Sudo ls
Su –
Apt-get update
Apt-get install sudo
Usermod -aG sudo cad
Su -c 'shutdown -r now'
Sudo apt-get update

VM installation guide - https://www.youtube.com/watch?v=x5MhydijWmc

Hyperleder Fabric Prerequisites Setup

1. Curl Installation
2. NodeJs Installation
3. Git Installation
4. Python Installation
5. Libtool
6. Docker CE
7. Docker Compose

Curl Installation

- Run below command to install Curl.
-
- sudo apt-get install curl
- **sudo apt install curl**
- Verify the installation and check the version of Curl using below command.
-
- curl --version

NodeJs Installation

- Open the terminal window and run below command to download and execute the nodejs file.
-
- curl -sL https://deb.nodesource.com/setup_10.x | sudo -E bash –
-
- Then run below command.
-
- sudo apt-get update
-
- Run below command to start the installation for NodeJs.
-
- sudo apt-get install nodejs
-
- Run below command to check if Nodejs is successfully installed or not. This should return the version
- of NodeJs.
-
- node --version

Git Installation

- Open the terminal window and run below command. This will start the installation for Git.
-
- sudo apt-get install git
-
- Run below command to check if Git is successfully installed or not. This should return the version of Git.
-
- git --version

Python Installation

- In the terminal window, run below command to install Python.
-
- sudo apt-get install python
-
- Verify the installation by running below command and that should return the version of Python.
-
- python --version

-

Lib Tools Installation

- Install Lib tools using below command.
-
- sudo apt-get install libltdl-dev

Install Docker CE (Community Edition)

- First download and then install it using below commands.
-
- wget https://download.docker.com/linux/ubuntu/dists/xenial/pool/stable/amd64/docker-ce 18.06.3~ce~3-0~ubuntu amd64.deb
-
-
- sudo dpkg -i docker-ce 18.06.3~ce~3-0~ubuntu amd64.deb
-
-
- Check the version of docker using below command and this should return the version of docker.
-
- docker –version
- **Docker sudo apt install docker**

Install Docker Compose

- Run below commands to setup Docker compose.
-
- sudo apt-get install python-pip
-
- pip --version
-
- sudo pip install docker-compose
-

- Verify the installation and check the version from below command.
-
- docker-compose version

Docker Compose

Sudo apt install docker-compose

Hyperledger Installation

Step 1: Run below command to download and setup Fabric.

- curl -sSL https://bit.ly/2ysbOFE | bash -s
-

You may encounter below issue when you run above command.

```
failed to get default registry endpoint from daemon (Got permission
denied while trying to connect to the Docker daemon socket at
unix:///var/run/docker.sock:
```

To fix this you need run below command.

- sudo chmod 666 /var/run/docker.sock

Peer channel

Step 1: Go to fabric-samples folder by using below command.

- `cd fabric-samples`

Step 2: Go to test-network folder by using below command.

- `cd test-network`

Step 3: Run below command to start your test-network

- `sudo ./network.sh up`

This start the network, you can run below command to check docker containers.

- `sudo docker ps`

This shows you three docker containers

1. One for Org1 peer node
2. One for Org2 peer node
3. One for Orderer

```
user@user-VirtualBox:~/fabric-samples/test-network$ sudo docker ps
CONTAINER ID        IMAGE                               COMMAND             CREATED             STATUS              PORTS
                    NAMES
42b0cc7bd413        hyperledger/fabric-peer:latest      "peer node start"   53 seconds ago      Up 47 seconds       0.0.0.0:
7051->7051/tcp              peer0.org1.example.com
0bbd3434dd98        hyperledger/fabric-peer:latest      "peer node start"   53 seconds ago      Up 47 seconds       7051/tcp
, 0.0.0.0:9051->9051/tcp    peer0.org2.example.com
bb282a207caa        hyperledger/fabric-orderer:latest   "orderer"           53 seconds ago      Up 48 seconds       0.0.0.0:
7050->7050/tcp              orderer.example.com
```

When you start the network, you will also not get any channel by default. You can check the channel by using below command.

- `sudo docker exec peer0.org1.example.com peer channel list`

This command shows you that, you don't have any channel created.

```
user@user-VirtualBox:~/fabric-samples/test-network$ sudo docker exec peer0.org1.example.com pee
r channel list
2020-07-20 04:24:03.863 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connec
tions initialized
Channels peers has joined:
user@user-VirtualBox:~/fabric-samples/test-network$
```

Step 4: Create new channel by using below command.

- `sudo ./network.sh createChannel -c testchannel`

This will create a new channel with the name testchannel.

To verify this channel creation, run below command on both the peers.

- `sudo docker exec peer0.org1.example.com peer channel list`
- `sudo docker exec peer0.org2.example.com peer channel list`

```
user@user-VirtualBox:~/fabric-samples/test-network$ sudo docker exec peer0.org1.example.com pee
r channel list
2020-07-20 04:30:57.454 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connec
tions initialized
Channels peers has joined:
testchannel
user@user-VirtualBox:~/fabric-samples/test-network$ sudo docker exec peer0.org2.example.com pee
r channel list
2020-07-20 04:31:37.895 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connec
tions initialized
Channels peers has joined:
testchannel
```

Step 5: To stop the network, you need to run below command.

- `sudo ./network.sh down`

couchdb

Step 1: Go to fabric-samples folder by using below command.

- `cd fabric-samples`

Step 2: Go to test-network folder by using below command.

- `cd test-network`

Step 3: Run below command to start the network and create couchDB containers as well.

- `sudo ./network.sh up -s couchdb`

This command starts your network and create couchdb container for each peer as well.

```
4febd384f8ec        hyperledger/fabric-peer:latest      "peer node start"        8 seconds ago       Up Less than a second   0.
0.0.0:7051->7051/tcp                        peer0.org1.example.com
f4d646ce6b54        hyperledger/fabric-peer:latest      "peer node start"        8 seconds ago       Up 1 second             70
51/tcp, 0.0.0.0:9051->9051/tcp              peer0.org2.example.com
e6b9e6496662        couchdb:3.1                         "tini -- /docker-e..."   21 seconds ago      Up 8 seconds            43
69/tcp, 9100/tcp, 0.0.0.0:5984->5984/tcp    couchdb0
985ff2a3d5e0        couchdb:3.1                         "tini -- /docker-e..."   21 seconds ago      Up 9 seconds            43
69/tcp, 9100/tcp, 0.0.0.0:7984->5984/tcp    couchdb1
885e3ab21013        hyperledger/fabric-orderer:latest   "orderer"                21 seconds ago      Up 9 seconds            0.
0.0.0:7050->7050/tcp                        orderer.example.com
```

Step 4: Create new channel by using below command.

- `sudo ./network.sh createChannel -c testchannel1`

This will create a new channel with the name testchannel1.

Step 5: To stop the network, you need to run below command.

- `sudo ./network.sh down`

CA

Step 1: Go to fabric-samples folder by using below command.

- `cd fabric-samples`

Step 2: Go to test-network folder by using below command.

- `cd test-network`

Step 3: Run below command to start your test-network and create CA container for each organization (one for orderer, one for org1 peer and one for org2 peer)

- `sudo ./network.sh up -ca`

```
CONTAINER ID        IMAGE                               COMMAND                  CREATED             STATUS                  PO
RTS                 NAMES
67799db77f67        hyperledger/fabric-peer:latest      "peer node start"        7 seconds ago       Up Less than a second   70
51/tcp, 0.0.0.0:9051->9051/tcp   peer0.org2.example.com
10677ec49128        hyperledger/fabric-peer:latest      "peer node start"        7 seconds ago       Up 1 second             0.
0.0.0:7051->7051/tcp             peer0.org1.example.com
ea584da2f05d        hyperledger/fabric-orderer:latest   "orderer"                7 seconds ago       Up 1 second             0.
0.0.0:7050->7050/tcp             orderer.example.com
5a82a9350f0d        hyperledger/fabric-ca:latest        "sh -c 'fabric-ca-..."   38 seconds ago      Up 32 seconds           70
54/tcp, 0.0.0.0:9054->9054/tcp   ca_orderer
4166d278cbd8        hyperledger/fabric-ca:latest        "sh -c 'fabric-ca-..."   38 seconds ago      Up 32 seconds           70
54/tcp, 0.0.0.0:8054->8054/tcp   ca_org2
b039ae22ebc0        hyperledger/fabric-ca:latest        "sh -c 'fabric-ca-..."   38 seconds ago      Up 31 seconds           0.
0.0.0:7054->7054/tcp             ca_org1
```

Step 4: Create new channel by using below command.

- `sudo ./network.sh createChannel -c testchannel2`

This will create a new channel with the name testchannel2.

Step 5: To stop the network, you need to run below command.

- `sudo ./network.sh down`

ALL

Step 1: Go to fabric-samples folder by using below command.

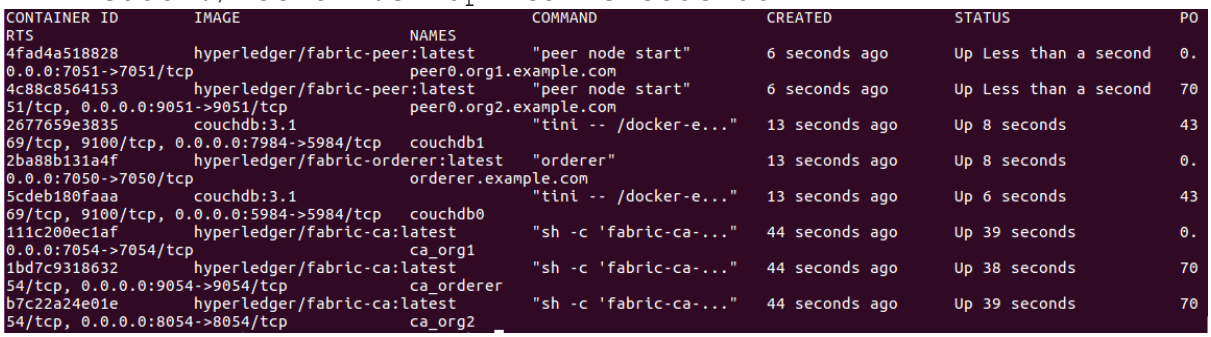
- `cd fabric-samples`

Step 2: Go to test-network folder by using below command.

- `cd test-network`

Step 3: Run below command to start your test-network with all the containers (2 peers, orderer, 3 ca, 2 couchdb).

- `sudo ./network.sh up -ca -s couchdb`



Step 4: Create new channel by using below command.

- `sudo ./network.sh createChannel -c testchannel3`

This will create a new channel with the name testchannel3.

Step 5: To stop the network, you need to run below command.

- `sudo ./network.sh down`

```
cd fabric-samples/first-network/
fabric-samples/first-network$ ll
```

```
Cat byfn.sh
```

```
./byfn.sh generate
```

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

```
./byfn.sh up -l java
```

Hyperledger: Hyperledger -https://www.hyperledger.org/

https://hyperledger-fabric.readthedocs.io/en/release-2.2/write_first_app.html

Curl –

sudo apt install curl

Docker

Sudo apt install docker

Docker Compose

Sudo apt install docker-compose

Golang

Sudo apt install golang-go

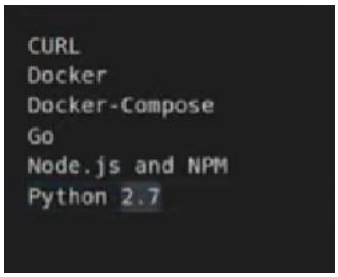
For Git clone

git config --global core.autocrlf false

git config --global core.longpaths true

Run the script

curl -sSL http://bit.ly/2ysbOFE | bash -s



Code: A57078

R20

Anurag University

IV- B Tech I-Semester End Examinations
Course: Introduction to BlockChain Technology
(Common to IT/CSE CS)

Time:3Hours **MaxMarks:60**

Section –A(Short Answer type questions)
Answer all questions: (10 x 2 =20Marks)

Unit-I

1. Explain the concept of a "distributed ledger" and how it relates to blockchain technology. (CO1,L2)

2. What is Proof of Stake (PoS), and how does it differ from Proof of Work? (CO1,L1)

Unit-II

3. Define "enum" in Solidity and explain its typical use. (CO2, L1)
4. Develop a Solidity function that uses a loop to iterate through an array of numbers and return the sum of all even numbers. (CO2,L3)

Unit-III

5. Explain the steps involved in developing a DApp with Truffle IDE. (CO3,L2)
6. Explain the importance of privacy and permissions in MultiChain. (CO3,L2)

Unit-IV

7. What is the role of APIs in Hyperledger, and how do they enable interaction with the blockchain network? (CO4,L2)
8. Discuss the concept of consensus in the context of Hyperledger. (CO4,L2)

Unit-V

9. Illustrate the role of blockchain in transforming the real estate industry. (CO5,L3)
10. Discuss the potential impact of blockchain on reducing corruption in government

operations and public services. (CO5,L3)

Section—B (Essay Answer type questions)
Answer all questions:(5x8=40 Marks)

Unit-I

11. A) Differentiate between a "soft fork" and a "hard fork" in blockchain. (CO1,L2)

OR

B) i. Describe how blockchain technology can ensure transaction anonymity while maintaining transparency. What are the potential challenges associated with balancing anonymity and transparency? (CO1,L3)

ii. Describe the purpose and structure of a Merkle Tree in blockchain data organization. (CO1,L2)

Unit-II

12. A) Discuss the concept of inheritance in Solidity. How does it enable code reuse and organization in smart contracts? (CO2,L2)

OR

B) i. Explain the concept of self-destruction in Solidity. (CO2, L1)

ii. Implement a constructor in a Solidity contract that initializes key variables when the contract is deployed. Provide a practical example where this would be necessary. (CO2, L3)

Unit-III

13. A) Discuss the significance of truffle test in the context of DApp development. How does it ensure the functionality and security of smart contracts? (CO3,L3)

OR

B) i. Describe the key steps involved in setting up a private blockchain using MultiChain. What are the essential configuration parameters? (CO3, L2)

ii. Define "Blockchain Bytes" and explain their relevance in the context of MultiChain and blockchain data representation. (CO3, L1)

Unit-IV

14. A) Discuss the concept of consensus in the context of Hyperledger. How does it ensure trust and agreement among network participants? (CO4,L2)

OR

B) Explain the Hyperledger Fabric model and its role in creating a private, permissioned blockchain network. What are the key components of this model? (CO4, L2)

Unit-V

15. A) Discuss a healthcare-related case study where blockchain has been utilized to enhance patient data security and interoperability. What were the key benefits of this implementation?. (CO5,L2)

OR

B) i. What are some challenges that businesses may face when implementing blockchain solutions, and how can they address these challenges? (CO5,L2)

ii. Discuss the potential impact of blockchain on reducing corruption in government operations and public services. Provide specific examples or case studies to support your analysis. (CO5,L3)