**GUARD platform**

Blockchain component configuration and setup guide

This document is intended to be used as step-by-step configuration instruction to set up working environment of Blockchain network playing the role of permanent ledger in GUARD platform.

Prerequisites:

* 3 Virtual Machines with installed Docker playing role of Blockchain Peers (one of them also playing role of Orderer)
* Account with administrative access during configuration
* Access to the docker images
* Access to valid SSL certificates

Legend:

1. **Bold red** – instructions
2. *Italic black* – code snippets/commands to be used directly or after adjusting
3. **Run docker containers**

*docker run -d -p 7150:7150 \*

*--name orderers-tlsca \*

*-e FABRIC\_CA\_HOME='/var/hyperledger/production' \*

*-v ~/orderers-tlsca-data:/var/hyperledger/production \*

*hyperledger/fabric-ca:amd64-1.4.9 sh -c 'fabric-ca-server start'*

*docker run -d -p 7151:7151 \*

*--name peers-tlsca \*

*-e FABRIC\_CA\_HOME='/var/hyperledger/production' \*

*-v ~/peers-tlsca-data:/var/hyperledger/production \*

*hyperledger/fabric-ca:amd64-1.4.9 sh -c 'fabric-ca-server start'*

*docker run -d -p 7152:7152 \*

*--name orderers-rca \*

*-e FABRIC\_CA\_HOME='/var/hyperledger/production' \*

*-v ~/orderers-rca-data:/var/hyperledger/production \*

*hyperledger/fabric-ca:amd64-1.4.9 sh -c 'fabric-ca-server start'*

*docker run -d -p 7153:7153 \*

*--name peers-rca \*

*-e FABRIC\_CA\_HOME='/var/hyperledger/production' \*

*-v ~/peers-rca-data:/var/hyperledger/production \*

*hyperledger/fabric-ca:amd64-1.4.9 sh -c 'fabric-ca-server start'*

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

1. **Run scripts for set proper configuration**

*source config\_v1.sh*

*source functions.sh*

1. **Download Hyperledger Fabric binaries**
2. **Run commands for enroll users in fabric-ca-client**

*bin/fabric-ca-client enroll -u https://$ORDERERS\_TLSCA\_SERVER\_USER:$ORDERERS\_TLSCA\_SERVER\_PASS@10.1.0.4:7150 -H ${PWD}/data/orderers-tlsca-admin/ --tls.certfiles ${PWD}/certs/orderers-tlsca-cert.pem*

*bin/fabric-ca-client enroll -u https://$PEERS\_TLSCA\_SERVER\_USER:$PEERS\_TLSCA\_SERVER\_PASS@10.1.0.4:7151 -H ${PWD}/data/peers-tlsca-admin/ --tls.certfiles ${PWD}/certs/peers-tlsca-cert.pem*

*bin/fabric-ca-client enroll -u https://$ORDERERS\_RCA\_SERVER\_USER:$ORDERERS\_RCA\_SERVER\_PASS@10.1.0.4:7152 -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem*

*bin/fabric-ca-client enroll -u https://$PEERS\_RCA\_SERVER\_USER:$PEERS\_RCA\_SERVER\_PASS@10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem*

1. **For each VMs run commands for register users in fabric-ca-client**

*bin/fabric-ca-client register --id.name $ORDERER1\_USER --id.secret $ORDERER1\_TLSCA\_PASS --id.type orderer -u https://10.1.0.4:7150 -H ${PWD}/data/orderers-tlsca-admin/ --tls.certfiles ${PWD}/certs/orderers-tlsca-cert.pem*

*bin/fabric-ca-client register --id.name $ORDERERS\_ADMIN\_USER --id.secret $ORDERERS\_ADMIN\_TLSCA\_PASS --id.type admin -u https://10.1.0.4:7150 -H ${PWD}/data/orderers-tlsca-admin/ --tls.certfiles ${PWD}/certs/orderers-tlsca-cert.pem*

*bin/fabric-ca-client register --id.name $PEER1\_USER --id.secret $PEER1\_TLSCA\_PASS --id.type peer -u https://10.1.0.4:7151 -H ${PWD}/data/peers-tlsca-admin/ --tls.certfiles ${PWD}/certs/peers-tlsca-cert.pem*

*bin/fabric-ca-client register --id.name $PEERS\_ADMIN\_USER --id.secret $PEERS\_ADMIN\_TLSCA\_PASS --id.type admin -u https://10.1.0.4:7151 -H ${PWD}/data/peers-tlsca-admin/ --tls.certfiles ${PWD}/certs/peers-tlsca-cert.pem --id.attrs "hf.Registrar.Roles=client,hf.Registrar.Attributes=\*,hf.Revoker=true,hf.GenCRL=true,admin=true:ecert,abac.init=true:ecert"*

*bin/fabric-ca-client register --id.name $ORDERER1\_USER --id.secret $ORDERER1\_RCA\_PASS --id.type orderer -u https://10.1.0.4:7152 -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem*

*bin/fabric-ca-client register --id.name $ORDERERS\_ADMIN\_USER --id.secret $ORDERERS\_ADMIN\_RCA\_PASS --id.type admin -u https://10.1.0.4:7152 -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem*

*bin/fabric-ca-client register --id.name $PEER1\_USER --id.secret $PEER1\_RCA\_PASS --id.type peer -u https://10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem*

*bin/fabric-ca-client register --id.name $PEERS\_ADMIN\_USER --id.secret $PEERS\_ADMIN\_RCA\_PASS --id.type admin -u https://10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem --id.attrs "hf.Registrar.Roles=client,hf.Registrar.Attributes=\*,hf.Revoker=true,hf.GenCRL=true,admin=true:ecert,abac.init=true:ecert"*

*bin/fabric-ca-client register --id.name $PEERS\_API\_USER --id.secret $PEERS\_API\_PASS --id.type client -u https://10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem*

#v2

*bin/fabric-ca-client register --id.name $ORDERER2\_USER --id.secret $ORDERER2\_TLSCA\_PASS --id.type orderer -u https://10.1.0.4:7150 -H ${PWD}/data/orderers-tlsca-admin/ --tls.certfiles ${PWD}/certs/orderers-tlsca-cert.pem*

*bin/fabric-ca-client register --id.name $PEER2\_USER --id.secret $PEER2\_TLSCA\_PASS --id.type peer -u https://10.1.0.4:7151 -H ${PWD}/data/peers-tlsca-admin/ --tls.certfiles ${PWD}/certs/peers-tlsca-cert.pem*

*bin/fabric-ca-client register --id.name $ORDERER2\_USER --id.secret $ORDERER2\_RCA\_PASS --id.type orderer -u https://10.1.0.4:7152 -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem*

*bin/fabric-ca-client register --id.name $PEER2\_USER --id.secret $PEER2\_RCA\_PASS --id.type peer -u https://10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem*

#v3

*bin/fabric-ca-client register --id.name $ORDERER3\_USER --id.secret $ORDERER3\_TLSCA\_PASS --id.type orderer -u https://10.1.0.4:7150 -H ${PWD}/data/orderers-tlsca-admin/ --tls.certfiles ${PWD}/certs/orderers-tlsca-cert.pem*

*bin/fabric-ca-client register --id.name $PEER3\_USER --id.secret $PEER3\_TLSCA\_PASS --id.type peer -u https://10.1.0.4:7151 -H ${PWD}/data/peers-tlsca-admin/ --tls.certfiles ${PWD}/certs/peers-tlsca-cert.pem*

*bin/fabric-ca-client register --id.name $ORDERER3\_USER --id.secret $ORDERER3\_RCA\_PASS --id.type orderer -u https://10.1.0.4:7152 -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem*

*bin/fabric-ca-client register --id.name $PEER3\_USER --id.secret $PEER3\_RCA\_PASS --id.type peer -u https://10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem*

1. **For each VMs run commands for enroll addresses in fabric-ca-client and move configuration**

#v1

*bin/fabric-ca-client enroll -u https://$ORDERER1\_USER:$ORDERER1\_TLSCA\_PASS@10.1.0.4:7150 --enrollment.profile tls -H ${PWD}/data/orderers-tlsca-admin/ --tls.certfiles ${PWD}/certs/orderers-tlsca-cert.pem -M ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER1\_USER/tls --csr.hosts orderer1,localhost,orderer1.germanywestcentral.cloudapp.azure.com,10.1.0.4*

*bin/fabric-ca-client enroll -u https://$ORDERERS\_ADMIN\_USER:$ORDERERS\_ADMIN\_TLSCA\_PASS@10.1.0.4:7150 --enrollment.profile tls -H ${PWD}/data/orderers-tlsca-admin/ --tls.certfiles ${PWD}/certs/orderers-tlsca-cert.pem -M ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/users/$ORDERERS\_ADMIN\_USER/tls*

*bin/fabric-ca-client enroll -u https://$PEER1\_USER:$PEER1\_TLSCA\_PASS@10.1.0.4:7151 --enrollment.profile tls -H ${PWD}/data/peers-tlsca-admin/ --tls.certfiles ${PWD}/certs/peers-tlsca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER1\_USER/tls --csr.hosts peer1,localhost,peer1.germanywestcentral.cloudapp.azure.com,10.1.0.4*

*bin/fabric-ca-client enroll -u https://$PEERS\_ADMIN\_USER:$PEERS\_ADMIN\_TLSCA\_PASS@10.1.0.4:7151 --enrollment.profile tls -H ${PWD}/data/peers-tlsca-admin/ --tls.certfiles ${PWD}/certs/peers-tlsca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/users/$PEERS\_ADMIN\_USER/tls*

*bin/fabric-ca-client enroll -u https://$ORDERER1\_USER:$ORDERER1\_RCA\_PASS@10.1.0.4:7152 -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem -M ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER1\_USER/msp --csr.hosts orderer1,localhost,orderer1.germanywestcentral.cloudapp.azure.com,10.1.0.4*

*bin/fabric-ca-client enroll -u https://$ORDERERS\_ADMIN\_USER:$ORDERERS\_ADMIN\_RCA\_PASS@10.1.0.4:7152 -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem -M ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/users/$ORDERERS\_ADMIN\_USER/msp*

*bin/fabric-ca-client enroll -u https://$PEER1\_USER:$PEER1\_RCA\_PASS@10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER1\_USER/msp --csr.hosts peer1,localhost,peer1.germanywestcentral.cloudapp.azure.com,10.1.0.4*

*bin/fabric-ca-client enroll -u https://$PEERS\_ADMIN\_USER:$PEERS\_ADMIN\_RCA\_PASS@10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/users/$PEERS\_ADMIN\_USER/msp*

*bin/fabric-ca-client enroll -u https://$PEERS\_API\_USER:$PEERS\_API\_PASS@10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/users/$PEERS\_API\_USER/msp*

*mv ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER1\_USER/tls/keystore/\* ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER1\_USER/tls/keystore/priv\_sk*

*mv ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/users/$ORDERERS\_ADMIN\_USER/tls/keystore/\* ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/users/$ORDERERS\_ADMIN\_USER/tls/keystore/priv\_sk*

*mv ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER1\_USER/tls/keystore/\* ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER1\_USER/tls/keystore/priv\_sk*

*mv ${PWD}/crypto-config/peerOrganizations/guard-project.eu/users/$PEERS\_ADMIN\_USER/tls/keystore/\* ${PWD}/crypto-config/peerOrganizations/guard-project.eu/users/$PEERS\_ADMIN\_USER/tls/keystore/priv\_sk*

*mv ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER1\_USER/msp/keystore/\* ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER1\_USER/msp/keystore/priv\_sk*

*mv ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/users/$ORDERERS\_ADMIN\_USER/msp/keystore/\* ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/users/$ORDERERS\_ADMIN\_USER/msp/keystore/priv\_sk*

*mv ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER1\_USER/msp/keystore/\* ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER1\_USER/msp/keystore/priv\_sk*

*mv ${PWD}/crypto-config/peerOrganizations/guard-project.eu/users/$PEERS\_ADMIN\_USER/msp/keystore/\* ${PWD}/crypto-config/peerOrganizations/guard-project.eu/users/$PEERS\_ADMIN\_USER/msp/keystore/priv\_sk*

*mv ${PWD}/crypto-config/peerOrganizations/guard-project.eu/users/$PEERS\_API\_USER/msp/keystore/\* ${PWD}/crypto-config/peerOrganizations/guard-project.eu/users/$PEERS\_API\_USER/msp/keystore/priv\_sk*

#v2

*bin/fabric-ca-client enroll -u https://$ORDERER2\_USER:$ORDERER2\_TLSCA\_PASS@10.1.0.4:7150 --enrollment.profile tls -H ${PWD}/data/orderers-tlsca-admin/ --tls.certfiles ${PWD}/certs/orderers-tlsca-cert.pem -M ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER2\_USER/tls --csr.hosts orderer2,localhost,orderer2.germanywestcentral.cloudapp.azure.com,10.1.0.6*

*bin/fabric-ca-client enroll -u https://$PEER2\_USER:$PEER2\_TLSCA\_PASS@10.1.0.4:7151 --enrollment.profile tls -H ${PWD}/data/peers-tlsca-admin/ --tls.certfiles ${PWD}/certs/peers-tlsca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER2\_USER/tls --csr.hosts peer2,localhost,peer2.germanywestcentral.cloudapp.azure.com,10.1.0.6*

*bin/fabric-ca-client enroll -u https://$ORDERER2\_USER:$ORDERER2\_RCA\_PASS@10.1.0.4:7152 -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem -M ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER2\_USER/msp --csr.hosts orderer2,localhost,orderer2.germanywestcentral.cloudapp.azure.com,10.1.0.6*

*bin/fabric-ca-client enroll -u https://$PEER2\_USER:$PEER2\_RCA\_PASS@10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER2\_USER/msp --csr.hosts peer2,localhost,peer2.germanywestcentral.cloudapp.azure.com,10.1.0.6*

*mv ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER2\_USER/tls/keystore/\* ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER2\_USER/tls/keystore/priv\_sk*

*mv ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER2\_USER/tls/keystore/\* ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER2\_USER/tls/keystore/priv\_sk*

*mv ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER2\_USER/msp/keystore/\* ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER2\_USER/msp/keystore/priv\_sk*

*mv ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER2\_USER/msp/keystore/\* ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER2\_USER/msp/keystore/priv\_sk*

#v3

*bin/fabric-ca-client enroll -u https://$ORDERER3\_USER:$ORDERER3\_TLSCA\_PASS@10.1.0.4:7150 --enrollment.profile tls -H ${PWD}/data/orderers-tlsca-admin/ --tls.certfiles ${PWD}/certs/orderers-tlsca-cert.pem -M ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER3\_USER/tls --csr.hosts orderer3,localhost,orderer3.germanywestcentral.cloudapp.azure.com,10.1.0.8*

*bin/fabric-ca-client enroll -u https://$PEER3\_USER:$PEER3\_TLSCA\_PASS@10.1.0.4:7151 --enrollment.profile tls -H ${PWD}/data/peers-tlsca-admin/ --tls.certfiles ${PWD}/certs/peers-tlsca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER3\_USER/tls --csr.hosts peer3,localhost,peer3.germanywestcentral.cloudapp.azure.com,10.1.0.8*

*bin/fabric-ca-client enroll -u https://$ORDERER3\_USER:$ORDERER3\_RCA\_PASS@10.1.0.4:7152 -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem -M ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER3\_USER/msp --csr.hosts orderer3,localhost,orderer3.germanywestcentral.cloudapp.azure.com,10.1.0.8*

*bin/fabric-ca-client enroll -u https://$PEER3\_USER:$PEER3\_RCA\_PASS@10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER3\_USER/msp --csr.hosts peer3,localhost,peer3.germanywestcentral.cloudapp.azure.com,10.1.0.8*

*mv ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER3\_USER/tls/keystore/\* ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER3\_USER/tls/keystore/priv\_sk*

*mv ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER3\_USER/tls/keystore/\* ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER3\_USER/tls/keystore/priv\_sk*

*mv ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER3\_USER/msp/keystore/\* ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/orderers/$ORDERER3\_USER/msp/keystore/priv\_sk*

*mv ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER3\_USER/msp/keystore/\* ${PWD}/crypto-config/peerOrganizations/guard-project.eu/peers/$PEER3\_USER/msp/keystore/priv\_sk*

1. **Generate root Membership Service Providers for orderers. Will be saved in genesis.block:**

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

#v1

*#fill msp/cacerts*

*bin/fabric-ca-client getcacert -u https://10.1.0.4:7152 -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem -M ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/msp*

*#fill msp/tlscacerts*

*bin/fabric-ca-client getcacert -u https://10.1.0.4:7150 -H ${PWD}/data/orderers-tlsca-admin/ --tls.certfiles ${PWD}/certs/orderers-tlsca-cert.pem -M ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/msp --enrollment.profile tls*

*#fill msp/admincerts*

*bin/fabric-ca-client certificate list --id $ORDERERS\_ADMIN\_USER -H ${PWD}/data/orderers-rca-admin/ --tls.certfiles ${PWD}/certs/orderers-rca-cert.pem --store ${PWD}/crypto-config/ordererOrganizations/guard-project.eu/msp/admincerts*

1. **Generate root Membership Service Providers for peers. Will be saved in genesis.block**

*#fill msp/cacerts*

*bin/fabric-ca-client getcacert -u https://10.1.0.4:7153 -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/msp*

*#rename certificate to peers-msp-cert.pem to match peers-config.yaml*

*mv ${PWD}/crypto-config/peerOrganizations/guard-project.eu/msp/cacerts/\* ${PWD}/crypto-config/peerOrganizations/guard-project.eu/msp/cacerts/peers-msp-cert.pem*

*#fill msp/tlscacerts*

*bin/fabric-ca-client getcacert -u https://10.1.0.4:7151 -H ${PWD}/data/peers-tlsca-admin/ --tls.certfiles ${PWD}/certs/peers-tlsca-cert.pem -M ${PWD}/crypto-config/peerOrganizations/guard-project.eu/msp --enrollment.profile tls*

*#fill msp/admincerts*

*bin/fabric-ca-client certificate list --id $PEERS\_ADMIN\_USER -H ${PWD}/data/peers-rca-admin/ --tls.certfiles ${PWD}/certs/peers-rca-cert.pem --store ${PWD}/crypto-config/peerOrganizations/guard-project.eu/msp/admincerts*

1. **Save file config/peers-config.yaml to /crypto-config/peerOrganizations/guard-project.eu/msp/config.yaml folder**
2. **Generate genesis.block**

bin/configtxgen -profile GuardOrdererGenesis -channelID system -outputBlock ./channel-artifacts/genesis.block

1. **Generate channel**

bin/configtxgen -profile GuardChannel -outputCreateChannelTx ./channel-artifacts/channel.tx -channelID "guard"

1. **Generate anchor tx for guard peer**

bin/configtxgen -profile GuardChannel -outputAnchorPeersUpdate ./channel-artifacts/guardAnchors.tx -channelID "guard" -asOrg GuardMSP

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

1. **For each ordererX folder copy admincerts with certificates from folder crypto-config/ordererOrganizations/guard-project.eu/msp/admincerts/**
2. **For each ordererX/msp/cacerts rename to `orderers-msp-cert.pem`**
3. **For each ordererX/tls/tlscacerts rename to `orderers-tlsca-cert.pem`**
4. **Run docker containers**

*docker run -d -p 7050:7050 \*

*--name orderer1 \*

*--health-cmd='wget --no-verbose --tries=1 --spider http://localhost:8443/healthz || exit 1 ' \*

*--health-interval=2s \*

*--restart always \*

*-e FABRIC\_LOGGING\_SPEC='INFO' \*

*-e ORDERER\_OPERATIONS\_LISTENADDRESS='localhost:8443' \*

*-e ORDERER\_METRICS\_PROVIDER='prometheus' \*

*-e ORDERER\_GENERAL\_GENESISFILE='/var/hyperledger/orderer/genesis.block' \*

*-e ORDERER\_GENERAL\_BOOTSTRAPMETHOD='file' \*

*-e ORDERER\_CHANNELPARTICIPATION\_ENABLED='true' \*

*-e ORDERER\_GENERAL\_LISTENADDRESS='0.0.0.0' \*

*-e ORDERER\_GENERAL\_LISTENPORT='7050' \*

*-e ORDERER\_GENERAL\_LOCALMSPDIR='/var/hyperledger/orderer/msp' \*

*-e ORDERER\_GENERAL\_LOCALMSPID='GuardOrdererMSP' \*

*-e ORDERER\_GENERAL\_TLS\_CERTIFICATE='/var/hyperledger/orderer/tls/signcerts/cert.pem' \*

*-e ORDERER\_GENERAL\_TLS\_ENABLED='true' \*

*-e ORDERER\_GENERAL\_TLS\_PRIVATEKEY='/var/hyperledger/orderer/tls/keystore/priv\_sk' \*

*-e ORDERER\_GENERAL\_TLS\_ROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_GENERAL\_CLUSTER\_CLIENTCERTIFICATE='/var/hyperledger/orderer/tls/signcerts/cert.pem' \*

*-e ORDERER\_GENERAL\_CLUSTER\_CLIENTPRIVATEKEY='/var/hyperledger/orderer/tls/keystore/priv\_sk' \*

*-e ORDERER\_GENERAL\_CLUSTER\_ROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_ADMIN\_TLS\_CERTIFICATE='/var/hyperledger/orderer/tls/signcerts/cert.pem' \*

*-e ORDERER\_ADMIN\_TLS\_ENABLED='true' \*

*-e ORDERER\_ADMIN\_TLS\_PRIVATEKEY='/var/hyperledger/orderer/tls/keystore/priv\_sk' \*

*-e ORDERER\_ADMIN\_TLS\_ROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_ADMIN\_TLS\_CLIENTROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_ADMIN\_LISTENADDRESS='localhost:9050' \*

*-v ~/orderer1-data:/var/hyperledger/production \*

*-v ~/crypto-config/ordererOrganizations/guard-project.eu/orderers/orderer1:/var/hyperledger/orderer \*

*hyperledger/fabric-orderer:amd64-2.3.1 orderer*

*docker run -d -p 7050:7050 \*

*--name orderer2 \*

*--health-cmd='wget --no-verbose --tries=1 --spider http://localhost:8443/healthz || exit 1 ' \*

*--health-interval=2s \*

*--restart always \*

*-e FABRIC\_LOGGING\_SPEC='INFO' \*

*-e ORDERER\_OPERATIONS\_LISTENADDRESS='localhost:8443' \*

*-e ORDERER\_METRICS\_PROVIDER='prometheus' \*

*-e ORDERER\_GENERAL\_GENESISFILE='/var/hyperledger/orderer/genesis.block' \*

*-e ORDERER\_GENERAL\_BOOTSTRAPMETHOD='file' \*

*-e ORDERER\_CHANNELPARTICIPATION\_ENABLED='true' \*

*-e ORDERER\_GENERAL\_LISTENADDRESS='0.0.0.0' \*

*-e ORDERER\_GENERAL\_LISTENPORT='7050' \*

*-e ORDERER\_GENERAL\_LOCALMSPDIR='/var/hyperledger/orderer/msp' \*

*-e ORDERER\_GENERAL\_LOCALMSPID='GuardOrdererMSP' \*

*-e ORDERER\_GENERAL\_TLS\_CERTIFICATE='/var/hyperledger/orderer/tls/signcerts/cert.pem' \*

*-e ORDERER\_GENERAL\_TLS\_ENABLED='true' \*

*-e ORDERER\_GENERAL\_TLS\_PRIVATEKEY='/var/hyperledger/orderer/tls/keystore/priv\_sk' \*

*-e ORDERER\_GENERAL\_TLS\_ROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_GENERAL\_CLUSTER\_CLIENTCERTIFICATE='/var/hyperledger/orderer/tls/signcerts/cert.pem' \*

*-e ORDERER\_GENERAL\_CLUSTER\_CLIENTPRIVATEKEY='/var/hyperledger/orderer/tls/keystore/priv\_sk' \*

*-e ORDERER\_GENERAL\_CLUSTER\_ROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_ADMIN\_TLS\_CERTIFICATE='/var/hyperledger/orderer/tls/signcerts/cert.pem' \*

*-e ORDERER\_ADMIN\_TLS\_ENABLED='true' \*

*-e ORDERER\_ADMIN\_TLS\_PRIVATEKEY='/var/hyperledger/orderer/tls/keystore/priv\_sk' \*

*-e ORDERER\_ADMIN\_TLS\_ROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_ADMIN\_TLS\_CLIENTROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_ADMIN\_LISTENADDRESS='localhost:9050' \*

*-v ~/orderer2-data:/var/hyperledger/production \*

*-v ~/crypto-config/ordererOrganizations/guard-project.eu/orderers/orderer2:/var/hyperledger/orderer \*

*hyperledger/fabric-orderer:amd64-2.3.1 orderer*

*docker run -d -p 7050:7050 \*

*--name orderer3 \*

*--health-cmd='wget --no-verbose --tries=1 --spider http://localhost:8443/healthz || exit 1 ' \*

*--health-interval=2s \*

*--restart always \*

*-e FABRIC\_LOGGING\_SPEC='INFO' \*

*-e ORDERER\_OPERATIONS\_LISTENADDRESS='localhost:8443' \*

*-e ORDERER\_METRICS\_PROVIDER='prometheus' \*

*-e ORDERER\_GENERAL\_GENESISFILE='/var/hyperledger/orderer/genesis.block' \*

*-e ORDERER\_GENERAL\_BOOTSTRAPMETHOD='file' \*

*-e ORDERER\_CHANNELPARTICIPATION\_ENABLED='true' \*

*-e ORDERER\_GENERAL\_LISTENADDRESS='0.0.0.0' \*

*-e ORDERER\_GENERAL\_LISTENPORT='7050' \*

*-e ORDERER\_GENERAL\_LOCALMSPDIR='/var/hyperledger/orderer/msp' \*

*-e ORDERER\_GENERAL\_LOCALMSPID='GuardOrdererMSP' \*

*-e ORDERER\_GENERAL\_TLS\_CERTIFICATE='/var/hyperledger/orderer/tls/signcerts/cert.pem' \*

*-e ORDERER\_GENERAL\_TLS\_ENABLED='true' \*

*-e ORDERER\_GENERAL\_TLS\_PRIVATEKEY='/var/hyperledger/orderer/tls/keystore/priv\_sk' \*

*-e ORDERER\_GENERAL\_TLS\_ROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_GENERAL\_CLUSTER\_CLIENTCERTIFICATE='/var/hyperledger/orderer/tls/signcerts/cert.pem' \*

*-e ORDERER\_GENERAL\_CLUSTER\_CLIENTPRIVATEKEY='/var/hyperledger/orderer/tls/keystore/priv\_sk' \*

*-e ORDERER\_GENERAL\_CLUSTER\_ROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_ADMIN\_TLS\_CERTIFICATE='/var/hyperledger/orderer/tls/signcerts/cert.pem' \*

*-e ORDERER\_ADMIN\_TLS\_ENABLED='true' \*

*-e ORDERER\_ADMIN\_TLS\_PRIVATEKEY='/var/hyperledger/orderer/tls/keystore/priv\_sk' \*

*-e ORDERER\_ADMIN\_TLS\_ROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_ADMIN\_TLS\_CLIENTROOTCAS='[/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem]' \*

*-e ORDERER\_ADMIN\_LISTENADDRESS='localhost:9050' \*

*-v ~/orderer3-data:/var/hyperledger/production \*

*-v ~/crypto-config/ordererOrganizations/guard-project.eu/orderers/orderer3:/var/hyperledger/orderer \*

*hyperledger/fabric-orderer:amd64-2.3.1 orderer*

1. **For each ordererX folder copy admincerts with certificates from folder crypto-config/peersOrganizations/guard-project.eu/msp/admincerts/**
2. **For each peerX/msp/cacerts rename to `peers-msp-cert.pem`**
3. **For each peerX/tls/tlscacerts rename to `peers-tlsca-cert.pem`**
4. **Copy core.yaml file from builders-config folder to `peerX` folder**
5. **Copy all from builders-config/bin to ~/builders**
6. **Copy peers-config.yaml to `peerX/msp` and rename it `config.yaml`, remove content from vm.endpoint section**
7. **Add execute permission to builders/\* catalog:**

*chmod +x builders/\**

1. **For each VMs run docker “peer” containers**

*docker run -d -p 8050:8050 \*

*--name peer1 \*

*--health-cmd='wget --no-verbose --tries=1 --spider http://localhost:9443/healthz || exit 1 ' \*

*--health-interval=2s \*

*--restart always \*

*-e FABRIC\_LOGGING\_SPEC='DEBUG' \*

*-e CORE\_PEER\_LISTENADDRESS='0.0.0.0:8050' \*

*-e CORE\_PEER\_ADDRESS='peer1.germanywestcentral.cloudapp.azure.com:8050' \*

*-e CORE\_PEER\_GOSSIP\_EXTERNALENDPOINT='peer1.germanywestcentral.cloudapp.azure.com:8050' \*

*-e CORE\_PEER\_CHAINCODELISTENADDRESS='localhost:7052' \*

*-e CORE\_PEER\_GOSSIP\_USELEADERELECTION='true' \*

*-e CORE\_PEER\_GOSSIP\_BOOTSTRAP='peer2.germanywestcentral.cloudapp.azure.com:8050 peer3.germanywestcentral.cloudapp.azure.com:8050' \*

*-e CORE\_PEER\_ID='peer1' \*

*-e FABRIC\_CFG\_PATH='/var/hyperledger/fabric' \*

*-e CORE\_PEER\_LOCALMSPID='GuardMSP' \*

*-e CORE\_PEER\_PROFILE\_ENABLED='true' \*

*-e CORE\_PEER\_TLS\_CERT\_FILE='/var/hyperledger/fabric/tls/signcerts/cert.pem' \*

*-e CORE\_PEER\_TLS\_ENABLED='true' \*

*-e CORE\_PEER\_TLS\_KEY\_FILE='/var/hyperledger/fabric/tls/keystore/priv\_sk' \*

*-e CORE\_PEER\_TLS\_ROOTCERT\_FILE='/var/hyperledger/fabric/tls/tlscacerts/peers-tlsca-cert.pem' \*

*-e CORE\_OPERATIONS\_LISTENADDRESS='localhost:9443' \*

*-e CORE\_METRICS\_PROVIDER='prometheus' \*

*-v ~/peer1-data:/var/hyperledger/production \*

*-v ~/crypto-config/peerOrganizations/guard-project.eu/peers/peer1:/var/hyperledger/fabric \*

*-v ~/builders:/builders/external/bin \*

*hyperledger/fabric-peer:amd64-2.3.1 peer node start*

*docker run -d -p 8050:8050 \*

*--name peer2 \*

*--health-cmd='wget --no-verbose --tries=1 --spider http://localhost:9443/healthz || exit 1 ' \*

*--health-interval=2s \*

*--restart always \*

*-e FABRIC\_LOGGING\_SPEC='INFO' \*

*-e CORE\_PEER\_LISTENADDRESS='0.0.0.0:8050' \*

*-e CORE\_PEER\_ADDRESS='peer2.germanywestcentral.cloudapp.azure.com:8050' \*

*-e CORE\_PEER\_GOSSIP\_EXTERNALENDPOINT='peer2.germanywestcentral.cloudapp.azure.com:8050' \*

*-e CORE\_PEER\_CHAINCODELISTENADDRESS='localhost:7052' \*

*-e CORE\_PEER\_GOSSIP\_USELEADERELECTION='true' \*

*-e CORE\_PEER\_GOSSIP\_BOOTSTRAP='peer1.germanywestcentral.cloudapp.azure.com:8050 peer3.germanywestcentral.cloudapp.azure.com:8050' \*

*-e CORE\_PEER\_ID='peer2' \*

*-e FABRIC\_CFG\_PATH='/var/hyperledger/fabric' \*

*-e CORE\_PEER\_LOCALMSPID='GuardMSP' \*

*-e CORE\_PEER\_PROFILE\_ENABLED='true' \*

*-e CORE\_PEER\_TLS\_CERT\_FILE='/var/hyperledger/fabric/tls/signcerts/cert.pem' \*

*-e CORE\_PEER\_TLS\_ENABLED='true' \*

*-e CORE\_PEER\_TLS\_KEY\_FILE='/var/hyperledger/fabric/tls/keystore/priv\_sk' \*

*-e CORE\_PEER\_TLS\_ROOTCERT\_FILE='/var/hyperledger/fabric/tls/tlscacerts/peers-tlsca-cert.pem' \*

*-e CORE\_OPERATIONS\_LISTENADDRESS='localhost:9443' \*

*-e CORE\_METRICS\_PROVIDER='prometheus' \*

*-v ~/peer1-data:/var/hyperledger/production \*

*-v ~/crypto-config/peerOrganizations/guard-project.eu/peers/peer2:/var/hyperledger/fabric \*

*-v ~/builders:/builders/external/bin \*

*hyperledger/fabric-peer:amd64-2.3.1 peer node start*

*docker run -d -p 8050:8050 \*

*--name peer3 \*

*--health-cmd='wget --no-verbose --tries=1 --spider http://localhost:9443/healthz || exit 1 ' \*

*--health-interval=2s \*

*--restart always \*

*-e FABRIC\_LOGGING\_SPEC='INFO' \*

*-e CORE\_PEER\_LISTENADDRESS='0.0.0.0:8050' \*

*-e CORE\_PEER\_ADDRESS='peer3.germanywestcentral.cloudapp.azure.com:8050' \*

*-e CORE\_PEER\_GOSSIP\_EXTERNALENDPOINT='peer3.germanywestcentral.cloudapp.azure.com:8050' \*

*-e CORE\_PEER\_CHAINCODELISTENADDRESS='localhost:7052' \*

*-e CORE\_PEER\_GOSSIP\_USELEADERELECTION='true' \*

*-e CORE\_PEER\_GOSSIP\_BOOTSTRAP='peer1.germanywestcentral.cloudapp.azure.com:8050 peer2.germanywestcentral.cloudapp.azure.com:8050' \*

*-e CORE\_PEER\_ID='peer3' \*

*-e FABRIC\_CFG\_PATH='/var/hyperledger/fabric' \*

*-e CORE\_PEER\_LOCALMSPID='GuardMSP' \*

*-e CORE\_PEER\_PROFILE\_ENABLED='true' \*

*-e CORE\_PEER\_TLS\_CERT\_FILE='/var/hyperledger/fabric/tls/signcerts/cert.pem' \*

*-e CORE\_PEER\_TLS\_ENABLED='true' \*

*-e CORE\_PEER\_TLS\_KEY\_FILE='/var/hyperledger/fabric/tls/keystore/priv\_sk' \*

*-e CORE\_PEER\_TLS\_ROOTCERT\_FILE='/var/hyperledger/fabric/tls/tlscacerts/peers-tlsca-cert.pem' \*

*-e CORE\_OPERATIONS\_LISTENADDRESS='localhost:9443' \*

*-e CORE\_METRICS\_PROVIDER='prometheus' \*

*-v ~/peer1-data:/var/hyperledger/production \*

*-v ~/crypto-config/peerOrganizations/guard-project.eu/peers/peer3:/var/hyperledger/fabric \*

*-v ~/builders:/builders/external/bin \*

*hyperledger/fabric-peer:amd64-2.3.1 peer node start*

1. **Copy chaincode folder to home folder**

*cd chaincode/src*

*docker build -t chaincode/storage-chaincode:1.0*

1. **Pack chaincode image**

*cd ../packaging/*

*tar cfz code.tar.gz connection.json*

*tar cfz storage-chaincode-guard.tgz code.tar.gz metadata.json*

1. **Rename /home/aleksander/crypto-config/peerOrganizations/guard-project.eu/users/peers-admin/tls/tlscacerts to peers-tlsca-cert.pem**
2. **Rename /home/aleksander/crypto-config/ordererOrganizations/guard-project.eu/users/orderers-admin/tls/tlscacerts to orderers-tlsca-cert.pem**
3. **Rename /home/aleksander/crypto-config/peerOrganizations/guard-project.eu/users/peers-admin/msp/cacerts to peers-msp-cert.pem**
4. **Copy peers-config.yaml file to /home/aleksander/crypto-config/peerOrganizations/guard-project.eu/users/peers-admin/msp and rename it to config.yaml**
5. **Run docker “peers-cli” container:**

*docker run -it --rm \*

*--name peers-cli \*

*-e FABRIC\_LOGGING\_SPEC='INFO' \*

*-e CORE\_PEER\_ID='peers-cli' \*

*-e CORE\_PEER\_LOCALMSPID='GuardMSP' \*

*-e CORE\_PEER\_MSPCONFIGPATH='/var/hyperledger/admin/msp' \*

*-e CORE\_PEER\_TLS\_CERT\_FILE='/var/hyperledger/fabric/tls/signcerts/cert.pem' \*

*-e CORE\_PEER\_TLS\_ENABLED='true' \*

*-e CORE\_PEER\_TLS\_KEY\_FILE='/var/hyperledger/fabric/tls/keystore/priv\_sk' \*

*-e CORE\_PEER\_TLS\_ROOTCERT\_FILE='/var/hyperledger/fabric/tls/tlscacerts/peers-tlsca-cert.pem' \*

*-e GOPATH='/opt/gopath' \*

*-e ORDERER\_CA='/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem' \*

*-v ~/crypto-config/peerOrganizations/guard-project.eu/users/peers-admin/tls:/var/hyperledger/fabric/tls \*

*-v ~/crypto-config/peerOrganizations/guard-project.eu/users/peers-admin/msp:/var/hyperledger/admin/msp \*

*-v ~/crypto-config/ordererOrganizations/guard-project.eu/users/orderers-admin/tls/tlscacerts:/var/hyperledger/orderer/tls/tlscacerts \*

*-v ~/channel-artifacts:/channel-artifacts \*

*-v ~/chaincode/packaging:/chaincode \*

*hyperledger/fabric-tools:amd64-2.3.1 /bin/bash*

1. **Execute commands inside “peers-cli” container:**

*CORE\_PEER\_ADDRESS=10.1.0.4:8050 peer channel create -o 10.1.0.4:7050 -c guard -f /channel-artifacts/channel.tx --tls true --cafile $ORDERER\_CA*

*CORE\_PEER\_ADDRESS=10.1.0.4:8050 peer channel join -b guard.block*

*CORE\_PEER\_ADDRESS=10.1.0.6:8050 peer channel join -b guard.block*

*CORE\_PEER\_ADDRESS=10.1.0.8:8050 peer channel join -b guard.block*

*CORE\_PEER\_ADDRESS=10.1.0.4:8050 peer lifecycle chaincode install /chaincode/storage-chaincode-guard.tgz*

*storage-chaincode:bbe96219374addcb3b9fb5c8060c9e08462cf61cb967e313cee35ebf42ed6812*

*exit*

1. **Run docker “chaincode-storage” container:**

*docker run -d -p 7052:7052 \*

*--name chaincode-storage \*

*--restart always \*

*-e CHAINCODE\_CCID='storage-chaincode:bbe96219374addcb3b9fb5c8060c9e08462cf61cb967e313cee35ebf42ed6812' \*

*-e CHAINCODE\_ADDRESS='0.0.0.0:7052' \*

*chaincode/storage-chaincode:1.0*

1. **Execute commands inside “chaincode-storage” container:**

*CORE\_PEER\_ADDRESS=10.1.0.4:8050 peer lifecycle chaincode approveformyorg --channelID guard --name storage-chaincode --version 1.0 --init-required --package-id 'storage-chaincode:bbe96219374addcb3b9fb5c8060c9e08462cf61cb967e313cee35ebf42ed6812' --sequence 1 -o 10.1.0.4:7050 --tls --cafile $ORDERER\_CA*

*CORE\_PEER\_ADDRESS=10.1.0.4:8050 peer lifecycle chaincode commit -o 10.1.0.4:7050 --channelID guard --name storage-chaincode --version 1.0 --sequence 1 --init-required --tls true --cafile $ORDERER\_CA --peerAddresses 10.1.0.4:8050 --tlsRootCertFiles $CORE\_PEER\_TLS\_ROOTCERT\_FILE*

*CORE\_PEER\_ADDRESS=10.1.0.4:8050 peer chaincode invoke -o 10.1.0.4:7050 --isInit --tls true --cafile $ORDERER\_CA -C guard -n storage-chaincode --peerAddresses 10.1.0.4:8050 --tlsRootCertFiles $CORE\_PEER\_TLS\_ROOTCERT\_FILE -c '{"Args":[""]}' --waitForEvent*

*CORE\_PEER\_ADDRESS=peer1.germanywestcentral.cloudapp.azure.com:8050 peer chaincode invoke -o orderer1.germanywestcentral.cloudapp.azure.com:7050 --tls true --cafile $ORDERER\_CA -C guard -n storage-chaincode --peerAddresses peer1.germanywestcentral.cloudapp.azure.com:8050 --tlsRootCertFiles $CORE\_PEER\_TLS\_ROOTCERT\_FILE -c '{"Args":["save", "test"]}' --waitForEvent*

*3df12a6881f78325f3cc01816dc0a1d67dda300e800d4f41f113ea7620ed35bc*

*CORE\_PEER\_ADDRESS=peer1.germanywestcentral.cloudapp.azure.com:8050 peer chaincode invoke -o orderer1.germanywestcentral.cloudapp.azure.com:7050 --tls true --cafile $ORDERER\_CA -C guard -n storage-chaincode --peerAddresses peer1.germanywestcentral.cloudapp.azure.com:8050 --tlsRootCertFiles $CORE\_PEER\_TLS\_ROOTCERT\_FILE -c '{"Args":["retrieve", "3df12a6881f78325f3cc01816dc0a1d67dda300e800d4f41f113ea7620ed35bc"]}' --waitForEvent*

*CORE\_PEER\_ADDRESS=10.1.0.6:8050 peer lifecycle chaincode install /chaincode/storage-chaincode-guard.tgz*

*CORE\_PEER\_ADDRESS=10.1.0.8:8050 peer lifecycle chaincode install /chaincode/storage-chaincode-guard.tgz*

1. **Run docker “orderers-cli” containers:**

*docker run -it --rm \*

*--name orderers-cli \*

*-e FABRIC\_LOGGING\_SPEC='INFO' \*

*-e CORE\_PEER\_ID='orderers-cli' \*

*-e CORE\_PEER\_LOCALMSPID='GuardOrdererMSP' \*

*-e CORE\_PEER\_MSPCONFIGPATH='/var/hyperledger/admin/msp' \*

*-e CORE\_PEER\_TLS\_CERT\_FILE='/var/hyperledger/fabric/tls/signcerts/cert.pem' \*

*-e CORE\_PEER\_TLS\_ENABLED='true' \*

*-e CORE\_PEER\_TLS\_KEY\_FILE='/var/hyperledger/fabric/tls/keystore/priv\_sk' \*

*-e CORE\_PEER\_TLS\_ROOTCERT\_FILE='/var/hyperledger/fabric/tls/tlscacerts/orderers-tlsca-cert.pem' \*

*-e GOPATH='/opt/gopath' \*

*-e ORDERER\_CA='/var/hyperledger/orderer/tls/tlscacerts/orderers-tlsca-cert.pem' \*

*-v ~/crypto-config/ordererOrganizations/guard-project.eu/users/orderers-admin/tls:/var/hyperledger/fabric/tls \*

*-v ~/crypto-config/ordererOrganizations/guard-project.eu/users/orderers-admin/msp:/var/hyperledger/admin/msp \*

*-v ~/crypto-config/ordererOrganizations/guard-project.eu/users/orderers-admin/tls/tlscacerts:/var/hyperledger/orderer/tls/tlscacerts \*

*-v ~/channel-artifacts:/channel-artifacts \*

*-v ~/chaincode/packaging:/chaincode \*

*hyperledger/fabric-tools:amd64-2.3.1 /bin/bash*

1. **Execute commands inside “orderers-cli” container:**

*CORE\_PEER\_ADDRESS=10.1.0.4:8050 peer channel fetch config system\_config\_block.pb -o 10.1.0.4:7050 -c system --tls --cafile $ORDERER\_CA*

*configtxlator proto\_decode --input system\_config\_block.pb --type common.Block --output system\_config\_block.json*

*jq .data.data[0].payload.data.config system\_config\_block.json > system\_config.json*

*configtxlator proto\_encode --input system\_config.json --type common.Config --output system\_config.pb*

*configtxlator proto\_encode --input modified\_system\_config.json --type common.Config --output modified\_system\_config.pb*

*configtxlator compute\_update --channel\_id system --original system\_config.pb --updated modified\_system\_config.pb --output system\_config\_update.pb*

*configtxlator proto\_decode --input system\_config\_update.pb --type common.ConfigUpdate --output system\_config\_update.json*

*echo '{"payload":{"header":{"channel\_header":{"channel\_id":"'system'", "type":2}},"data":{"config\_update":'$(cat system\_config\_update.json)'}}}' | jq . > system\_config\_update\_in\_envelope.json*

*configtxlator proto\_encode --input system\_config\_update\_in\_envelope.json --type common.Envelope --output system\_config\_update\_in\_envelope.pb*

*CORE\_PEER\_ADDRESS=10.1.0.4:8050 peer channel update -f system\_config\_update\_in\_envelope.pb -o 10.1.0.4:7050 -c system --tls --cafile $ORDERER\_CA*

*##### --name orderers-cli \*

*CORE\_PEER\_ADDRESS=10.1.0.4:8050 peer channel fetch config config\_block.pb -o 10.1.0.4:7050 -c guard --tls --cafile $ORDERER\_CA*

*configtxlator proto\_decode --input config\_block.pb --type common.Block --output config\_block.json*

*jq .data.data[0].payload.data.config config\_block.json > config.json*

*configtxlator proto\_encode --input config.json --type common.Config --output config.pb*

*configtxlator proto\_encode --input modified\_config.json --type common.Config --output modified\_config.pb*

*configtxlator compute\_update --channel\_id guard --original config.pb --updated modified\_config.pb --output config\_update.pb*

*configtxlator proto\_decode --input config\_update.pb --type common.ConfigUpdate --output config\_update.json*

*echo '{"payload":{"header":{"channel\_header":{"channel\_id":"'guard'", "type":2}},"data":{"config\_update":'$(cat config\_update.json)'}}}' | jq . > config\_update\_in\_envelope.json*

*configtxlator proto\_encode --input config\_update\_in\_envelope.json --type common.Envelope --output config\_update\_in\_envelope.pb*

*CORE\_PEER\_ADDRESS=10.1.0.4:8050 peer channel update -f config\_update\_in\_envelope.pb -o 10.1.0.4:7050 -c guard --tls --cafile $ORDERER\_CA*

1. **Build blockchain-connector**

*cd blockchain-connector*

*mvn clean package*

*docker rmi blockchain-connector:1.0*

*docker build -t blockchain-connector:1.0 .*

*cd ..*

1. **Run docker “blockchain-connector” container**

*docker run -d -p 8080:8080 \*

*--name blockchain-connector \*

*--health-cmd='wget --no-verbose --tries=1 --spider http://localhost:8080/health || exit 1 ' \*

*--health-interval=2s \*

*--restart always \*

*-v ${PWD}/application.properties:/application.properties \*

*-v ${PWD}/crypto-materials:/crypto-materials \*

*blockchain-connector:1.0*

1. **Testing blockchain with sample transactions:**

*{*

*"txId": "63f6e0ddf5824d68413215932477b4735af92db7c006860ca0ceeaa4a194523a",*

*"key": "da093e65b29f32c241ee3b14db4821ff5fd28e18be61d540be756f3abd872f3f"*

*}*

*{"test":"test"}*

*---------------------------------------------------*

*{"test2":"test2"}*

*{*

*"txId": "2535f87ceaa4ffa81f61bdef1a499c3ac830960ce2e1d8d97b8afdf2079d4d8c",*

*"key": "232173463b59d7b7022ecc71a307ffb92db9b64752e1be34ebb33a3485800cc5"*

*}*

*---------------------------------------------------*