

Quick Start guide to ICP4i ACE on RHOS 4.2

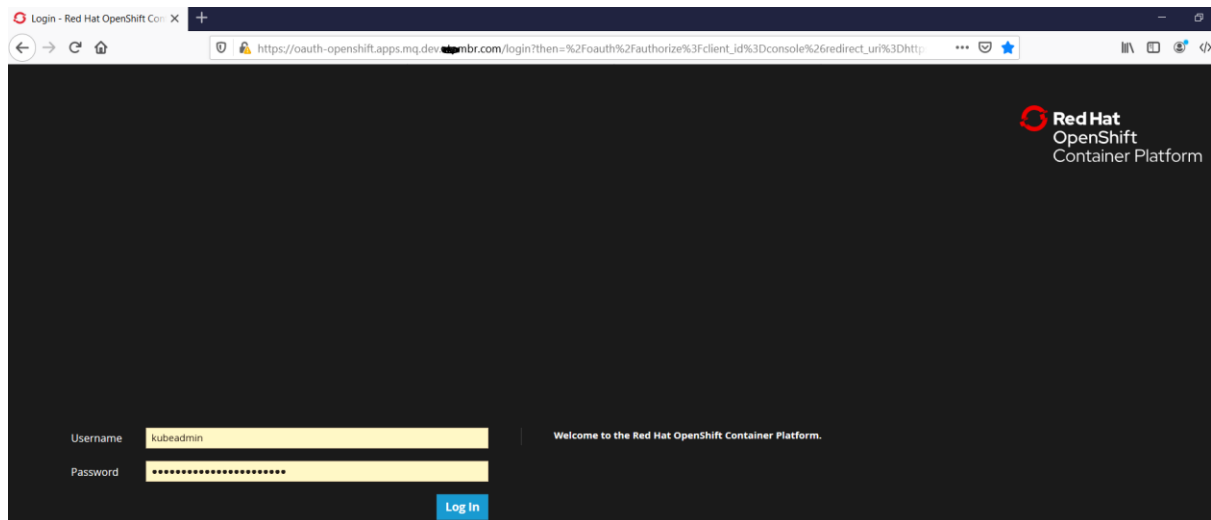
Contents

| | |
|--|----|
| IBM Cloud Pak for Integration – RHOS 4.2..... | 3 |
| RHOS ICP Access point and navigation | 3 |
| Getting key information from RHOS 4.2 | 3 |
| Command line log in | 3 |
| Get the onboard docker image registry | 4 |
| Get the pull secret..... | 4 |
| Get the Network file system | 4 |
| ICP Proxy Address | 5 |
| Images, Image Streams and Image Repository | 7 |
| ICP4i via ICP Access point and navigation..... | 8 |
| Key Information from ICP | 10 |
| ICP login information for kubectl commands | 10 |
| ICP Helm Releases..... | 12 |
| Helm releases..... | 12 |
| ICP Catalog | 13 |
| ICP4i console (Platform home) | 14 |
| Running up an ACE instance on ICP4i on RHOS 4.2 | 15 |
| Add a new ace server and associate a BAR..... | 15 |
| Information you'll need to configure the release | 17 |
| Configuration parameters..... | 18 |
| Observing the ACE deployment via ICP4i | 22 |
| Manage the ACE server..... | 23 |
| Call Test LiveLiness Probe | 25 |
| Testing Liveliness Probe | 26 |
| Observing the ACE deployment via RHOS | 27 |

IBM Cloud Pak for Integration – RHOS 4.2

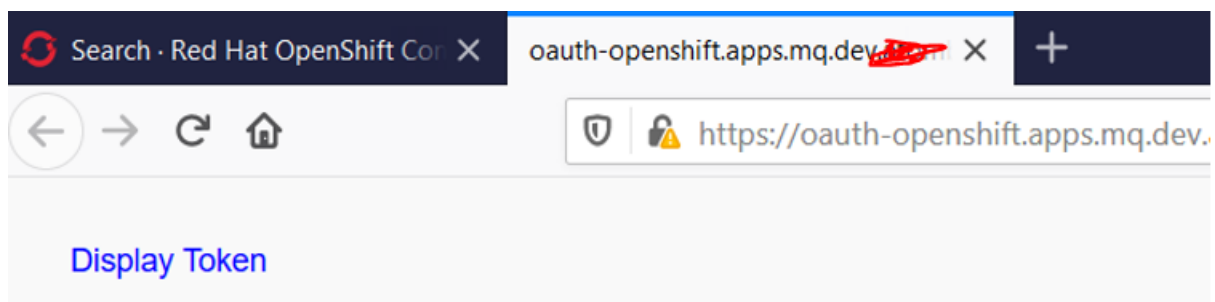
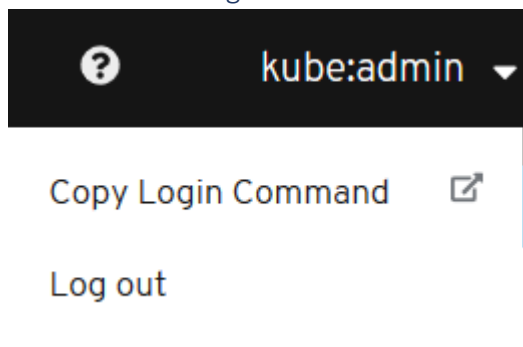
RHOS ICP Access point and navigation

Openshift: <https://oauth-openshift.apps.mq.dev.###mbr.com/login?>



Getting key information from RHOS 4.2

Command line log in



Your API token is

x7HVSb2v-D3-Wp0i1Z9o7X0sD0-WZV0vz-d2K0bVz0Y

Log in with this token

```
oc login --token=x7HVSb2v-D3-Wp0i1Z9o7X0sD0-WZV0vz-d2K0bVz0Y --server=https://api.mq.dev.6443mbr.com:6443
```

Use this token directly against the API

```
curl -H "Authorization: Bearer x7HVSb2v-D3-Wp0i1Z9o7X0sD0-WZV0vz-d2K0bVz0Y" "https://api.mq.dev.6443mbr.com:6443/apis/user.openshift.io/v1/users/~"
```

```
C:\openshift>oc login --token=x7HVSb2v-D3-Wp0i1Z9o7X0sD0-WZV0vz-d2K0bVz0Y --server=https://api.mq.dev.6443mbr.com:6443
Logged into "https://api.mq.dev.6443mbr.com:6443" as "kube:admin" using the token provided.
```

You have access to 59 projects, the list has been suppressed. You can list all projects with 'oc projects'

Using project "tracing".

```
C:\openshift>
```

Get the onboard docker image registry

```
C:\openshift>oc get route -n openshift-image-registry
NAME          HOST/PORT
default-route default-route-openshift-image-registry.apps.mq.dev.6443mbr.com
PATH          SERVICES  PORT  TERMINATION  WILDCARD
image-registry <all>    reencrypt  None
C:\openshift>
```

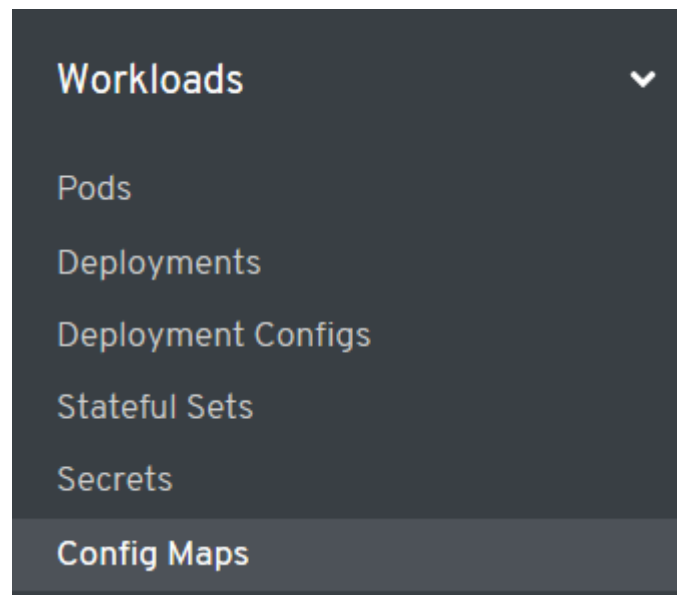
Get the pull secret

```
C:\openshift>oc get secrets
NAME          TYPE          DATA  AGE
builder-dockercfg-rqtnk  kubernetes.io/dockercfg  1      58d
builder-token-4pms7      kubernetes.io/service-account-token  4      58d
builder-token-q5zdl      kubernetes.io/service-account-token  4      58d
default-dockercfg-dgzbh  kubernetes.io/dockercfg  1      58d
default-token-8blm6      kubernetes.io/service-account-token  4      58d
default-token-8frfz      kubernetes.io/service-account-token  4      58d
deployer-dockercfg-7tkh4  kubernetes.io/dockercfg  1      58d
deployer-token-cv8sw      kubernetes.io/service-account-token  4      58d
deployer-token-t2h8x      kubernetes.io/service-account-token  4      58d
image-bot-dockercfg-hqdjc  kubernetes.io/dockercfg  1      58d
image-bot-token-mf6hx     kubernetes.io/service-account-token  4      58d
image-bot-token-trslc     kubernetes.io/service-account-token  4      58d
od-sec-tracing-dockercfg-h9crx  kubernetes.io/dockercfg  1      48d
od-sec-tracing-token-bqhsK  kubernetes.io/service-account-token  4      48d
od-sec-tracing-token-jxtkx  kubernetes.io/service-account-token  4      48d
C:\openshift>
```

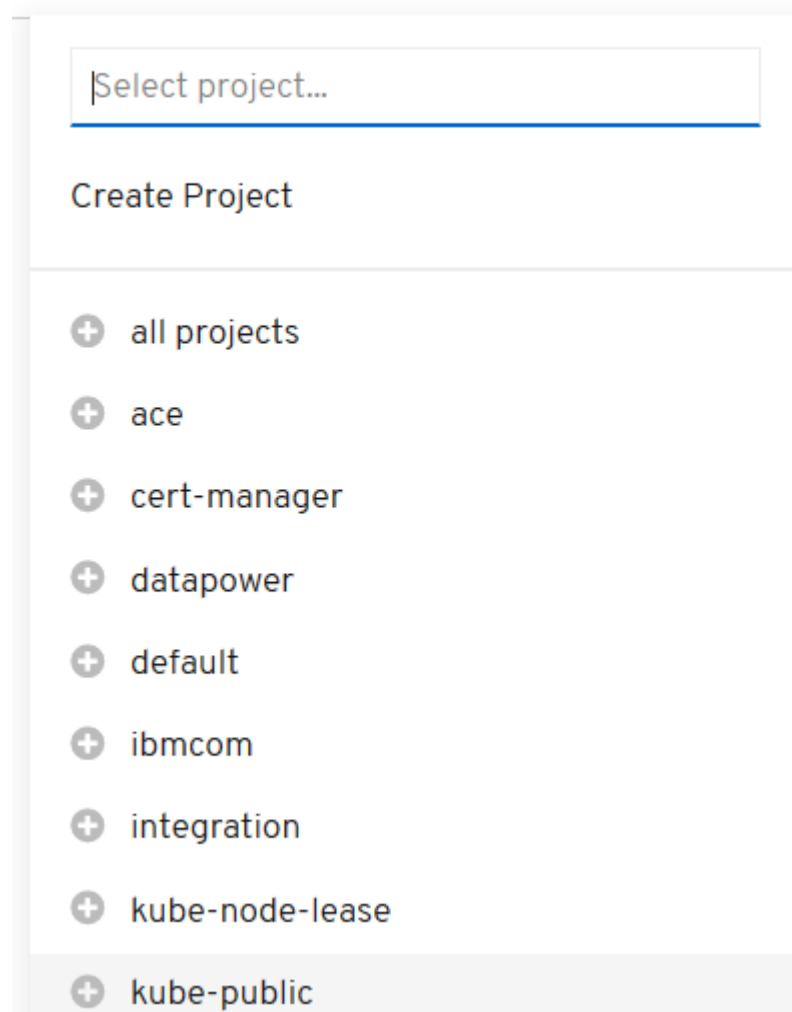
Get the Network file system

```
C:\openshift>oc get sc
NAME          PROVISIONER          AGE
aws-efs       openshift.org/aws-efs  51d
gp2 (default)  kubernetes.io/aws-ebs  59d
```

ICP Proxy Address



Project: kube-public ▼




Project: kube-public ▼

Config Maps

Create Config Map

Name ↑

CM ibmcloud-cluster-info

 **Red Hat**
OpenShift Container Platform

⚙️ Administrator ▼

Home ▼

Dashboards

Projects

Search

Explore

Events

Operators

Workloads ▼

Pods

Deployments

Deployment Configs

Stateful Sets

Secrets

Config Maps

You are logged in as a temporary a

Project: kube-public ▼

Config Maps > Config Map Details

CM ibmcloud-cluster-info


Overview **YAML**

Config Map Overview

Name
ibmcloud-cluster-info

Namespace
NS kube-public

Labels
No labels


Annotations
1 Annotation 

Data

proxy_address

icp-proxy.apps.mq.dev ~~ibm~~combr.com

Images, Image Streams and Image Repository

 **Red Hat**
OpenShift Container Platform

Cron Jobs

Jobs

Daemon Sets

Replica Sets

Replication Controllers

Horizontal Pod Autoscalers

Networking

Storage

Builds

Build Configs

Builds

Image Streams

Monitoring

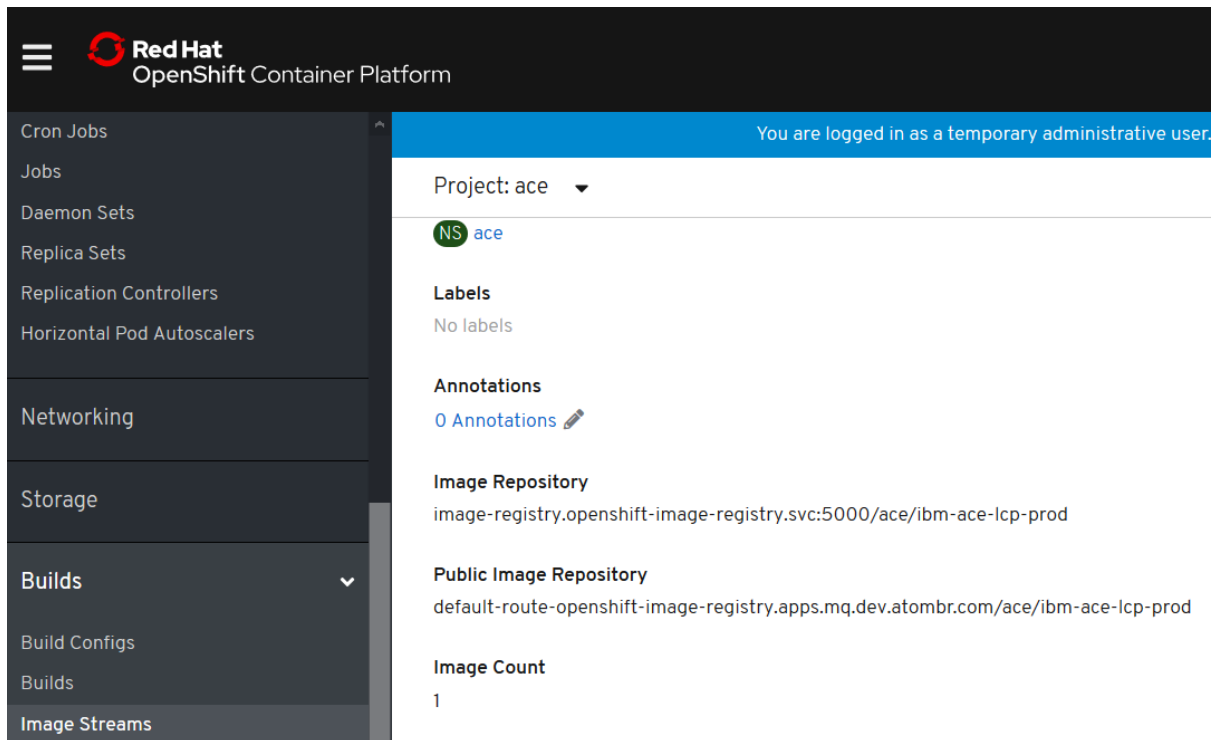
You are logged in as a temporary user

Project: ace

Image Streams

Create Image Stream

| Name | Namespace |
|-------------------------------|-----------|
| ibm-ace-content-server-prod | NS ace |
| ibm-ace-dashboard-prod | NS ace |
| ibm-ace-designer-flows-prod | NS ace |
| ibm-ace-icp-configurator-prod | NS ace |
| ibm-ace-lcp-prod | NS ace |



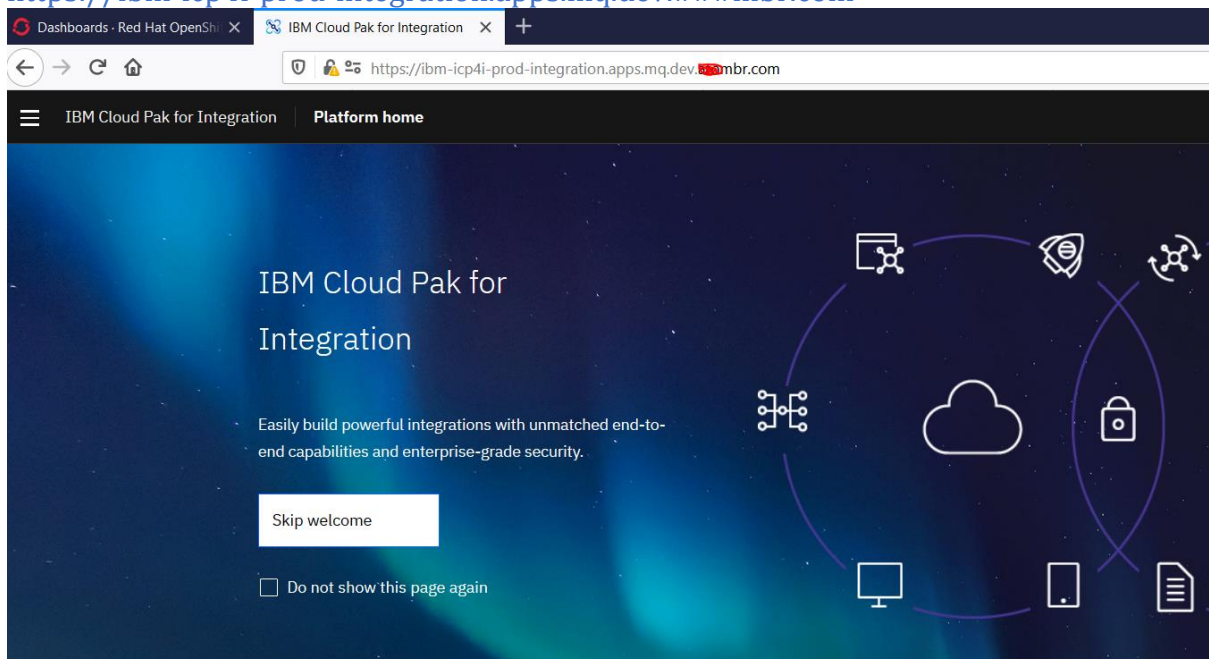
Example image for ace standalone: image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-lcp-prod

ICP4i via ICP Access point and navigation

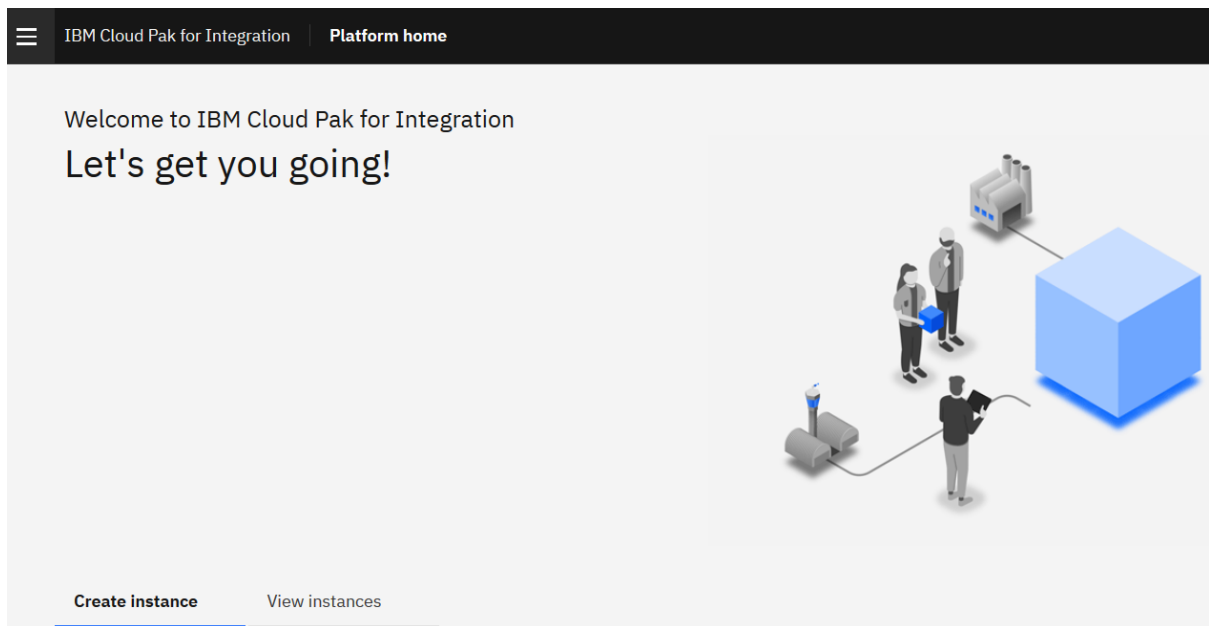
Add <https://ibm-icp4i-prod-integration>

To the head of the base Openshift cluster URL apps.mq.dev.###mbr.com

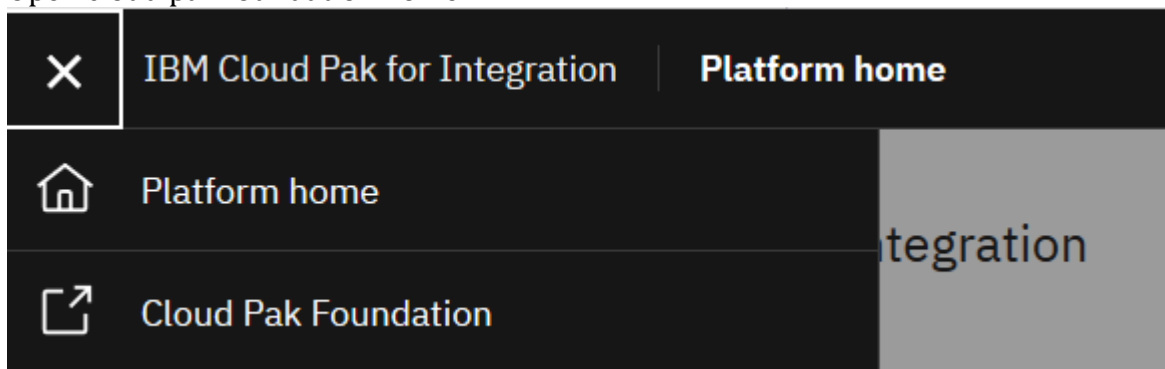
<https://ibm-icp4i-prod-integration.apps.mq.dev.###mbr.com>




Skip Welcome



Open cloud pak foundation home



 <https://icp-console.apps.mq.dev.ibm.com/oidc/login.jsp>

IBM Cloud Pak

Fast. Flexible.
Intelligent. Open.
Enterprise-grade.

Log in to your account

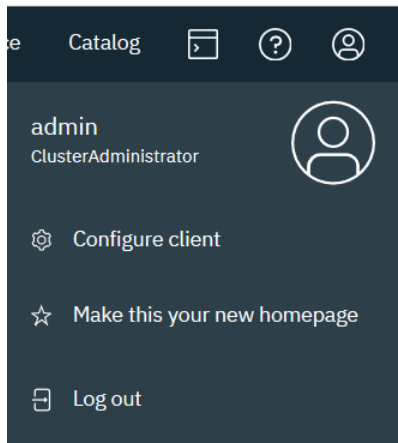
To log in with SSO, enter only the username and click the log in button

Username

admin

Key Information from ICP

ICP login information for kubectl commands



Configure client



Before you run commands in the `kubectl` command line interface for this cluster, you must configure the client.

Prerequisites:

Install CLI tools

To configure the CLI, paste the displayed configuration commands into your terminal window and run them:

```
kubectl config set-cluster mycluster --server=https://api.mq.dev.ator ^
kubectl config set-context mycluster-context --cluster=mycluster
kubectl config set-credentials admin --token=60gmRI2vyr_0zoDwRJRysSO:
kubectl config set-context mycluster-context --user=admin --namespace
kubectl config use-context mycluster-context
```



```
Administrator: Command Prompt

C:\kubect>kubectl config set-cluster mycluster --server=https://api.mq.dev.🐙mbr.com:6443 --insecure-skip-tls-verify=true
Cluster "mycluster" set.

C:\kubect>kubectl config set-context mycluster-context --cluster=mycluster
Context "mycluster-context" modified.

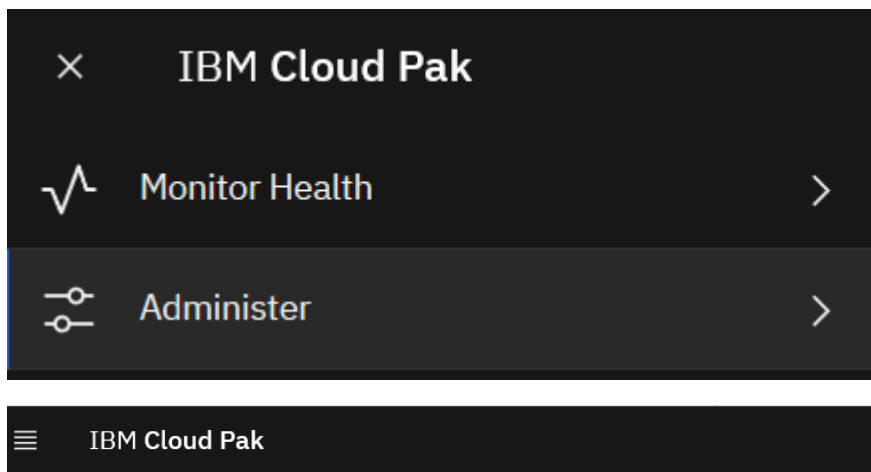
C:\kubect>kubectl config set-credentials admin --token=60gmRI2vyr_0zoDwRJrysSOJjB9_PDK_Iotwr3JDkkCu
User "admin" set.

C:\kubect>kubectl config set-context mycluster-context --user=admin --namespace=tracing
Context "mycluster-context" modified.

C:\kubect>kubectl config use-context mycluster-context
Switched to context "mycluster-context".

C:\kubect>
```

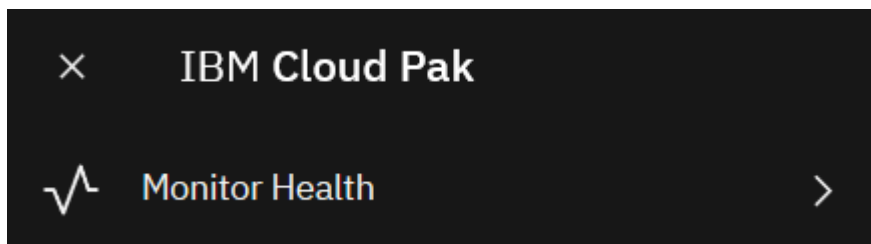
ICP Helm Releases



Helm Repositories


| <input type="text" value="Search repositories"/> | |
|--|---|
| Name | Url |
| ibm-charts | https://raw.githubusercontent.com/IBM/charts/master/repo/stable/ |


Helm releases







Select Helm releases

Helm releases


 You are currently viewing only the helm releases of this cluster.

 Search releases

| Name ^ | Namespace | Status | Chart name | Current version |
|----------------|-----------|--|--|-----------------|
| abmqr2 | mq |  Deployed | ibm-mqadvanced-server-integration-prod | 5.0.0 |
| ace-dashboard1 | ace |  Deployed | ibm-ace-dashboard-icp4i-prod | 3.0.0 |
| ace1 | ace |  Deployed | ibm-ace-server-icp4i-prod | 3.0.0 |
| aceda | ace |  Deployed | ibm-ace-server-icp4i-prod | 3.0.0 |

ICP Catalog

Catalog

 Search the catalog...

All Categories >

AI & Watson

Blockchain

Business Automation

Data

Data Science & Analytics

DevOps

Integration

IoT

Network

Operations

Runtimes & Frameworks

Security


Classification ▾

Cloud Platform ▾

Architecture ▾

Qualification


Cloud Paks




ibm-icp4i-prod

ibm-entitled-charts

A Helm chart for the IBM Cloud Pak for Integration Navigator






ibm-icp4i-pr

local-charts

A Helm chart for


Helm Charts




aqua-enforcer

ibm-community-charts

A Helm chart for the Aqua Enforcer






aqua-scanner

ibm-community-charts

A Helm chart for the aqua scanner cli component




ICP4i console (Platform home)

IBM Cloud Pak for Integration

Platform home

Create instance


View instances



API Connect

Create, manage and secure your APIs

[Create instance](#)



App Connect

Unlock the power of your data to drive new opportunities

[Create instance](#)

View instances and select the ace-dashboard

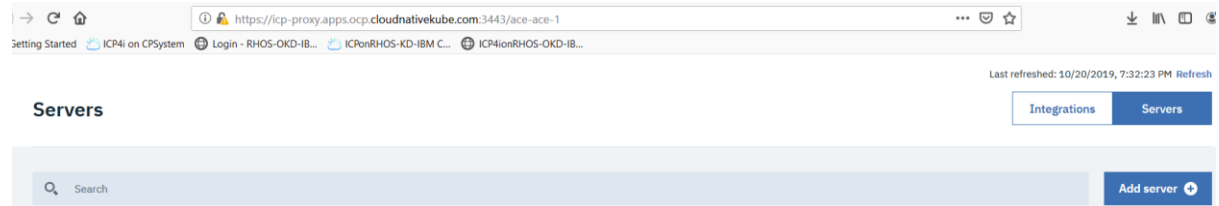
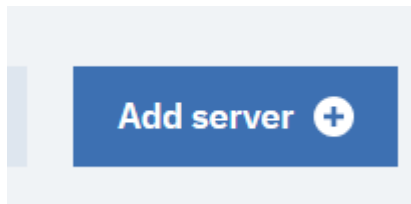
Create instance

View instances

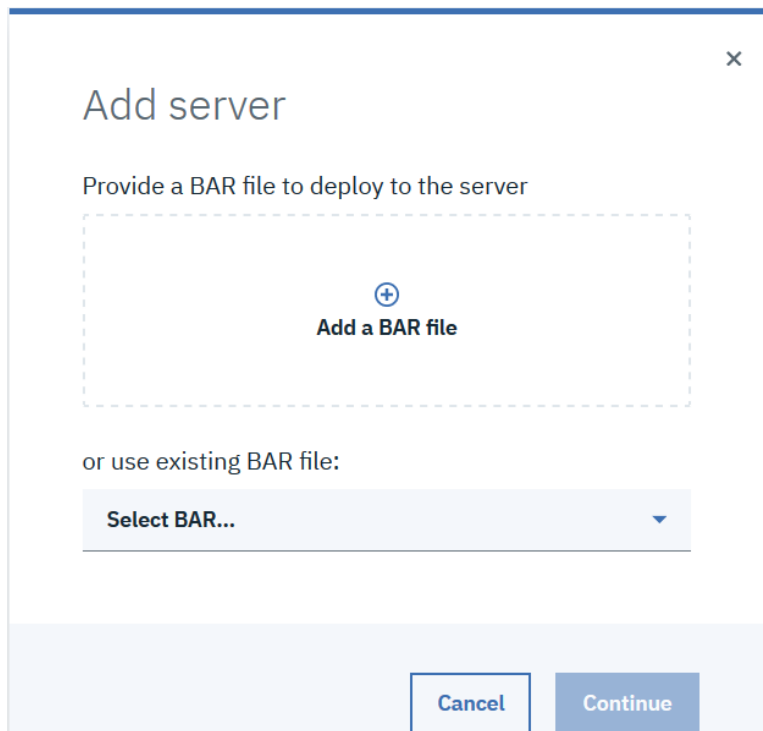
| Capability type | Instance name | Namespace |
|-----------------|--------------------------------|-----------|
| App Connect | ace-dashboard1 | ace |

Running up an ACE instance on ICP4i on RHOS 4.2

Add a new ace server and associate a BAR





Add a server



Navigate to a bar

<https://github.com/DAVEXACOM/ACEonICPIntSupportingMaterial/tree/master/ace-livelinessProbe>


| ACEonICPIntMicSoE > ACEonICPIntMicSoE-master > BARfiles | |
|---|--------------------|
| <input type="checkbox"/> Name | Date modified |
|  .project | 20/10/2019 5:39 PM |
| <input checked="" type="checkbox"/>  SoE.bar | 20/10/2019 5:39 PM |

×

Add server

Provide a BAR file to deploy to the server

×



SoE.bar

or use existing BAR file:

Select BAR... ▼

Cancel

Continue

Add server

You will now configure and install a Helm release to deploy to the server. It is important to copy the Content URL and select the current Namespace.

Content URL:

<https://ace-dashboard1-ibm-ace-dashboard-icp4i-prod:3443/v1/directories/SoE?34f7d17a-e417-4c0f-8df0-66f363f8364c>



Namespace:

ace


Take a copy of the url to the BAR (content) file

<https://ace-dashboard1-ibm-ace-dashboard-icp4i-prod:3443/v1/directories/SoE?34f7d17a-e417-4c0f-8df0-66f363f8364c>

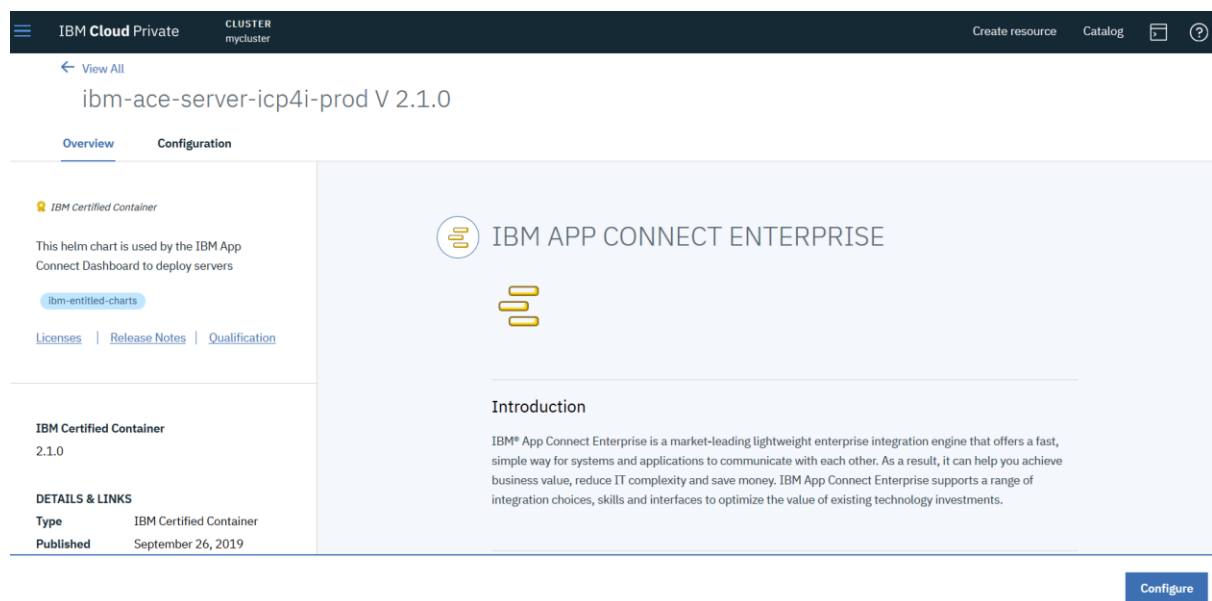
If the integration server requires any configuration to be applied then you will need to use the following download to provide the configuration prior to install. Refer to the README.md inside the download on how to create the required secrets:

[Download configuration package](#)

Cancel

Configure release 

Overview of Helm Chart. Switch to configuration



The screenshot shows the IBM Cloud Private interface for configuring the `ibm-ace-server-icp4i-prod` Helm chart (version 2.1.0). The left sidebar contains navigation links: `View All`, `Overview` (selected), and `Configuration`. Below these are links for `IBM Certified Container`, `ibm-entitled-charts`, `Licenses`, `Release Notes`, and `Qualification`. The main content area displays the `IBM APP CONNECT ENTERPRISE` logo and an introduction paragraph. At the bottom right, there is a `Configure` button.

Information you'll need to configure the release

Collect information you need to configure the release.

ICP4i Proxy address (from RHOS UI -> configMaps): icp-proxy.apps.ocp.cloudnativekubernetes.com

Content URL (from ICP4i): <https://ace-1-ibm-ace-dashboard-icp4i-prod:3443/v1/directories/SoE?23e24c6c-c709-4b5e-a7ec-0dd05b96394f>

Images – default in chart is `cp.icr.io/ibm-ace-server-prod:11.0.0.6.1` needs to be changed

Get the images from RHOS UI

image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-server-prod:11.0.0.6.1

image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-mqclient-server-prod:11.0.0.6.1

image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-mq-server-prod:11.0.0.6.1

image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-icp-configurator-prod:11.0.0.6.1

image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-designer-flows-prod:11.0.0.6.1

image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-lcp-prod:11.0.0.6.1

image-registry.openshift-image-registry.svc:5000/ace/icp4i-od-agent

image-registry.openshift-image-registry.svc:5000/ace/icp4i-od-collector

replace cp.icr.io with **image-registry.openshift-image-registry.svc:5000/ace** in the configuration

pull secret(from oc command oc get secrets):

deployer-dockercfg-Nxxxxx

Managed NFS name (from oc command oc get sc):

gp2

Configuration parameters

IBM Cloud Pak

[← View All](#)

ibm-ace-server-icp4i-prod V 3.0.0

[Overview](#) [Configuration](#)

Configuration

This helm chart is used by the IBM App Connect Dashboard to deploy servers. Edit these parameters for configuration.

Helm release name *

aceda

Target namespace *

ace

License *

☒ I have read and agreed to the [License agreement](#)

Target cluster *

local-cluster

☒ local-cluster

Parameters

To install this chart, additional configuration is needed in Quick start. To customize installation, view and edit All parameters.

> **Quick start**
Required and recommended parameters to view and edit.

☒ **All parameters**
Other required, optional, and read-only parameters to view and edit.

Content Server URL *

https://ace-dashboard1-ibm-ace-dashboard-icp4i-prod:3443/v1/directories/SoE?34f7d17a-e417-4c0f-8df0-66f363f8364c

☒ **Production usage**

Which type of image to run

App Connect Enterprise only

Architecture scheduling preference *

amd64

IBM App Connect Designer flows *


Disabled

Set the pull secret (deployer-dockercfg-6lnxt) and the image locations and names correctly

| Images | |
|---|---|
| Define images to be used | |
| Docker image for App Connect Enterprise * | Docker image for App Connect Enterprise with MQ client * |
| image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-server-prod:11.0.0.6.1 | image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-mqclient-server-prod:11.0.0.6.1 |
| Docker image for App Connect Enterprise with MQ server * | Configurator Docker image * |
| image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-mq-server-prod:11.0.0.6.1 | image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-icp-configurator-prod:11.0.0.6.1 |
| Designer flows Docker image * | Connectors Docker image *¹ |
| image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-designer-flows-prod:11.0.0.6.1 | image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-icp-prod:11.0.0.6.1 |
| Image pull policy | Image pull secret |
| IfNotPresent | deployer-dockercfg-7tkh4 |

Leave service as default

| | |
|---------------------------|---------------------------|
| Service | |
| Service settings | |
| Service type * | Endpoint type * |
| ClusterIP | HTTP |
| Web UI Port * | HTTP port * |
| 7600 | 7800 |
| HTTPS port * | Switch AgentC Port |
| 7843 | Enter value |
| Switch AgentP Port | Switch Admin Port |
| Enter value | Enter value |

| | |
|--|---|
| Integration Server | |
| Define configuration for the Integration Server | |
| Integration server name  | List of key aliases for the keystore |
| myIntSRV | Enter value |
| List of certificate aliases for the truststore | Name of the default application |
| Enter value | Enter value |
| The name of the secret to create or to use that contains the server configuration | File system group ID |
| Enter value | Enter value |

Reduce replicas to 1

| | |
|---|-------------------------|
| Configuration for App Connect Enterprise (without MQ) deployments | |
| Configuration settings for specifying required resources when running App Connect Enterprise without MQ | |
| CPU request * | Memory request * |
| 200m | 256Mi |
| CPU limit * | Memory limit * |
| 1 | 1024Mi |
| Replica count | |
| 1 | |


Set gp2

Data persistent volume claims (PVCs)

Settings for the PVCs (applicable only when running with a queue manager)

Name *

data

Storage class name 

gp2

Size *

2Gi

Increase initial delay

Readiness probe

Settings for the readiness probe that checks if the integration server is ready

Initial delay (seconds)

30

OD agent image repository

image-registry.openshift-image-registry.svc:5000/ace/icp4i-od-agent

OD agent image tag

1.0.1

OD agent liveness probe initial delay (seconds)

10



OD agent readiness probe initial delay (seconds)

10

OD collector image repository 

image-registry.openshift-image-registry.svc:5000/ace/icp4i-od-collector

OD collector image tag

1.0.1

OD collector liveness probe initial delay (seconds)

10



OD collector readiness probe initial delay (seconds)

10

OD tracing instance namespace

Enter value

Install

Might have to wait a little while – then RIGHT Click on View helm releases and open in a new tab



Installation started. For progress view your Helm releases.

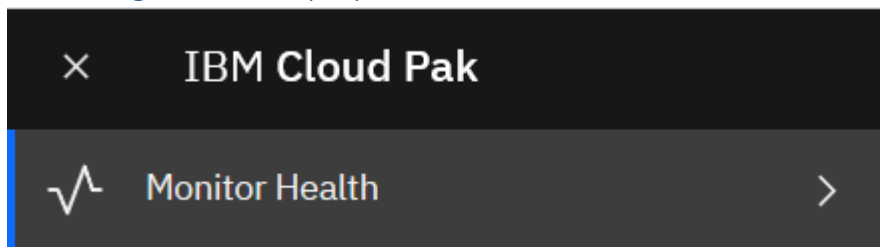
[View Helm Releases](#)

Then click the X to dismiss the dialog.



This way you can leave the parameters in the chart and try again if your Helm release fails without having to enter all params again.


Observing the ACE deployment via ICP4i



Then select helm releases

| Name ▲ | Namespace | Status | Chart name |
|----------------|-----------|------------|--|
| abmqr2 | mq | ● Deployed | ibm-mqadvanced-server-integration-prod |
| ace-dashboard1 | ace | ● Deployed | ibm-ace-dashboard-icp4i-prod |
| ace1 | ace | ● Deployed | ibm-ace-server-icp4i-prod |
| aceda | ace | ● Deployed | ibm-ace-server-icp4i-prod |

Select aceda

 IBM Cloud Pak

[← View All](#)
aceda
● Deployed
UPDATED: February 7, 2020 at 2:21 PM

Page down

Note the instructions for managing and connecting to the integration server

Notes


If you launched the deploy from the ACE Dashboard, then you can return to the ACE Dashboard to manage the server.

The HTTP and HTTPS endpoints for the ACE Integration Server are exposed with Routes.

```
export ACE_HTTP_HOSTNAME=$(kubectl get route aceda-http -o jsonpath="{.status.ingress[0].host}")
export ACE_HTTPS_HOSTNAME=$(kubectl get route aceda-https -o jsonpath="{.status.ingress[0].host}")

echo "HTTP workload can use: http://${ACE_HTTP_HOSTNAME}"
echo "HTTPS workload can use: https://${ACE_HTTPS_HOSTNAME}"
```

Manage the ACE server


 IBM Cloud Pak for Integration | Platform home


Create instance


View instances


| Capability type | Instance name |
|-----------------|----------------|
| App Connect | ace-dashboard1 |

Servers

 Search



eghigh2
Server



IS1
Server



myIntSRV
Server

[Dashboard](#) / [Server: myIntSRV](#)



 Search


LivelinessProbe
API



LivelinessProbe

[Documentation](#)

[Contents](#)

[Properties](#)

[Other resources](#)

REST API Base URL

<http://aceda-http-ace.apps.mq.dev.atombr.com:80/livelinessProbe/v1>

OpenAPI document

<http://aceda-http-ace.apps.mq.dev.atombr.com:80/livelinessProbe/v1/swagger.json>

 Search

/message

Call Test LiveLiness Probe

Github Source Repos for ACE Liveliness Probe (The SoE ACE project)

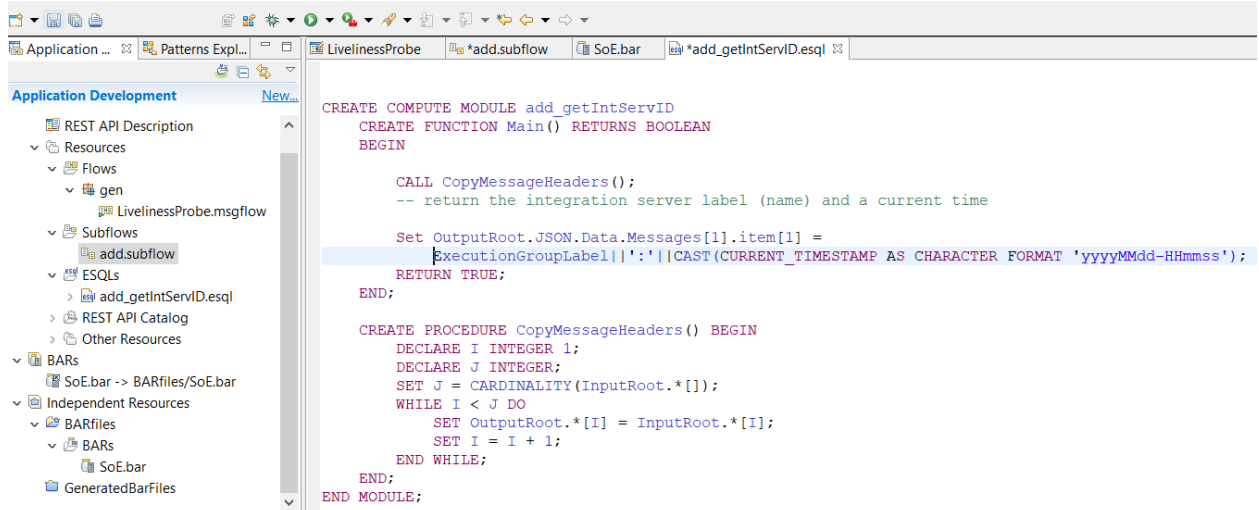
<https://github.com/DAVEXACOM/ACEonICPIntMicSoE>

Description

The Liveliness Probe Service is an other restful service (this is not the service baked into the cloud pak images). Its just a user echo test service

Integration Development - LivelinessProbe/add_getIntServID.esql - IBM App Connect Enterprise Toolkit - C:\Users\DavidArnold\IBM\ACET11\ACEonICPIntMicSoE

File Edit Source Navigate Search Project Run Window Help



```
CREATE COMPUTE MODULE add_getIntServID
CREATE FUNCTION Main() RETURNS BOOLEAN
BEGIN

    CALL CopyMessageHeaders();
    -- return the integration server label (name) and a current time

    Set OutputRoot.JSON.Data.Messages[1].item[1] =
        ExecutionGroupLabel||':'||CAST(CURRENT_TIMESTAMP AS CHARACTER FORMAT 'yyyymmdd-HH:mm:ss');
    RETURN TRUE;
END;

CREATE PROCEDURE CopyMessageHeaders() BEGIN
    DECLARE I INTEGER 1;
    DECLARE J INTEGER;
    SET J = CARDINALITY(InputRoot.*[1]);
    WHILE I < J DO
        SET OutputRoot.*[I] = InputRoot.*[I];
        SET I = I + 1;
    END WHILE;
END;
END MODULE;
```

You can use oc get routes or kubectl get routes

```
Administrator: Command Prompt

C:\kubect\kubect\ config use-context mycluster-context
Switched to context "mycluster-context".

C:\kubect\kubect\ get route aceda-http -o jsonpath="{.status.ingress[0].host}"
Error from server (NotFound): routes.route.openshift.io "aceda-http" not found

C:\kubect\kubect\ get route aceda-http
Error from server (NotFound): routes.route.openshift.io "aceda-http" not found

C:\kubect\kubect\ get routes -n ace
NAME          HOST/PORT          PATH          SERVICES          PORT          TERMINATION          WILDCARD
ace1-http     ace1-http-ace.apps.mq.dev.atombr.com          ace1-ibm-ace-server-icp4i-prod  ace-http      passthrough/None     None
ace1-https   ace1-https-ace.apps.mq.dev.atombr.com          ace1-ibm-ace-server-icp4i-prod  ace-https     passthrough/None     None
aceda-http   ace1-https-ace.apps.mq.dev.atombr.com          ace1-ibm-ace-server-icp4i-prod  ace-https     passthrough/None     None
aceda-https  ace1-https-ace.apps.mq.dev.atombr.com          ace1-ibm-ace-server-icp4i-prod  ace-https     passthrough/None     None
eghigh2-http eghigh2-http-ace.apps.mq.dev.atombr.com          eghigh2-ibm-ace-server-icp4i-prod ace-http      passthrough/None     None
eghigh2-https eghigh2-https-ace.apps.mq.dev.atombr.com          eghigh2-ibm-ace-server-icp4i-prod ace-https     passthrough/None     None
soe-bar-http soe-bar-http-ace.apps.mq.dev.atombr.com          soe-bar-ibm-ace-server-icp4i-prod ace-http      passthrough/None     None
soe-bar-https soe-bar-https-ace.apps.mq.dev.atombr.com          soe-bar-ibm-ace-server-icp4i-prod ace-https     passthrough/None     None

C:\kubect\
notes C:\openshift>oc get routes -n ace
NAME          HOST/PORT          PATH          SERVICES          PORT          TERMINATION          WILDCARD
ace1-http     ace1-http-ace.apps.mq.dev.atombr.com          ace1-ibm-ace-server-icp4i-prod  ace-http      passthrough/None     None
ace1-https   ace1-https-ace.apps.mq.dev.atombr.com          ace1-ibm-ace-server-icp4i-prod  ace-https     passthrough/None     None
If you aceda-http   ace1-https-ace.apps.mq.dev.atombr.com          ace1-ibm-ace-server-icp4i-prod  ace-https     passthrough/None     None
The H aceda-https  ace1-https-ace.apps.mq.dev.atombr.com          ace1-ibm-ace-server-icp4i-prod  ace-https     passthrough/None     None
eghigh2-http eghigh2-http-ace.apps.mq.dev.atombr.com          eghigh2-ibm-ace-server-icp4i-prod ace-http      passthrough/None     None
eghigh2-https eghigh2-https-ace.apps.mq.dev.atombr.com          eghigh2-ibm-ace-server-icp4i-prod ace-https     passthrough/None     None
expor soe-bar-http soe-bar-http-ace.apps.mq.dev.atombr.com          soe-bar-ibm-ace-server-icp4i-prod ace-http      passthrough/None     None
expor soe-bar-https soe-bar-https-ace.apps.mq.dev.atombr.com          soe-bar-ibm-ace-server-icp4i-prod ace-https     passthrough/None     None
C:\openshift>
```

aceda-http-ace.apps.mq.dev.atombr.com

Testing Liveliness Probe

Using RESTED rest client for firefox I this example

Base URI

http://aceda-http-ace.apps.mq.dev.###mbr.com/livelinessProbe/v1

input data:

```
{ "Messages": [ "test" ] }
```

Return data

returns the integration server name plus a current timestamp for

Post

http://aceda-http-ace.apps.mq.dev.###mbr.com/livelinessProbe/v1/message

Request

POST

aceda-http-ace.apps.mq.dev.###mbr.com/livelinessProbe/v1/message

Send request

Headers >

Basic auth >

Request body >

Type

Custom

{ "Messages": ["test"] }

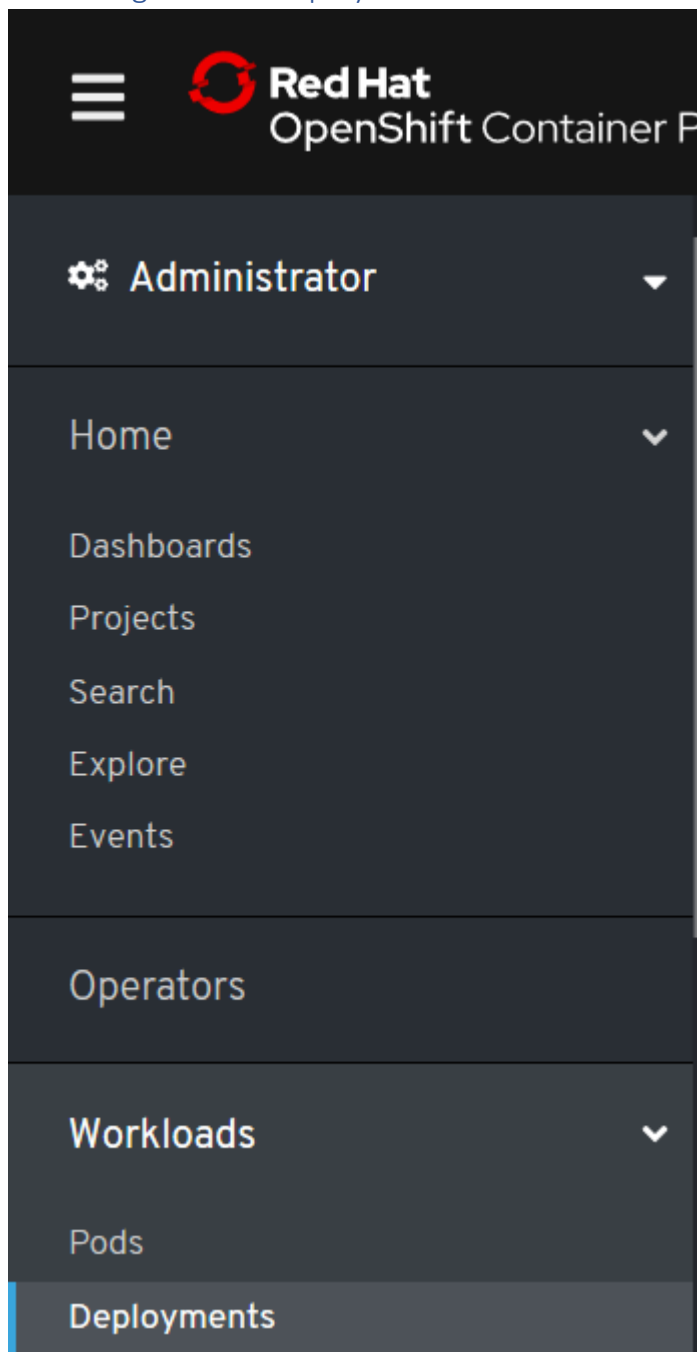
Response (0.09s) - http://aceda-http-ace.apps.mq.dev.svc.cluster.local/livelinessProbe/v1/message

200 OK

[Headers >](#)

```
{"Messages":{"item":"myIntSRV:20200207-035732"}}
```

Observing the ACE deployment via RHOS



| | | | |
|---|--------|--------------------------------------|-------------|
| Project: ace ▾ | | | |
| helm.sh/chart=ibm-ace-server-icp4i... release=ace1 | | | |
| D aceda-ibm-ace-server-icp4i-prod | NS ace | app.kubernetes.io/instance=aceda | 1 of 1 pods |
| | | app.kubernetes.io/managed...=Til... | |
| | | app.kubernet...=ibm-ace-server-... | |
| | | helm.sh/c...=ibm-ace-server-icp4i... | |
| | | release=aceda | |

| | | | |
|-----------------------------------|------|------|--------------------|
| Project: ace ▾ | | | |
| Deployments > Deployment Details | | | |
| D aceda-ibm-ace-server-icp4i-prod | | | |
| Overview | YAML | Pods | Environment Events |

| | | | |
|--|--|----------------------|--|
| Deployment Overview | | | |
| <div>1 pod</div> | | | |
| Name | | Update Strategy | |
| aceda-ibm-ace-server-icp4i-prod | | RollingUpdate | |
| Namespace | | Max Unavailable | |
| NS ace | | 1 of 1 pod | |
| Labels | | Max Surge | |
| app.kubernetes.io/instance=aceda app.kubernetes.io/managed-by=Tiller | | 1 greater than 1 pod | |
| app.kubernetes.io/name=ibm-ace-server-icp4i-prod | | Progress Deadline | |
| helm.sh/chart=ibm-ace-server-icp4i-prod release=aceda | | 10m 0s | |

Project: ace

Containers

| Name | Image | Resource Limits | Ports |
|---------------------------------|---|---------------------|------------------------------|
| aceda-ibm-ace-server-icp4i-prod | image-registry.openshift-image-registry.svc:5000/ace/ibm-ace-server-prod:11.0.0.6.1-amd64 | cpu: 1, memory: 1Gi | 7600/TCP, 7800/TCP, 7843/TCP |

Volumes

| Name | Mount Path | SubPath | Type | Permissions | Utilized By |
|----------|---------------------------------------|---------|---------------------------------|-------------|---------------------------------|
| webusers | /home/aceuser/initial-config/webusers | | aceda-ibm-ace-server-icp4i-prod | Read/Write | aceda-ibm-ace-server-icp4i-prod |

OverviewYAMLPodsEnvironmentEvents

Filter by name...

| | | | | | | | | |
|-----------|-----------|---------------|--------------------|-------------|----------|-----------|--------------------|--------|
| 1 Running | 0 Pending | 0 Terminating | 0 CrashLoopBackOff | 0 Completed | 0 Failed | 0 Unknown | Select All Filters | 1 Item |
|-----------|-----------|---------------|--------------------|-------------|----------|-----------|--------------------|--------|

| Name | Namespace | Pod Labels | Node | Status | Readiness |
|--|-----------|---|---|---------|-----------|
| aceda-ibm-ace-server-icp4i-prod-7b86fb56f5-mff8h | ace | app.kubernetes.io/name=ibm-ace-server app.kubernetes.io/version=11.0.0.6.1 helm.sh/chart=ibm-ace-server-11.0.0.6.1 pod-template-hash=7b86fb56f5 release=aceda | ip-10-0-136-189.ap-southeast-2.compute.internal | Running | Ready |