## Build ear image

1. Create emm ear image

|  |
| --- |
| apiVersion: image.openshift.io/v1  kind: ImageStream  metadata:  name: myemm-ear  namespace: mas-inst8809-manage |

1. Create emm ear build config

|  |
| --- |
| kind: BuildConfig  apiVersion: build.openshift.io/v1  metadata:  name: myemm-ear-build-config  namespace: mas-inst8809-manage  spec:  nodeSelector: null  output:  to:  kind: ImageStreamTag  name: 'myemm-ear:v1'  resources:  limits:  ephemeral-storage: 100Gi  requests:  ephemeral-storage: 30Gi  successfulBuildsHistoryLimit: 5  failedBuildsHistoryLimit: 5  strategy:  type: Docker  dockerStrategy:  pullSecret:  name: ibm-entitlement  forcePull: true  postCommit: {}  source:  type: Dockerfile  dockerfile: >  FROM cp.icr.io/cp/manage/manageadmin:8.4.5 AS ADMIN  WORKDIR /opt/IBM/SMP/maximo  RUN mkdir -p /opt/IBM/SMP/maximo/additional-server-files  COPY --chown=maximoinstall:0 tmp\_build\_files .  RUN rm -rf customizationCredentials && rm trust.p12 && rm  truststorePassword  # Remove translation files that will not be used.  WORKDIR /opt/IBM/SMP/maximo  RUN rm -f translation\_files.zip lang/MaximoLangPkgXliff\_Ar.zip  lang/MaximoLangPkgXliff\_Cs.zip lang/MaximoLangPkgXliff\_Da.zip  lang/MaximoLangPkgXliff\_De.zip lang/MaximoLangPkgXliff\_Es.zip  lang/MaximoLangPkgXliff\_Fi.zip lang/MaximoLangPkgXliff\_Fr.zip  lang/MaximoLangPkgXliff\_He.zip lang/MaximoLangPkgXliff\_Hr.zip  lang/MaximoLangPkgXliff\_Hu.zip lang/MaximoLangPkgXliff\_It.zip  lang/MaximoLangPkgXliff\_Ja.zip lang/MaximoLangPkgXliff\_Ko.zip  lang/MaximoLangPkgXliff\_Nl.zip lang/MaximoLangPkgXliff\_No.zip  lang/MaximoLangPkgXliff\_Pl.zip lang/MaximoLangPkgXliff\_Pt\_BR.zip  lang/MaximoLangPkgXliff\_Ru.zip lang/MaximoLangPkgXliff\_Sk.zip  lang/MaximoLangPkgXliff\_Sl.zip lang/MaximoLangPkgXliff\_Sv.zip  lang/MaximoLangPkgXliff\_Tr.zip lang/MaximoLangPkgXliff\_Zh\_CN.zip  lang/MaximoLangPkgXliff\_Zh\_TW.zip  RUN rm -rf tools/maximo/ar tools/maximo/cs tools/maximo/da tools/maximo/de  tools/maximo/es tools/maximo/fi tools/maximo/fr tools/maximo/he  tools/maximo/hr tools/maximo/hu tools/maximo/it tools/maximo/ja  tools/maximo/ko tools/maximo/nl tools/maximo/no tools/maximo/pl  tools/maximo/pt tools/maximo/ru tools/maximo/sk tools/maximo/sl  tools/maximo/sv tools/maximo/tr tools/maximo/zh tools/maximo/zht  WORKDIR /opt/IBM/SMP/maximo/tools/maximo  RUN \  mkdir -p /opt/IBM/SMP/maximo/applications/maximo/businessobjects/classes/psdi/app/signature/apps &&\  mkdir -p /opt/IBM/SMP/maximo/tools/maximo/log &&\  ./pkginstall.sh && ./updatedblitepreprocessor.sh -disconnected &&\  find /opt/IBM/SMP/maximo/applications -type d -exec chmod 777 {} + &&\  chmod ugo+rw -R /opt/IBM/SMP/maximo/applications &&\  find /opt/IBM/SMP/maximo/tools/maximo -type d -exec chmod 777 {} + &&\  chmod ugo+rw -R /opt/IBM/SMP/maximo/tools/maximo &&\  chmod 777 /opt/IBM/SMP/maximo/tools/maximo/log &&\  find /opt/IBM/SMP/maximo/tools/maximo/en -type f -name postupdatedb.sh -exec chmod -v 777 {} +  WORKDIR /opt/IBM/SMP/maximo/deployment/was-liberty-default  RUN ./maximo-all.sh && ./buildmaximoui-war.sh && ./buildmaximo-xwar.sh &&  ./maximo-cron.sh  ENV MXE\_MASDEPLOYED=1  ENV MXE\_USESQLSERVERSEQUENCE=1  ENV LC\_ALL=en\_US.UTF-8  ENV LANGUAGE\_BASE=en  ENV LANGUAGE\_ADD=  WORKDIR /opt/IBM/SMP/maximo/tools/maximo  WORKDIR /opt/IBM/SMP  RUN \  wget -c http://124.222.21.115:3000/ezmaxmobile.zip &&\  unzip ezmaxmobile.zip &&\  cd ezmaxmobile &&\  chmod u+x ./buildemmear.sh &&\  ./buildemmear.sh &&\  ls -al default &&\  ls -al was-liberty-default &&\  ls -al was-liberty-default/emm-server/apps &&\  ls -al /opt/IBM/SMP/maximo/deployment/was-liberty-default/deployment/maximo-all/maximo-all-server &&\  ls -al /opt/IBM/SMP/maximo/additional-server-files  WORKDIR /opt/IBM/SMP/maximo/tools/maximo  #CMD sleep 123456789  secrets:  - secret:  name: inst8809-masdev-manage-truststorepasswd  destinationDir: tmp\_build\_files  configMaps:  - configMap:  name: inst8809-masdev-truststore-cfg  destinationDir: tmp\_build\_files  runPolicy: Serial |

1. Build emm ear image

## Build liberty image

1. Create liberty image

|  |
| --- |
| apiVersion: image.openshift.io/v1  kind: ImageStream  metadata:  name: myemm-liberty  namespace: mas-inst8809-manage |

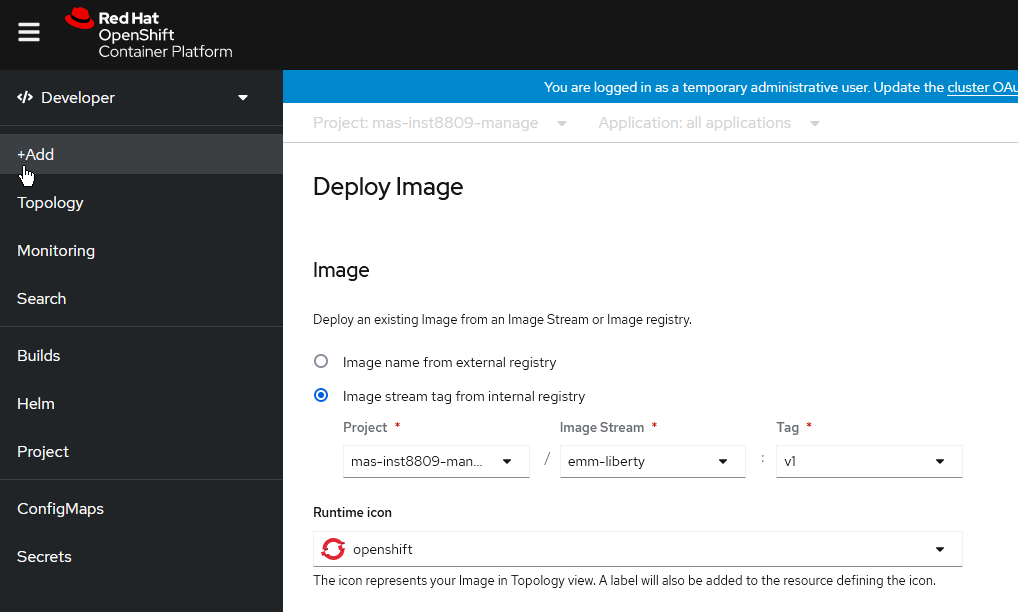
1. Create liberty build config

|  |
| --- |
| apiVersion: build.openshift.io/v1  kind: BuildConfig  metadata:  name: myemm-liberty-build-config  namespace: mas-inst8809-manage  spec:  nodeSelector: null  output:  to:  kind: ImageStreamTag  name: 'myemm-liberty:v1'  resources:  limits:  ephemeral-storage: 50Gi  requests:  ephemeral-storage: 10Gi  successfulBuildsHistoryLimit: 5  failedBuildsHistoryLimit: 5  strategy:  type: Docker  dockerStrategy:  pullSecret:  name: ibm-entitlement  forcePull: true  postCommit: {}  source:  type: Dockerfile  dockerfile: >  FROM cp.icr.io/cp/manage/ubi-wlp-manage:2.2.13 AS LIBERTY  #ARG VERBOSE=true  COPY --chown=1001:0 emm-server/apps/ /config/dropins  RUN rm /opt/ibm/wlp/usr/servers/defaultServer/server.env && mkdir  /managefiles/additional-server-files  COPY --chown=1001:0 additional-server-files/  /managefiles/additional-server-files/  ENV MXE\_MASDEPLOYED=1  ENV MXE\_USESQLSERVERSEQUENCE=1  ENV LC\_ALL=en\_US.UTF-8  USER 1001  # This is overridden by container cmd when deployed in OpenShift  # CMD /opt/ibm/wlp/bin/server run defaultServer  CMD sleep 123456789  images:  - from:  kind: ImageStreamTag  name: 'myemm-ear:v1'  paths:  - sourcePath: >-  /opt/IBM/SMP/ezmaxmobile/was-liberty-default/emm-server  destinationDir: .  - sourcePath: /opt/IBM/SMP/maximo/additional-server-files  destinationDir: .  triggers:  - type: ImageChange  imageChange:  lastTriggeredImageID: >-  image-registry.openshift-image-registry.svc:5000/mas-inst8809-manage/myemm-ear@sha256:b047eb1cf0b5689cdcb724775ce255edd70401b7e510e0e712217a091fd6d1d3  from:  kind: ImageStreamTag  name: 'myemm-ear:v1'  runPolicy: Serial |

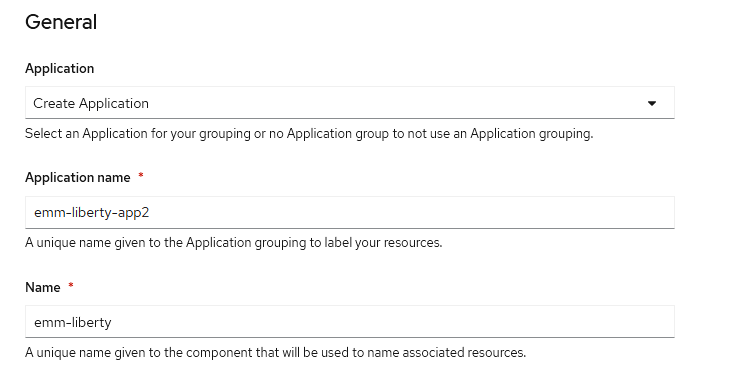
1. Build liberty image

## Deploy **Image**

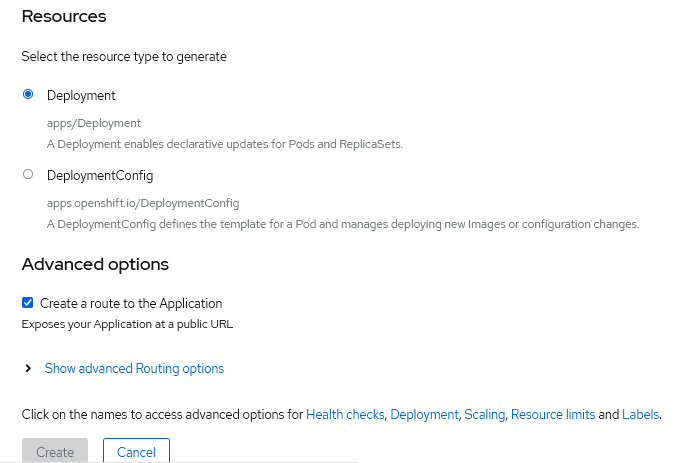
1. Login OCP and go to “Developer” view from the navigation
2. Select “Add” from navigation tree and then select the “Container images”
3. In the “ Deploy Image” page, fill the form like this
   1. Select the image from internal registry



* 1. Fill the application name



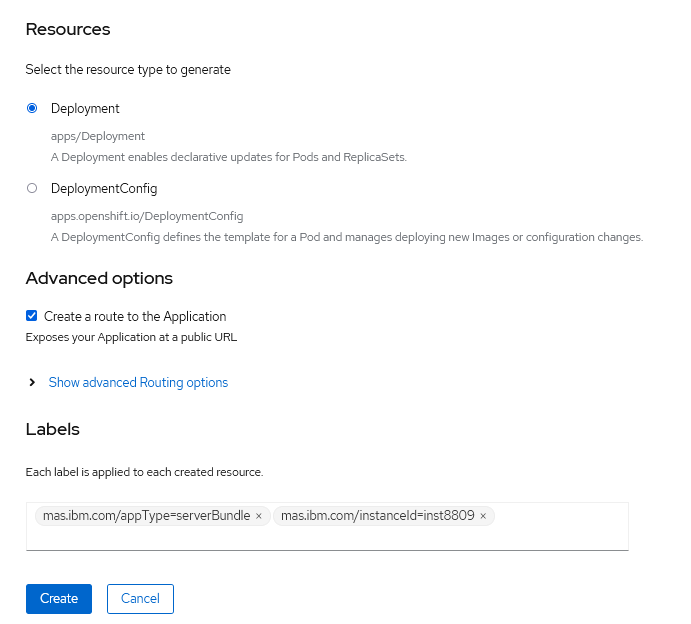
* 1. In the “Advanced options” click the “Labels” link



* 1. In the “Labels” section, add two labels:

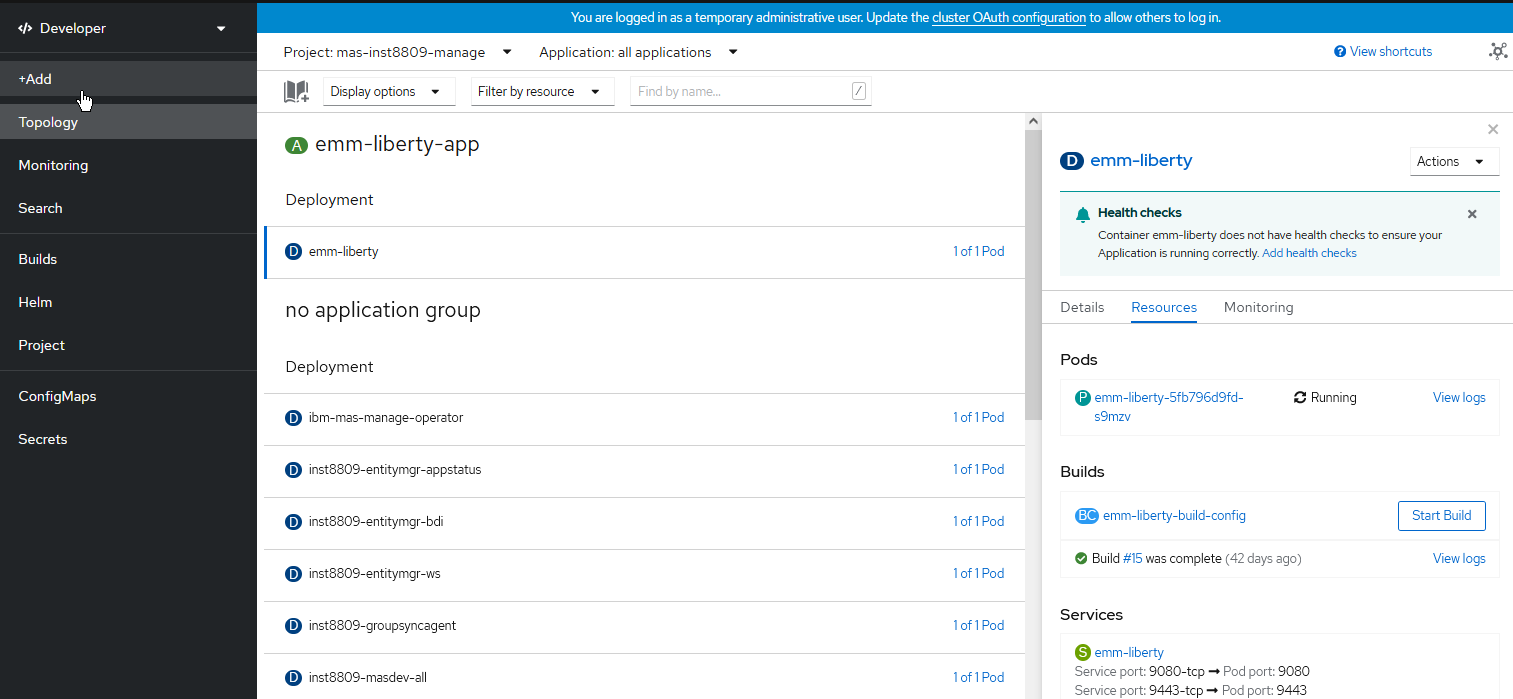
*mas.ibm.com/appType=serverBundle*

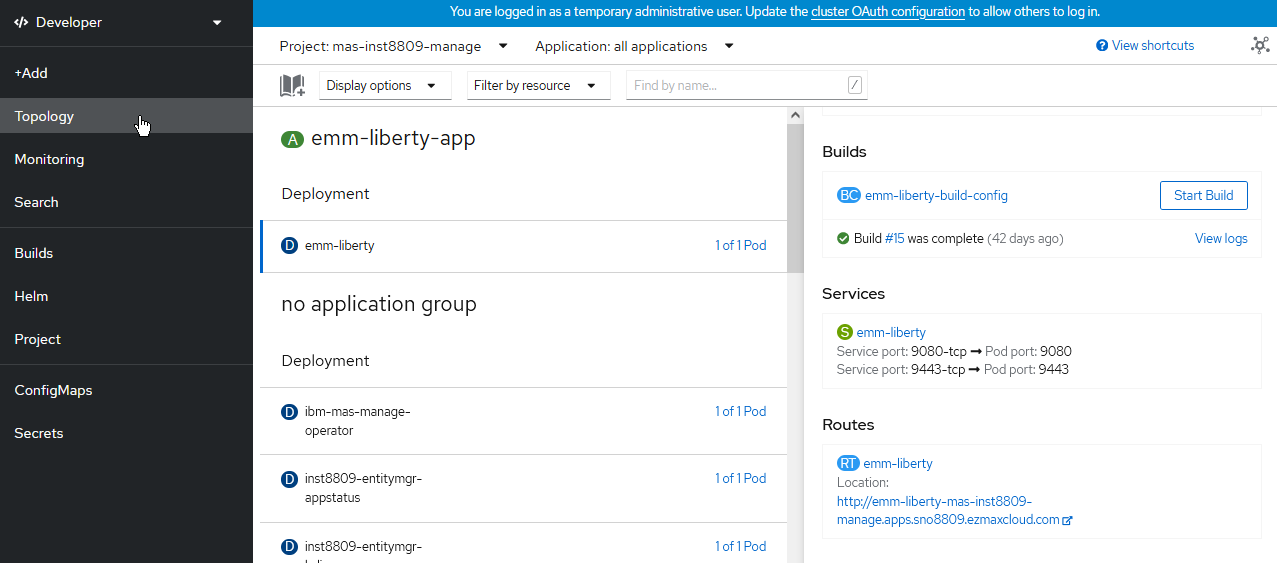
*mas.ibm.com/instanceId=inst8809*



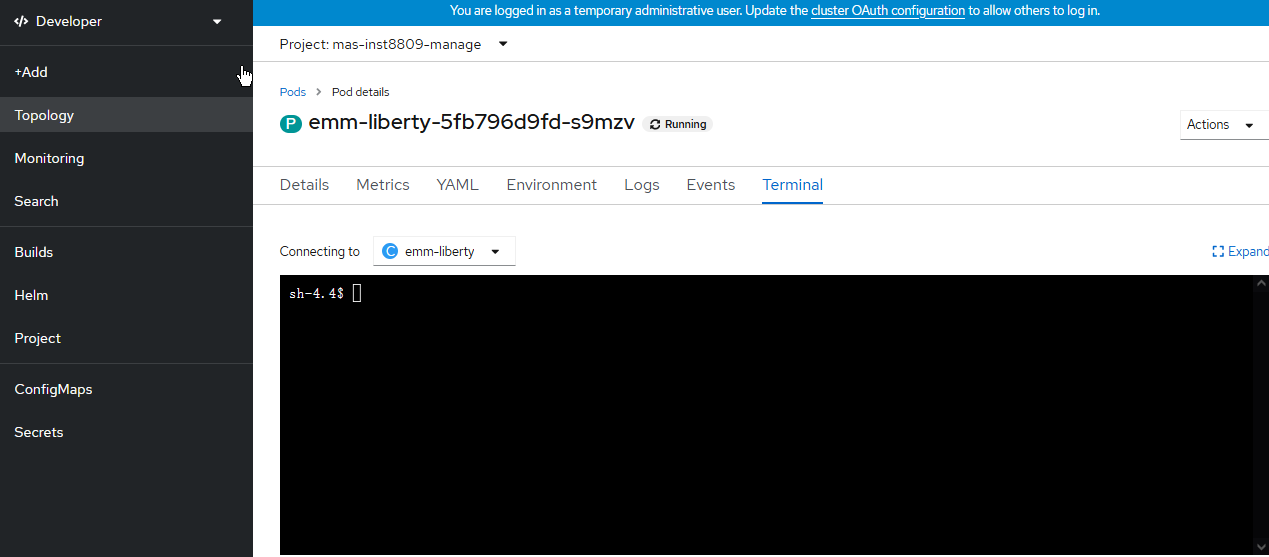
* 1. Select the ”Create” to create the development

1. Select “Topology” from navigation tree and select the development which created above





1. Go to the Terminal of the Pod



1. Unzip the ezmm\_liberty.zip to c:/temp folder



1. Copy the files to Pod

|  |
| --- |
| > oc login --token=<token> --server=https://api.sno8809.ezmaxcloud.com:6443  > kubectl cp /temp/server.xml mas-inst8809-manage/**<pod\_name>**:/opt/ibm/wlp/usr/servers/defaultServer/server.xml  > kubectl cp /temp/setenv.sh mas-inst8809-manage/**<pod\_name>**:/opt/ibm/wlp/usr/servers/defaultServer/setenv.sh  > kubectl cp /temp/jvm.options mas-inst8809-manage/**<pod\_name>**:/opt/ibm/wlp/usr/servers/defaultServer/jvm.options  > kubectl exec **<pod\_name>** -c **<contain\_name>** -it -- mkdir /opt/ibm/wlp/usr/servers/defaultServer/truststore  > kubectl exec **<pod\_name>** -c **<contain\_name>** -it -- mkdir /opt/ibm/wlp/usr/servers/defaultServer/managefiles  > kubectl cp /temp/truststore/trust.p12 mas-inst8809-manage/**<pod\_name>**:/opt/ibm/wlp/usr/servers/defaultServer/truststore/trust.p12  > kubectl cp /temp/managefiles/key.p12 mas-inst8809-manage/**<pod\_name>**:/opt/ibm/wlp/usr/servers/defaultServer/managefiles/key.p12  > kubectl cp /temp/managefiles/trust.p12 mas-inst8809-manage/**<pod\_name>**:/opt/ibm/wlp/usr/servers/defaultServer/managefiles/trust.p12  > kubectl exec **<pod\_name>** -c **<contain\_name>** -it -- mv /opt/ibm/wlp/usr/servers/defaultServer/dropins/emm.ear /opt/ibm/wlp/usr/servers/defaultServer/apps/emm.ear |

1. Edit the yaml of route to enable https

|  |
| --- |
| Sped:  port:  **targetPort: 9443-tcp**  tls:  termination: reencrypt  certificate: |  -----BEGIN CERTIFICATE-----  MIIDvTCCAqWgAwIBAgIQNNh4DOhNzriA8o39SXWLxTANBgkqhkiG9w0BAQsFADCB  hTELMAkGA1UEBhMCR0IxDzANBgNVBAcTBkxvbmRvbjEPMA0GA1UECRMGTG9uZG9u  MS4wLAYDVQQLEyVJQk0gTWF4aW1vIEFwcGxpY2F0aW9uIFN1aXRlIChQdWJsaWMp  MSQwIgYDVQQDExtwdWJsaWMuaW5zdDg4MDkubWFzLmlibS5jb20wHhcNMjIxMTAy  MjE1OTI1WhcNMjMxMTAyMjE1OTI1WjAAMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8A  MIIBCgKCAQEAqnHRlrJO9uBSFWUGY5E9mejUFiQeNb96KZoT34eqB3soPIChYENT  pz17krSmr6riBCJWO4iW+L5C/+YH1rkr8/GbUqa1ioDw8X84CESHlHP7LAJxq4Af  IOZUTSxfNUkGi/57nVVxJNpGSRJpNoMMp1m7mdifyAL3tRhxkWB0oLXgxAuCqwOr  9kAH8vCtQgPb5F1wJKvTgMGWIS+BfCYRFEB3M0e0+iCxxODTbNyeJjIfnhB5E7Xe  3V02yWGrxCyJh9nTZsRjHesrXksJlFrwZcEdP4oXEd75gIDOd9ZISV56O52+3V2Q  uApkL027LUPlkftjER8mEvU6uy8EniPnTQIDAQABo4GsMIGpMA4GA1UdDwEB/wQE  AwIFoDAMBgNVHRMBAf8EAjAAMB8GA1UdIwQYMBaAFNvb3GFcZ1HyEzYW3gQcgnow  n0yDMGgGA1UdEQEB/wReMFyCK21hbmFnZS5pbnN0ODgwOS5hcHBzLnNubzg4MDku  ZXptYXhjbG91ZC5jb22CLSoubWFuYWdlLmluc3Q4ODA5LmFwcHMuc25vODgwOS5l  em1heGNsb3VkLmNvbTANBgkqhkiG9w0BAQsFAAOCAQEAOJ2mKn9N++T3Ypi7waSJ  bnXgxUaPwSoFcNReM9s5fAZZ879PTlA8N2UbpqqvSDOBSvx82wBUt0G2eOVSEMjG  ga7KD7oUwKKUlFKkwWKLqXbdZ1VADRZX12B6B98dQ90hS4kuoXwzRWz8D5GZB1LH  WYiLOs4OQPbyGUmjZ/iBrAEjTbeHwSKNl/l9e4slK0KycKWAb876xep6m71C7CUG  08Uxzij0CV6YVWkve1c8D2Z56QgNavdvEWE13JEEE2NwMFdndmET8K7Al2okygT4  s+1zeprCWQmPxUcxpEXf4q2Jln7POqtGEaaItMhy/LZ+Fe68wSMtuS0L709ByyVS  8A==  -----END CERTIFICATE-----  key: |  -----BEGIN RSA PRIVATE KEY-----  MIIEpAIBAAKCAQEAqnHRlrJO9uBSFWUGY5E9mejUFiQeNb96KZoT34eqB3soPICh  YENTpz17krSmr6riBCJWO4iW+L5C/+YH1rkr8/GbUqa1ioDw8X84CESHlHP7LAJx  q4AfIOZUTSxfNUkGi/57nVVxJNpGSRJpNoMMp1m7mdifyAL3tRhxkWB0oLXgxAuC  qwOr9kAH8vCtQgPb5F1wJKvTgMGWIS+BfCYRFEB3M0e0+iCxxODTbNyeJjIfnhB5  E7Xe3V02yWGrxCyJh9nTZsRjHesrXksJlFrwZcEdP4oXEd75gIDOd9ZISV56O52+  3V2QuApkL027LUPlkftjER8mEvU6uy8EniPnTQIDAQABAoIBAHf7g2SnxHp5Uo8I  /MReW5Tal2G2ryV60s4yTkcOQxzdD2f3zKHtAxg5aaHOVYBXtJ7gZP+nmU8k8Bkf  nukClPQ2YfZ+sFD9j58QxSrcwuw38Nn8JnRpzn/kY/engqYJ7/fgfrg0i1874DpU  IYhkHCY6PpuAKIotvJ3YxSlnUvJFr/RxJ0LIYIwidXm0dteFbl12PJG2SwO3o0Xp  GM37sgC2osU20cKRpcFIQn1Xej0GKOQL7MQgWgEVwZ7qGMKmMU7tg5cKVaDSP0Sd  hXhwjzJodth1jEQziyN9lQmk2aY6lt+37AXhJtk86DeVLv89ie61wLXqX6gDZPKv  SwuiFIECgYEA1fTtP+720YaP7KJYiOmzp9hDSK1sRSQOJPkL+UThrnXdUonJcTUV  4mSaAno1JaRLw3eLl9cuD1shCDuzHcFpRcUMxHvs5X70QbQVxYoHVpBosEgzmZR/  5XEISLhdLTOynEKMzdmLOWRbNUARtSSYau+n64CQV6eDnZDd5L2btFECgYEAy/AE  l6haY2rX83jCOROJbP9vllM4EmsdH7AntkziKsVQ3hQ3Am1tDHdLRI8djpFS8b2d  gIQj8CESj16dyYxJAhksScorWvoz5XcKmqwgLJ9mPzbzM9gWn1utb28XEa22zb7r  NZdadJF4BIajXaxaUrwqB6oQ/U6a55a4yx2e8D0CgYEA1C/ydpxkNYmRBOpAiY/U  N1pX6CTtomhOisZe5Jg2hVGo7ClwGRC/omWJA3n1aV5WMQVG8JBbwqjqyoKCcbkV  0TbKIbaRS0yrqtmRWfY4rtXKrUD9w4sth/ugbpZA+goyUZwCjkf4NjPb7KScqxwe  +LayvvEitXuRINvB5A9S8vECgYA7tx5BIiTFc6WkPznv6I6XAfy0sc17Zy3x92zJ  bp3TFE/nl1e8ODp0KgI6IwXulxMIghM6bHbxe12p6XSebZt7J8/WE4K6cwT7Le5G  +Bu7tTkFTZvNFHsbUoboWEkS/iDHAh5xZM4xz96qxJnVdgBEtaBA+kVbRscJBRM7  NoMxqQKBgQCEi2iSK58g6o4r6TCAzrs3xsP3TjVVaDdExbI60/iU2Q9dbRLBzQqT  CIn/hDwrSoXsxB7XALybzDxDDlKG7nyQr7JmudeF73cuZttQsCDanId2oTpkD6q2  ORjYc/STkir2g6y+qeGEiH0zRwDseu321KcI0IDcSLVF4QfiippMcg==  -----END RSA PRIVATE KEY-----  destinationCACertificate: |  -----BEGIN CERTIFICATE-----  MIID4TCCAsmgAwIBAgIRAJpyUhmRgSMq25gE/AHCb70wDQYJKoZIhvcNAQELBQAw  gYkxCzAJBgNVBAYTAkdCMQ8wDQYDVQQHEwZMb25kb24xDzANBgNVBAkTBkxvbmRv  bjEwMC4GA1UECxMnSUJNIE1heGltbyBBcHBsaWNhdGlvbiBTdWl0ZSAoSW50ZXJu  YWwpMSYwJAYDVQQDEx1pbnRlcm5hbC5pbnN0ODgwOS5tYXMuaWJtLmNvbTAeFw0y  MjExMDIyMDQ3MDVaFw00MjEwMjgyMDQ3MDVaMIGJMQswCQYDVQQGEwJHQjEPMA0G  A1UEBxMGTG9uZG9uMQ8wDQYDVQQJEwZMb25kb24xMDAuBgNVBAsTJ0lCTSBNYXhp  bW8gQXBwbGljYXRpb24gU3VpdGUgKEludGVybmFsKTEmMCQGA1UEAxMdaW50ZXJu  YWwuaW5zdDg4MDkubWFzLmlibS5jb20wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAw  ggEKAoIBAQC2UxfHSSrZ/Q++kde8Bn81KjTPWPzxnwZIP18S0mJc4e+Ia3Nn6C67  sa5/KbfMYU4mlt/ZWPalazQwcRlhCSPCilIatJ+IZYJ1Lw2RNyfR2F6SJxkRiVOj  Uzm9wTxUP+3ubGehbEZ7G200CyWIb/CsIp5Tw1YoB6o8rmxCYyVejFKAx4rnet7r  aXf0Sa74O5HIfCtFWe+6r9Q5+7v2Q8XhNWLwWi8pHGuYo6BHm8hlDRuGQFohXdWy  kQIURFGiRd80fHFuEoDi56zyxxWi3/vPz+OMHgvkD4Ngt0kLpAYK2MhFiuaB54oO  E8JNkyPmOvfaA1TdRY83mWh0DQ8iB0tzAgMBAAGjQjBAMA4GA1UdDwEB/wQEAwIC  BDAPBgNVHRMBAf8EBTADAQH/MB0GA1UdDgQWBBTlmDmFnobbo5x671Tl4UxlbJ8+  KDANBgkqhkiG9w0BAQsFAAOCAQEAlLGwYQi8ual7jzGKrkEYvD/yryOIhaLDFUrb  TrR/npPpJtryucoLd2c/ujENpeFrbg0VH8fxctnMlFH9RCHffESQ8tUejkKg9cSX  vtlWVzwChoUBxGwjYIqmjo6ZFlomhoN7LXbc4Xyp/f4IuYjVmfY8i67saBGP/MDJ  cWovMyJi+SiFleX/YITP4jRa5e+bwhULsGz2LmTUQDvoYXl7K8Evy5wFRR4kYiW+  TLAQD6gZrQ+TmJQd7w640AP019VE8n2MRzT8JCSnoSnfEpR5ptIEpqcWPoA5tmUc  8XLKl3GXngMAARNtdVZl+bPoDMG2XmFh1pHT1GZ3NS7RMbqURw==  -----END CERTIFICATE-----  wildcardPolicy: None |

## Register OIDC

## Start Liberty

|  |
| --- |
| > oc login --token=**<token>** --server=https://api.sno8809.ezmaxcloud.com:6443  > kubectl exec **<pod\_name>** -c **<container\_name>** -it -- /bin/sh  # cd /opt/ibm/wlp/usr/servers/defaultServer  # source ./setenv.sh  # /opt/ibm/wlp/bin/server start |