
- b. **Select and import the postman collection saved into “test” directory. This can be done in 2 ways:**
 - i. **Import all the sample at once.** Here entire postman collection is shared into directory “test” as file name “**post-all-collection.json**”
 - ii. **Import the collection one by one from collection saved into “test” folder having directory within by name as ‘Initiation.postman_collection.json’, ‘Supply Chain.postman_collection.json’ & ‘View History.postman_collection.json’.**

c. Test Case - 1: Initiation :

When imported postman collection, we will get a folder name '**Initiation**' , containing the HTTP requests to perform the following functions as:

PharmaNet: Drug-Counterfeit-Distributed-Application

- Collection-1
 - Initiation**
 - POST Sun Pharma
 - POST FedEx
 - POST Blue Dart
 - POST VG Pharma
 - POST upgrad
 - POST Paracetamol-001
 - POST Paracetamol-002
 - POST Paracetamol-003
 - POST Paracetamol-004
 - Supply Chain
 - View History

i. Create Manufacturer with the following details :

Name: 'Sun Pharma'

CRN number: 'MAN001'

Location: 'Chennai'

POST Sun Pharma

POST Sun Pharma

Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/> organizationName	Manufacturer			
<input checked="" type="checkbox"/> companyCRN	MAN001			
<input checked="" type="checkbox"/> companyName	Sun Pharma			
<input checked="" type="checkbox"/> location	Chennai			
<input checked="" type="checkbox"/> organisationRole	Manufacturer			
Key	Value	Description		

Status: 200 OK Time: 7.65 s Size: 567 B Save as Example

```

1
2
3
4
5
6
7
8
9
10
11
12
  "status": "success",
  "message": "New Company got registered",
  "newCompany": {
    "companyID": "Org.pharma-network.pharmanet.company@MAN001@Sun Pharma@",
    "name": "Sun Pharma",
    "location": "Chennai",
    "organisationRole": "Manufacturer",
    "hierarchyKey": "1",
    "createdAt": "2023-07-19T13:44:48.029Z"
  }
}

```

ii. Create a transporter with the following details:

Name: 'FedEx'

CRN number: 'TRA001'

Location: 'Delhi'

The screenshot shows the Postman application interface. On the left, the 'Personal' collections tree is visible, with 'POST FedEx' selected under 'Collection-1 / Initiation'. The main workspace displays a POST request to 'localhost:3000/registerCompany'. The 'Body' tab is active, showing the following JSON payload:

```

{
  "organizationName": "Transporter",
  "companyCRN": "TRA001",
  "companyName": "FedEx",
  "location": "Delhi",
  "organisationRole": "Transporter"
}

```

The response status is 200 OK, time taken is 17.38 s, and the size is 553 B. The response body is:

```

{
  "status": "success",
  "message": "New Company got registered",
  "newCompany": {
    "companyID": "org.pharma-network.pharmanet.company#TRA001#FedEx",
    "name": "FedEx",
    "location": "Delhi",
    "organisationRole": "Transporter",
    "hierarchyKey": "",
    "createdAt": "2023-07-19T13:48:17.206Z"
  }
}

```

iii. Create another transporter with the following details:

Name: 'Blue Dart'

CRN number: 'TRA002'

Location: 'Bangalore'

The screenshot shows the Postman application interface. On the left, the sidebar has sections for Personal, Collections, Environments, and History. Under 'Collections', there is a tree view with 'History', 'mssc-teaplantation', 'PharmaNet: Drug-Counterfeit-Distributed-Application' (expanded), 'Collection-1' (under it), and 'Initiation' (under Collection-1). Under 'Initiation', several POST requests are listed: Sun Pharma, FedEx, Blue Dart (selected and highlighted with a red arrow), VG Pharma, upgrad, Paracetamol-001, Paracetamol-002, Paracetamol-003, and Paracetamol-004. The 'Body' tab is selected in the main panel, showing a table with fields: organizationName (Value: Transporter), companyCRN (Value: TRA002), companyName (Value: Blue Dart), location (Value: Bangalore), and organisationRole (Value: Transporter). Below the table, the 'Pretty' tab of the JSON editor is selected, displaying the following JSON response:

```
1 "status": "success",
2 "message": "New Company got registered",
3 "newCompany": {
4     "companyID": "org.pharma-network.phairmanet.company@TRA002@Blue Dart@",
5     "name": "Blue Dart",
6     "location": "Bangalore",
7     "organisationRole": "Transporter",
8     "hierarchyKey": "",
9     "createdAt": "2023-07-19T13:50:21.660Z"
10 }
11 }
```

iv. Create a distributor with the following details:

Name: 'VG Pharma'

CRN number: 'DIST001'

Location: 'Vizag'

The screenshot shows the Postman interface with the following details:

- Collection:** Personal
- Environment:** No Environment
- Request:**
 - Method:** POST
 - URL:** localhost:3000/registerCompany
 - Body (x-www-form-urlencoded):**

Key	Value	Description
organizationName	Distributor	
companyCRN	DIST001	
companyName	VG Pharma	
location	Vizag	
organisationRole	Distributor	
- Response:**
 - Status: 200 OK
 - Time: 2.98 s
 - Size: 563 B
 - Pretty JSON:

```

1   "status": "success",
2   "message": "New Company got registered",
3   "newCompany": {
4     "companyID": "Org.pharma-network.phaznet.company0DIST001VG Pharma",
5     "name": "VG Pharma",
6     "location": "Vizag",
7     "organisationRole": "Distributor",
8     "hierarchyKey": "2",
9     "createdAt": "2023-07-19T13:51:57.696Z"
10   }
11 }
12 
```

v. Create a retailer with the following details:

Name: 'upgrad'

CRN number: 'RET002'

Location: 'Mumbai'

The screenshot shows the Postman application interface. On the left, the 'Personal' collection is selected, displaying various API endpoints. A red arrow points to the 'POST upgrad' endpoint under the 'Initiation' section. The main right panel shows a POST request to 'localhost:3000/registerCompany'. The 'Body' tab is selected, showing a JSON payload with the following fields:

Key	Value	Description
organizationName	Retailer	
companyCRN	RET002	
companyName	upgrad	
location	Mumbai	
organisationRole	Retailer	

Below the table, the response body is displayed in pretty JSON format:

```
1
2   "status": "success",
3   "message": "New Company got registered",
4   "newCompany": {
5     "companyID": "Org.pharma-network.pharmenet.companyRET002upgrad",
6     "name": "upgrad",
7     "location": "Mumbai",
8     "organisationRole": "Retailer",
9     "hierarchyKey": "3",
10    "createdAt": "2023-07-19T13:53:50.718Z"
11  }
12 }
```

vi. Create 4 strips of a drug named ‘Paracetamol’ with serial number starting from ‘001’ to ‘004’ .

- Strip 1 of drug named “Paracetamol” created.

The screenshot shows the Postman interface with the following details:

- Left Panel (Collections):** Shows a tree structure with 'Personal' at the top, followed by 'Collections', 'Environments', and 'History'. Under 'PharmaNet: Drug-Counterfeit-Distributed-Application' > 'Collection-1' > 'Initiation', there are several POST requests: 'POST Sun Pharma', 'POST FedEx', 'POST Blue Dart', 'POST VG Pharma', 'POST upgrad', 'POST Paracetamol-001' (highlighted with a red arrow), 'POST Paracetamol-002', 'POST Paracetamol-003', and 'POST Paracetamol-004'.
- Request Details:** The selected request is 'POST Paracetamol-001' to 'localhost:3000/addDrug'. The 'Body' tab is selected, showing the following JSON payload:

```

{
  "organizationName": "Manufacturer",
  "drugName": "Paracetamol",
  "serialNo": "001",
  "mfgDate": "2023-07-15",
  "expDate": "2023-07-15",
  "companyCRN": "MAN001"
}

```

- Response:** The response status is 200 OK, with a response time of 2.85 s and a size of 698 B. The response body is:

```

{
  "status": "success",
  "message": "New Drug got registered",
  "newDrug": {
    "productID": "org.pharma-network.pharmanet.drug#Paracetamol#001",
    "name": "Paracetamol",
    "manufacturer": "org.pharma-network.pharmanet.company#MAN001#Sun Pharma",
    "manufacturingDate": "2023-07-15",
    "expiryDate": "2023-07-15",
    "owner": "org.pharma-network.pharmanet.company#MAN001#Sun Pharma",
    "shipment": ""
  }
}

```

- Strip 2 of drug named “Paracetamol” created.

The screenshot shows the Postman interface with the following details:

- Left Panel (Collections):** Shows a tree structure with 'Personal' at the top, followed by 'Collections', 'Environments', and 'History'. Under 'PharmaNet: Drug-Counterfeit-Distributed-Application' > 'Collection-1' > 'Initiation', there are several POST requests: 'POST Sun Pharma', 'POST FedEx', 'POST Blue Dart', 'POST VG Pharma', 'POST upgrad', 'POST Paracetamol-001', 'POST Paracetamol-002' (highlighted with a red arrow), 'POST Paracetamol-003', and 'POST Paracetamol-004'.
- Request Details:** The selected request is 'POST Paracetamol-002' to 'localhost:3000/addDrug'. The 'Body' tab is selected, showing the following JSON payload:

```

{
  "organizationName": "Manufacturer",
  "drugName": "Paracetamol",
  "serialNo": "002",
  "mfgDate": "2023-07-15",
  "expDate": "2023-07-15",
  "companyCRN": "MAN001"
}

```

- Response:** The response status is 200 OK, with a response time of 2.86 s and a size of 698 B. The response body is:

```

{
  "status": "success",
  "message": "New Drug got registered",
  "newDrug": {
    "productID": "org.pharma-network.pharmanet.drug#Paracetamol#002",
    "name": "Paracetamol",
    "manufacturer": "org.pharma-network.pharmanet.company#MAN001#Sun Pharma",
    "manufacturingDate": "2023-07-15",
    "expiryDate": "2023-07-15",
    "owner": "org.pharma-network.pharmanet.company#MAN001#Sun Pharma",
    "shipment": ""
  }
}

```

- Strip 3 of drug named “Paracetamol” created.

The screenshot shows the Postman interface with the following details:

- Left Sidebar:** Personal collections, environments, and history.
- Current Collection:** PharmaNet: Drug-Counterfeit-Distributed-Application / Collection-1 / Initiation.
- Request:** POST Paracetamol-003 to localhost:3000/addDrug.
- Body:** x-www-form-urlencoded (selected).

Key	Value	Description
organizationName	Manufacturer	
drugName	Paracetamol	
serialNo	003	
mfgDate	2023-07-15	
expDate	2023-07-15	
companyCRN	MAN001	
- Response:**

```

1
2 "status": "success",
3 "message": "New Drug got registered",
4 "newDrug": {
5     "productID": "Org.pharma-network.phazmanet.drug@Paracetamol@003@",
6     "name": "Paracetamol",
7     "manufacturer": "Org.pharma-network.phazmanet.company@MAN001@Sun Pharma@",
8     "manufacturingDate": "2023-07-15",
9     "expiryDate": "2023-07-15",
10    "owner": "Org.pharma-network.phazmanet.company@MAN001@Sun Pharma@",
11    "shipment": ""
12 }
13 
```

- Strip 4 of drug named “paracetamol” created.

The screenshot shows the Postman interface with the following details:

- Left Sidebar:** Personal collections, environments, and history.
- Current Collection:** PharmaNet: Drug-Counterfeit-Distributed-Application / Collection-1 / Initiation.
- Request:** POST Paracetamol-004 to localhost:3000/addDrug.
- Body:** x-www-form-urlencoded (selected).

Key	Value	Description
organizationName	Manufacturer	
drugName	Paracetamol	
serialNo	004	
mfgDate	2023-07-15	
expDate	2023-07-15	
companyCRN	MAN001	
- Response:**

```

1
2 "status": "success",
3 "message": "New Drug got registered",
4 "newDrug": {
5     "productID": "Org.pharma-network.phazmanet.drug@Paracetamol@004@",
6     "name": "Paracetamol",
7     "manufacturer": "Org.pharma-network.phazmanet.company@MAN001@Sun Pharma@",
8     "manufacturingDate": "2023-07-15",
9     "expiryDate": "2023-07-15",
10    "owner": "Org.pharma-network.phazmanet.company@MAN001@Sun Pharma@",
11    "shipment": ""
12 }
13 
```

d. **Test Case - 2 : Supply Chain :**

When imported postman collection, we will get a folder name '**Supply Chain**' , containing the HTTP requests to perform the following functions as:

```
✓ PharmaNet: Drug-Counterfeit-Distributed-Application
  ✓ Collection-1
    > Initiation
      ✓ Supply Chain
        ✓ Part a
          POST VGPharma-PO
          POST Sun Pharma-Shipment
          POST FedEx-DeliverShipment
        ✓ Part b
          POST upgrad-PO
          POST VG Pharma-Shipment
          POST Blue Dart-Deliver Shipment
        ✓ Part c
          POST upgrad-Retail Drug
    > View History
```

- i. **Part a:** Purchase Order raised by 'VG Pharma' to purchase 3 strips of paracetamol from 'Sun Pharma' :

Expected Output: Display the PO object created in the response body.

The screenshot shows the Postman interface. On the left, the 'Personal' collection is expanded, showing 'History', 'mssc-teplantation', 'PharmaNet: Drug-Counterfeit-Distributed-Application' (which is expanded), 'Collection-1', 'Initiation', 'Supply Chain' (expanded), 'Part a' (expanded), and 'Part b'. Under 'Part a', there are four items: 'POST VGPharma-PO' (highlighted with a red arrow), 'POST Sun Pharma-Shipment', 'POST FedEx-DeliverShipment', and 'POST upgrad-PO'. Under 'Part b', there are three items: 'POST upgrad-PO', 'POST VG Pharma-Shipment', and 'POST Blue Dart-Deliver Shipment'. Under 'Part c', there is one item: 'POST upgrad-Retail Drug'. Under 'View History', there is one item: 'View History'. On the right, a POST request is being made to 'localhost:3000/createPO' with the following body parameters:

Key	Value	Description
organizationName	Distributor	
buyerCRN	DIST001	
sellerCRN	MAN001	
drugName	Paracetamol	
quantity	3	

The response body is:

```

1 "status": "success",
2 "message": "New PO got registered",
3 "newPurchaseOrder": {
4     "poID": "\$org.pharma-network.pharmanet.productOrders@DIST001@Paracetamol@",
5     "drugName": "Paracetamol",
6     "quantity": "3",
7     "buyer": "\$org.pharma-network.pharmanet.company@DIST001@VG Pharma@",
8     "seller": "\$org.pharma-network.pharmanet.company@MAN001@Sun Pharma@"
9 }
10
11
  
```

- ii. **Part a:** Shipment created by 'Sun Pharma' in response to the raised purchase order.

'FedEx' acts as the Transporter :

Expected Output: Display the Shipment object created in the response body.

The screenshot shows the Postman interface. On the left, the 'Personal' collection is expanded, showing 'History', 'mssc-teplantation', 'PharmaNet: Drug-Counterfeit-Distributed-Application' (which is expanded), 'Collection-1', 'Initiation', 'Supply Chain' (expanded), 'Part a' (expanded), and 'Part b'. Under 'Part a', there are two items: 'POST VGPharma-PO' and 'POST Sun Pharma-Shipment' (highlighted with a red arrow). Under 'Part b', there are three items: 'POST upgrad-PO', 'POST VG Pharma-Shipment', and 'POST Blue Dart-Deliver Shipment'. Under 'Part c', there is one item: 'POST upgrad-Retail Drug'. Under 'View History', there is one item: 'View History'. On the right, a POST request is being made to 'localhost:3000/createShipment' with the following body parameters:

Key	Value	Description
organizationName	Manufacturer	
buyerCRN	DIST001	
drugName	Paracetamol	
listOfAssets	001,002,003	
transporterCRN	TRA001	

The response body is:

```

1 "status": "success",
2 "message": "New Shipment got registered",
3 "newShipment": {
4     "shipmentID": "\$org.pharma-network.pharmanet.shipment@DIST001@Paracetamol@",
5     "creator": "\$x509::/C=US/ST=California/L=San Francisco/O=client/CN=Admin@manufacturer.pharma-supply-network.com::/C=US/ST=California/L=San Francisco/O=manufacturer.pharma-supply-network.com/CN=ca.manufacturer.pharma-supply-network.com", "assets": "\$org.pharma-network.pharmanet.drug@Paracetamol@001@", "\$org.pharma-network.pharmanet.drug@Paracetamol@002@", "\$org.pharma-network.pharmanet.drug@Paracetamol@003@", "transporter": "\$org.pharma-network.pharmanet.company@TRA001@FedEx@", "status": "in-transit"
  }
  
```

iii. **Part a:** ‘FedEx’ delivers the shipment to ‘VG pharma’

Expected Output: Display the data of each asset of the shipment.

The screenshot shows the Postman application interface. On the left, there is a sidebar with 'Personal' collections, environments, and history. Under 'Collection-1', there are several items: 'History', 'mssc-teplantation', 'PharmaNet: Drug-Counterfeit-Distributed-Application' (which is expanded to show 'Initiation', 'Supply Chain' (expanded to show 'Part a' with 'POST VGPharma-PO', 'POST Sun Pharma-Shipment', and 'POST FedEx-DeliverShipment'), 'Part b' (with 'POST upgrad-PO', 'POST VG Pharma-Shipment', and 'POST Blue Dart-Deliver Shipment'), 'Part c' (with 'POST upgrad-Retail Drug'), and 'View History'. A red arrow points from the sidebar to the 'POST FedEx-DeliverShipment' item in the 'Supply Chain' section. The main area of the interface shows a POST request to 'localhost:3000/updateShipment' for 'FedEx-DeliverShipment'. The 'Body' tab is selected, showing the following parameters:

Key	Value	Description
organizationName	Transporter	
buyerCRN	DIST001	
drugName	Paracetamol	
transporterCRN	TRA001	
organisationRole	Transporter	

Below the table, there is a JSON response body:

```

1
2
3
4
5
6
    "status": "success",
    "message": "New Updated Shipment got registered",
    "newUpdatedShipment": [{"productID": "\$org.pharma-network.phamanet.drug\Paracetamol\001\0", "name": "Paracetamol",
        "manufacturer": "\$org.pharma-network.phamanet.company\MAN001\Sun Pharma\", \"manufacturingDate\": \"2023-07-15\",
        "expiryDate": \"2023-07-15\", \"owner\": \"/org.pharma-network.phamanet.company\DIST001\VG Pharma\", \"shipment\": \"/org.
        pharma-network.phamanet.shipment\0IST001\Paracetamol\0\", \"productID\": \"/org.pharma-network.phamanet.
        drug\Paracetamol\002\0\", \"name\": \"Paracetamol\", \"manufacturer\": \"/org.pharma-network.phamanet.company\MAN001\Sun
        Pharma\", \"manufacturingDate\": \"2023-07-15\", \"expiryDate\": \"2023-07-15\", \"owner\": \"/org.pharma-network.phamanet.
        company\0IST001\VG Pharma\", \"shipment\": \"/org.pharma-network.phamanet.shipment\0IST001\Paracetamol\0\", \"productID\":
        \"/org.pharma-network.phamanet.drug\Paracetamol\003\0\", \"name\": \"Paracetamol\", \"manufacturer\": \"/org.
        pharma-network.phamanet.company\MAN001\Sun Pharma\", \"manufacturingDate\": \"2023-07-15\", \"expiryDate\": \"2023-07-15\",
        \"owner\": \"/org.pharma-network.phamanet.company\0IST001\VG Pharma\", \"shipment\": \"/org.pharma-network.phamanet.
        shipment\0IST001\Paracetamol\0\""}]
  
```

- iv. **Part b::** Purchase Order raised by 'upgrad' to purchase 2 strips of paracetamol from 'VG Pharma' :

Expected Output: Display the PO object created in the response body.

The screenshot shows the Postman interface with a collection named "PharmaNet: Drug-Counterfeit-Distributed-Application". A red arrow points to the "POST upgrad-PO" request under "Part b". The request URL is "localhost:3000/createPO". The "Body" tab is selected, showing the following JSON payload:

```

{
  "organizationName": "Retailer",
  "buyerCRN": "RET002",
  "sellerCRN": "DIST001",
  "drugName": "Paracetamol",
  "quantity": 2
}

```

The response body is displayed in JSON format:

```

{
  "status": "success",
  "message": "New PO got registered",
  "newPurchaseOrder": {
    "poID": "\$org.pharma-network.pharmanet.productOrders@RET002@Paracetamol@",
    "drugName": "Paracetamol",
    "quantity": "2",
    "buyer": "\$org.pharma-network.pharmanet.company@RET002@upgrad@",
    "seller": "\$org.pharma-network.pharmanet.company@DIST001@VG Pharma@"
  }
}

```

- v. **Part b::** Shipment created by 'VG Pharma' in response to the raised purchase order. 'Blue Dart' acts as the transporter :

Expected Output: Display the Shipment object created in the response body.

The screenshot shows the Postman interface with a collection named "PharmaNet: Drug-Counterfeit-Distributed-Application". A red arrow points to the "POST VG Pharma-Shipment" request under "Part b". The request URL is "localhost:3000/createShipment". The "Body" tab is selected, showing the following JSON payload:

```

{
  "organizationName": "Distributor",
  "buyerCRN": "RET002",
  "drugName": "Paracetamol",
  "listOfAssets": "001,002",
  "transporterCRN": "TRA002"
}

```

The response body is displayed in JSON format:

```

{
  "status": "success",
  "message": "New Shipment got registered",
  "newShipment": {
    "creator": "\$x5f99:/C=US/ST=California/L=San Francisco/OU=client/CN=Admin@distributor.pharma-supply-network.com::\$C=US/ST=California/L=San Francisco/O=distributor.pharma-supply-network.com/CN=ca.distributor.pharma-supply-network.com",
    "assets": "[\"\\$org.pharma-network.pharmanet.drug@Paracetamol@001@\", \"\\$org.pharma-network.pharmanet.drug@Paracetamol@002@\"]",
    "transporter": "\$org.pharma-network.pharmanet.company@TRA002@Blue Dart@"
  },
  "status": "in-transit"
}

```

vi. **Part b ::** ‘Blue Dart’ delivers the shipment to ‘upgrad’ :

Expected Output: Display the data of each asset of the shipment in the response body.

The screenshot shows the Postman interface with the following details:

- Left Panel (Collections):**
 - Personal
 - Collections
 - Environments
 - History
 - Part a
 - Part b
 - Part c
 - View History
- Request URL:** POST Blue Dart-Deliver Shipment
- Request Headers:** POST localhost:3000/updateShipment
- Request Body (JSON):**

Key	Value	Description
organizationName	Transporter	
buyerCRN	RET002	
drugName	Paracetamol	
transporterCRN	TRA002	
- Response:**
 - Status: 200 OK
 - Time: 2.94 s
 - Size: 1.3 KB
 - Save as Example

```

1 "status": "success",
2 "message": "New Updated Shipment got registered",
3 "newUpdatedShipment": [
4     {
5         "productID": "\$org.pharma-network.pharmanet.drug\$Paracetamol\$001\$",
6         "name": "Paracetamol", "manufacturer": "\$org.pharma-network.pharmanet.company\$MAN001\$Sun Pharma\$",
7         "manufacturingDate": "\$2023-07-15\\"", "expiryDate": "\$2023-07-15\\"", "owner": "\$org.pharma-network.
8         pharmanet.company\$RET002\$Upgrad\$", "shipment": "\$org.pharma-network.pharmanet.
9         shipment\$RET002\$Paracetamol\$002\$", "productID": "\$org.pharma-network.pharmanet.
10        drug\$Paracetamol\$002\$", "name": "Paracetamol", "manufacturer": "\$org.pharma-network.pharmanet.
11        company\$MAN001\$Sun Pharma\$", "manufacturingDate": "\$2023-07-15\\"", "expiryDate": "\$2023-07-15\\"",
12        "owner": "\$org.pharma-network.pharmanet.company\$RET002\$Upgrad\$", "shipment": "\$org.pharma-network.
13        pharmanet.shipment\$RET002\$Paracetamol\$002\$"
14     }
15 ]

```

vii. **Part c ::** A customer named 'Akash' with Aadhar Number 'AAD001' buys 1 paracetamol strip from the retailer 'upgrad' :

Expected Output: Display the data of the asset bought by Akash in the response body.

The screenshot shows the Postman interface with the following details:

- Request URL:** POST localhost:3000/retailDrug
- Body Content (x-www-form-urlencoded):**

Key	Description
organizationName	Retailer
drugName	Paracetamol
serialNo	001
retailerCRN	RET002
customerAadhar	AAD001
- Response Body (Pretty JSON):**

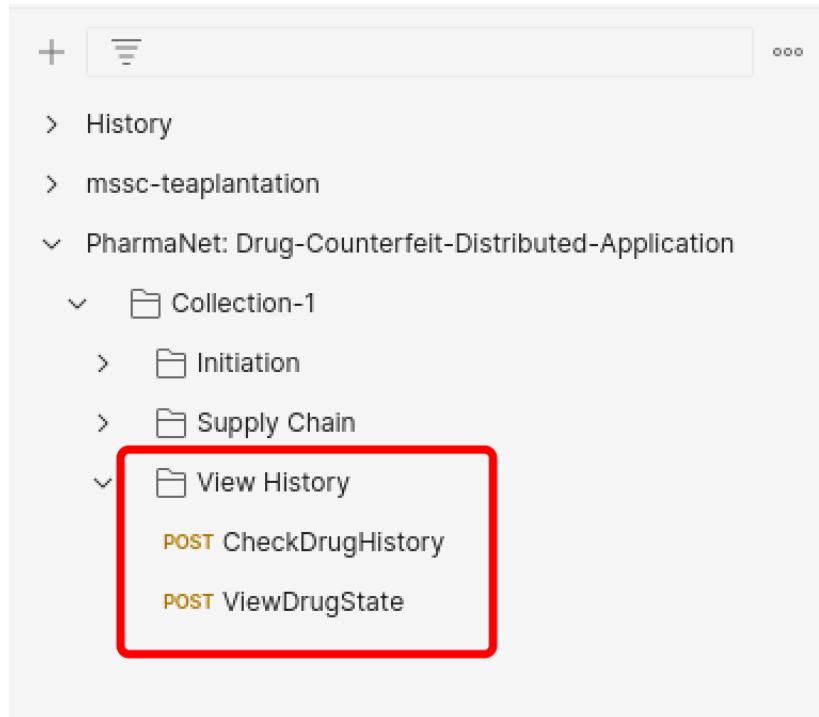
```

1  "status": "success",
2  "message": "New Purchase from customer",
3  "newPurchase": {
4    "productID": "org.pharma-network.pharmanet.drug#Paracetamol#001#",
5    "name": "Paracetamol",
6    "manufacturer": "org.pharma-network.pharmanet.company#MAN001#Sun Pharma#",
7    "manufacturingDate": "2023-07-15",
8    "expiryDate": "2023-07-15",
9    "owner": "AAD001",
10   "shipment": "org.pharma-network.pharmanet.shipment#RET002#Paracetamol#"
11 }
12
13

```

e. **Test Case - 3 : History Tracking :**

When imported postman collection, we will get a folder name '**View History**' , containing the HTTP requests to perform the following functions as:



- i. The customer 'Akash' wishes to check the history of the paracetamol that he bought from 'upgrad' :

Expected Output: The response body should display the entire lifecycle of the asset.

The screenshot shows the Postman interface with the following details:

- Collection:** CheckDrugHistory
- Method:** POST
- URL:** localhost:3000/viewHistory
- Body (x-www-form-urlencoded):**

Key	Value	Description
organizationName	Consumer	
- Response Status:** 200 OK
- Code Preview:**

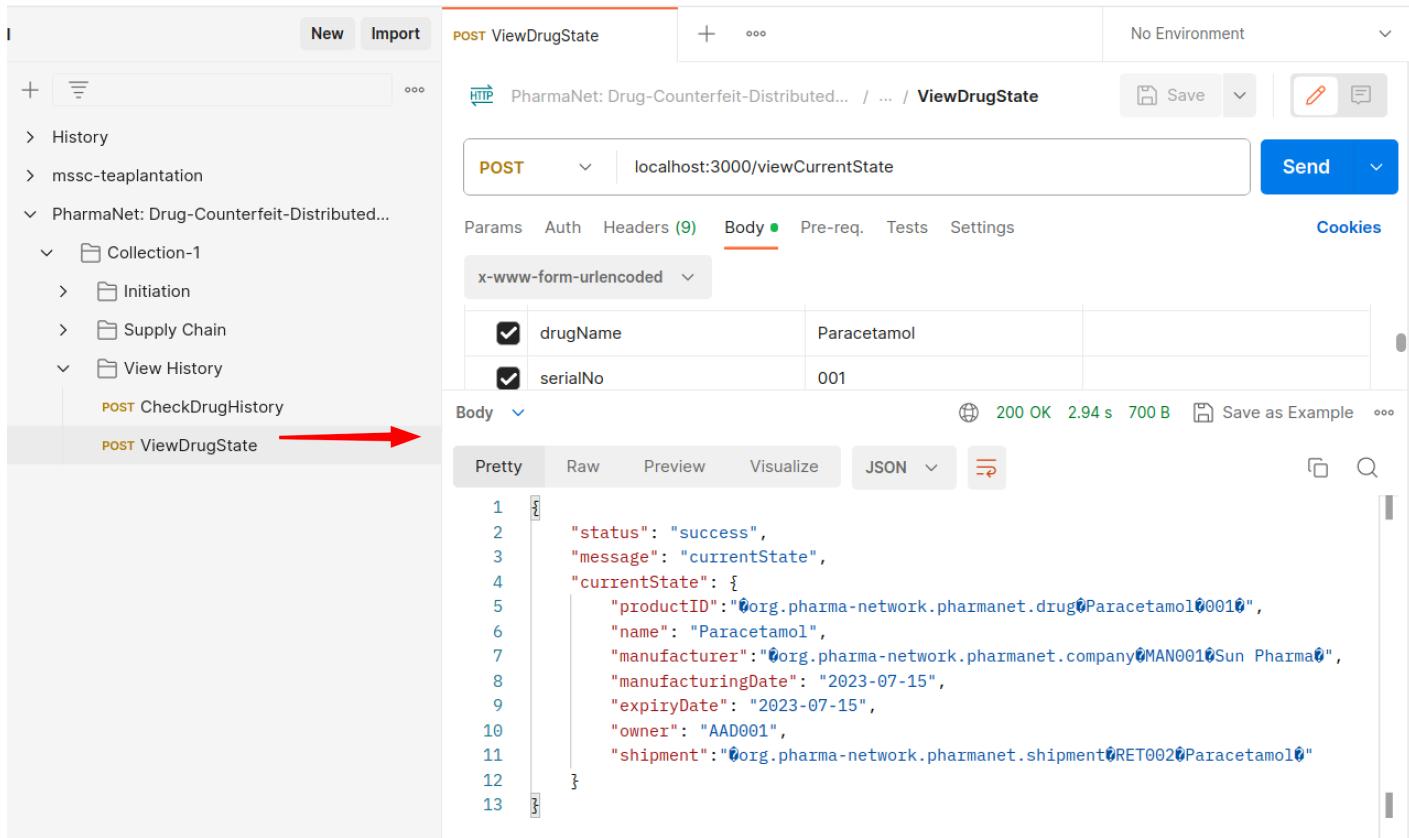
```

1
2   "status": "success",
3   "message": "historyOfDrug",
4   "historyOfDrug": [
5     {
6       "productID": "Org.pharma-network.pharmanet.drug@Paracetamol@001",
7       "name": "Paracetamol",
8       "manufacturer": "Org.pharma-network.pharmanet.company@MAN001@Sun Pharma",
9       "manufacturingDate": "2023-07-15",
10      "expiryDate": "2023-07-15",
11      "owner": "Org.pharma-network.pharmanet.company@MAN001@Sun Pharma",
12      "shipment": ""
13    },
14    {
15      "productID": "Org.pharma-network.pharmanet.drug@Paracetamol@001",
16      "name": "Paracetamol",
17      "manufacturer": "Org.pharma-network.pharmanet.company@MAN001@Sun Pharma",
18      "manufacturingDate": "2023-07-15",
19      "expiryDate": "2023-07-15",
20      "owner": "Org.pharma-network.pharmanet.company@TRA001@FedEx",
21      "shipment": ""
22    },
23    {
24      "productID": "Org.pharma-network.pharmanet.drug@Paracetamol@001",
25      "name": "Paracetamol",
26      "manufacturer": "Org.pharma-network.pharmanet.company@MAN001@Sun Pharma",
27      "manufacturingDate": "2023-07-15",
28      "expiryDate": "2023-07-15",
29      "owner": "Org.pharma-network.pharmanet.company@DIST001@VG Pharma",
30      "shipment": "Org.pharma-network.pharmanet.shipment@DIST001@Paracetamol"
31    },
32    {
33      "productID": "Org.pharma-network.pharmanet.drug@Paracetamol@001",
34      "name": "Paracetamol",
35      "manufacturer": "Org.pharma-network.pharmanet.company@MAN001@Sun Pharma",
36      "manufacturingDate": "2023-07-15",
37      "expiryDate": "2023-07-15",
38      "owner": "Org.pharma-network.pharmanet.company@TRA002@Blue Dart",
39      "shipment": "Org.pharma-network.pharmanet.shipment@DIST001@Paracetamol"
40    },
41    {
42      "productID": "Org.pharma-network.pharmanet.drug@Paracetamol@001",
43      "name": "Paracetamol",
44      "manufacturer": "Org.pharma-network.pharmanet.company@MAN001@Sun Pharma",
45      "manufacturingDate": "2023-07-15",
46      "expiryDate": "2023-07-15",
47      "owner": "Org.pharma-network.pharmanet.company@RET002@upgrad",
48      "shipment": "Org.pharma-network.pharmanet.shipment@RET002@Paracetamol"
49    }
50  ]
51 ]

```

- ii. The customer 'Akash' wishes to check the current state of the paracetamol that he bought from 'upgrad' :

Expected Output: The response body should display the current state of the asset.



The screenshot shows the Postman application interface. On the left, there's a sidebar with various projects and collections. A red arrow points to the 'POST ViewDrugState' entry under the 'PharmaNet: Drug-Counterfeit-Distributed...' collection. The main area shows a POST request to 'localhost:3000/viewCurrentState'. The 'Body' tab is selected, showing a JSON payload with 'drugName' set to 'Paracetamol' and 'serialNo' set to '001'. Below the request, the response is displayed in 'Pretty' format, showing a success message with a detailed product state object. The response status is 200 OK, and the time taken is 2.94 s.

```

1 {
2   "status": "success",
3   "message": "currentState",
4   "currentState": {
5     "productID": "org.pharma-network.pharmanet.drugParacetamol00010",
6     "name": "Paracetamol",
7     "manufacturer": "org.pharma-network.pharmanet.companyMAN001Sun Pharma",
8     "manufacturingDate": "2023-07-15",
9     "expiryDate": "2023-07-15",
10    "owner": "AAD001",
11    "shipment": "org.pharma-network.pharmanet.shipmentRET002Paracetamol"
12  }
13 }
```

7. Set of Useful commands and resources.

Below set of useful resources & commands can help to drop or update the network.

a. Command used to re-deploy the updated chaincode over the network.

NOTE: In case of change in the logic or anything of such sort, the updated chaincode can be installed over the network without re-deploying the network again. With every such updates required, **it is mandatory to change the chaincode version (VERSION NO) in shell-script file “fabricNetwork.sh”**. Below is the snippet from the “fabricNetwork.sh” file.

```
fabricNetwork.sh - pharma-net - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER          $ FabricNetwork.sh
PHARMA-NET         ...
    > application
    > chaincode
    > chaincode-advanced
    > network
        > bin
        > channel-artifacts
        > crypto-config
        > docker-base
        > fabric-samples
        > scripts
        ! configtx.yaml
        ! crypto-config.yaml
        docker-compose-e2e.yaml
        docker-compose-template.yaml
        docker-compose.yml
$ FabricNetwork.sh
> test
Network Information
Project Structure

network > $ FabricNetwork.sh
227  # }
228
229  # Generates Org certs using cryptogen tool
230  > function generateCerts() {
231  }
232
233  # Generate orderer genesis block, channel configuration transaction and
234  # anchor peer update transactions
235
236  #UPDATE required
237  > function generateChannelArtifacts() {
238  }
239
240  # timeout duration - the duration the CLI should wait for a response from
241  # another container before giving up
242  CLI_TIMEOUT=100
243
244  # default for delay between commands
245  CLI_DELAY=5
246
247  # channel name defaults to "registrationchannel"
248  CHANNEL_NAME="pharmachannel"
249  # version for updating chaincode
250  VERSION NO=1.0
251  # type of chaincode to be installed
252  TYPE="basic"
253
254  # use this as the default docker-compose yaml definition
255  COMPOSE_FILE=docker-compose-e2e.yaml
256
257  # use node as the default language for chaincode
258  LANGUAGE="node"
259
260  # default image tag
261  IMAGETAG="latest"
262
```

NOTE: Attached below the screenshot from the chaincode update using below command.

Ensure you are into the network directory as “pharma-net/network/”. Switch to the network directory using below command and then followed by the “update” command.

cd pharma-net/network/

./fabricNetwork.sh update

```
Updating chaincode for channel 'pharnachannel' with CLI timeout of '100' seconds and CLI delay of '5' seconds and chaincode version '1.1'
Continue? [Y/n] Y
proceeding ...
=====
        _VERSION=4.2
DOCKER_IMAGE_VERSION=v2.5.3
=====
===== WARNING =====
  Local Fabric binaries and docker images are
  out of sync. This may cause problems.
=====

START
Updating Chaincode Pharnet On Pharma Network

New Version : 1.1
+ peer chaincode install -n pharnet -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
Updating chaincode on peer0.manufacturer.pharma-supply-network.com ...
+ res0
+ set x+
[2023-07-19 15:51:54.062 UTC [chaincodeCmd] checkChaincodeCmdParams --> INFO 001 Using default escc
[2023-07-19 15:51:54.066 UTC [chaincodeCmd] checkChaincodeCmdParams --> INFO 001 Using default vscc
[2023-07-19 15:51:55.561 UTC [chaincodeCmd] Install --> INFO 003 Installed remotely response:<status:200 payload:"OK" >
===== Chaincode is installed on peer0.manufacturer =====

Updating chaincode on peer0.distributor.pharma-supply-network.com ...
+ peer chaincode install -n pharnet -v 1.1 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
+ res0
+ set x+
[2023-07-19 15:51:55.912 UTC [chaincodeCmd] checkChaincodeCmdParams --> INFO 001 Using default escc
[2023-07-19 15:51:55.913 UTC [chaincodeCmd] checkChaincodeCmdParams --> INFO 001 Using default vscc
[2023-07-19 15:51:55.499 UTC [chaincodeCmd] Install --> INFO 003 Installed remotely response:<status:200 payload:"OK" >
===== Chaincode is installed on peer0.distributor =====

Updating chaincode on peer0.retailer.pharma-supply-network.com ...
+ peer chaincode install -n pharnet -v 1.1 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
+ res0
+ set x+
[2023-07-19 15:51:55.779 UTC [chaincodeCmd] checkChaincodeCmdParams --> INFO 001 Using default escc
[2023-07-19 15:51:55.779 UTC [chaincodeCmd] checkChaincodeCmdParams --> INFO 001 Using default vscc
[2023-07-19 15:51:57.103 UTC [chaincodeCmd] Install --> INFO 003 Installed remotely response:<status:200 payload:"OK" >
===== Chaincode is installed on peer0.retailer =====

Updating chaincode on peer0.transporter.pharma-supply-network.com ...
+ peer chaincode install -n pharnet -v 1.1 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
+ res0
+ set x+
[2023-07-19 15:51:57.668 UTC [chaincodeCmd] checkChaincodeCmdParams --> INFO 001 Using default escc
[2023-07-19 15:51:57.668 UTC [chaincodeCmd] checkChaincodeCmdParams --> INFO 001 Using default vscc
[2023-07-19 15:51:57.668 UTC [chaincodeCmd] Install --> INFO 003 Installed remotely response:<status:200 payload:"OK" >
===== Chaincode is installed on peer0.transporter =====

Updating chaincode on peer0.consumer.pharma-supply-network.com ...
+ peer chaincode install -n pharnet -v 1.1 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
+ res0
+ set x+
[2023-07-19 15:51:58.630 UTC [chaincodeCmd] checkChaincodeCmdParams --> INFO 001 Using default escc
[2023-07-19 15:51:58.631 UTC [chaincodeCmd] checkChaincodeCmdParams --> INFO 001 Using default vscc
[2023-07-19 15:51:59.001 UTC [chaincodeCmd] Install --> INFO 003 Installed remotely response:<status:200 payload:"OK" >
===== Chaincode is installed on peer0.consumer =====

Upgrading chaincode on channel using peer0.manufacturer.pharma-supply-network.com ...
+ peer chaincode upgrade -o orderer.pharma-supply-network.com:7050 --channel pharnet -l node -v 1.1 -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/ -c '{"Args":["org.pharma-supply-network.com.pharnet:instantlate"]}' -P OR ('\'\''manufacturerMSP.member\'\'', \'\'\''retailerMSP.member\'\'', \'\'\''transporterMSP.member\'\'', \'\'\''consumerMSP.member\'\'')
+ res0
+ set x+

```

b. **Command used to make the network down.**

NOTE:

- i. If we make the network down using this way, as provided by the scripts then it is required to follow the steps again from **Point No. (2)** to start the network again because this way delete the client-identity and other generated artifacts, etc.
 - ii. Below is the screenshot from the network delete/remove using below command. **Ensure you are into the network directory as “pharma-net/network/”. Switch to the network directory using below command and then followed by the “down” command.**

cd pharma-net/network/

/fabricNetwork.sh down

```

spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ ./fabricNetwork.sh down
Stopping for channel 'pharmachannel' with CLI timeout of '100' seconds and CLI delay of '5' seconds and chaincode version '1.1'
Continue? [Y/n] Y
proceeding ...
(.) Running 39/30
  Container c1 Removed
  Container ca.manufacturer Removed
  Container ca.distributor.pharma-supply-network.com Removed
  Container peer1.distributor.pharma-supply-network.com Removed
  Container ca.manufacturer.pharma-supply-network.com Removed
  Container ca.supplier.pharma-supply-network.com Removed
  Container peer1.transporter.pharma-supply-network.com Removed
  Container peer0.orderer.pharma-supply-network.com Removed
  Container ca.consumer.pharma-supply-network.com Removed
  Container peer0.transporter.pharma-supply-network.com Removed
  Container peer1.consumer.pharma-supply-network.com Removed
  Container peer0.retailer.pharma-supply-network.com Removed
  Container peer1.retailer.pharma-supply-network.com Removed
  Container peer0.manufacturer.pharma-supply-network.com Removed
  Container peer0.distributor.pharma-supply-network.com Removed
  Container ca.transporter.pharma-supply-network.com Removed
  Container peer1.consumer.pharma-supply-network.com Removed
  Container peer0.retailer.pharma-supply-network.com Removed
  Volume network.pharma-supply-network.com Removed
  Volume network.peer0.manufacturer.pharma-supply-network.com Removed
  Volume network.peer1.consumer.pharma-supply-network.com Removed
  Volume network.peer0.supplier.pharma-supply-network.com Removed
  Volume network.peer0.consumer.pharma-supply-network.com Removed
  Volume network.peer0.transporter.pharma-supply-network.com Removed
  Volume network.peer0.retailer.pharma-supply-network.com Removed
  Volume network.peer1.distributor.pharma-supply-network.com Removed
  Volume network.peer1.retailer.pharma-supply-network.com Removed
  Network network.pharma-network Removed
.... No containers available for deletion ....
.... No Images available for deletion ....

```

fig. Shows the network delete command is given

To verify if we give command “docker ps” then no running container will be visible. Pls find the snapshot below

```

spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ docker ps
CONTAINER ID        IMAGE               COMMAND       CREATED          STATUS              PORTS          NAMES

```

Also we can check the volume attached to these containers is also empty

```

spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ docker volume ls
DRIVER      VOLUME NAME
spartans@spartans-Standard-PC-Q35-ICH9-2009:~/workspace/pharma-net/network$ 

```