Day 4

Info - How to see the values stored inside etcd database?

Info - What is an Openshift Job?

- any one time activity we can create them as a Job
- Example
    - delete all Persistent Volume which are unused
    - taking backup of etcd database

Info - What is an Openshift CronJob?

- any recurring activity but that will run for few minutes and terminate we can run them as a CronJob
- Example
    - taking backup of etcd database every Sunday midnight

Lab - Create a one-time job

```
cd ~/openshift-27may-2024
git pull
cd Day4/job
oc apply -f job.yml
oc get jobs
oc get pods
oc logs job/hello-job
```

Once you are done with the exercise, you may cleanup the resources

```
cd ~/openshift-27may-2024
cd Day4/job
oc delete -f job.yml
```

Lab - Create a recurring job using Cronjob

```
cd ~/openshift-27may-2024
git pull
cd Day4/cronjob
oc apply -f cronjob.yml
oc get cronjobs
oc get po -w
oc logs cronjobs/cron-job
```

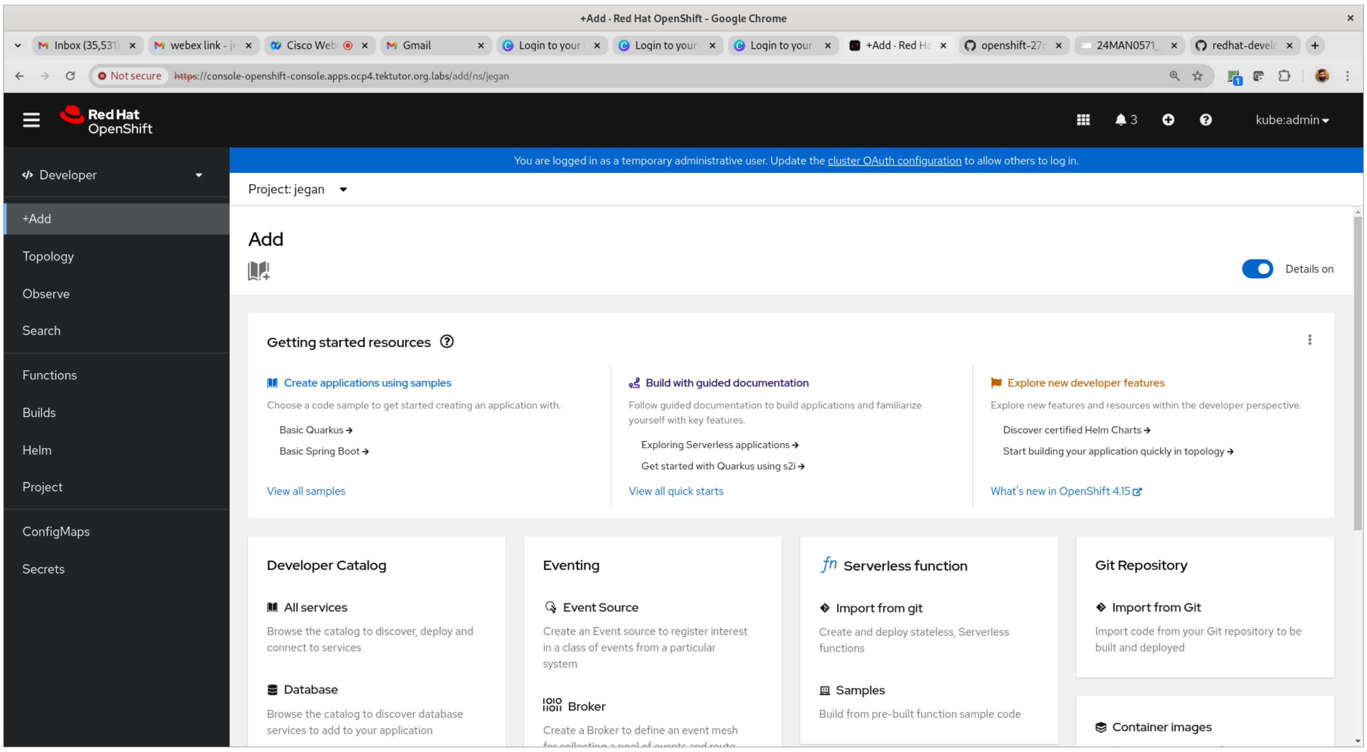
Once you are done with this exercise, you may delete the cronjob

```
cd ~/openshift-27may-2024
cd Day4/cronjob

oc delete -f cronjob.yml
```

Info - What is DeploymentConfig?

- In older versions of Kubernetes to deploy stateless application we had to use ReplicationController
- In Red Hat they wanted to support declarative style while scaling and while performing rolling update, hence they add a new type of custom resource in OpenShift called DeploymentConfig
- ReplicationController supports both Scaling and rolling update, which is not a good design as it does more than one thing (against SRP SOLID Design Priniciple)
- DeploymentConfig helps us deploy stateless applications
- Meanwhile, Google refactored the ReplicationController into two resources
  - 1. Deployment - which takes care of Rolling update
    2. ReplicaSet - which takes care of scaling up/down
- As per SOLID Design Priniciples
  - S - Single Responsibility Principle (SRP)
  - O - Open Closed Principle (OCP)
  - L - Liskov Substitution Principle
  - I - Interface Seggration
  - D - Dependency Injection or Dependency Inversion or Inversion of Control (IOC)
- By the Kubernetes added Deployment & ReplicaSet as an alternate to ReplicationController, the OpenShift team already added Deployment Config
- In new versions of OpenShift we would see
  - Deployment & ReplicaSet
  - DeploymentConfig (this was introduced in openshift when there was no Deployment & ReplicaSet, hence we should avoid using DeploymentConfig instead we should use Deployment)
  - ReplicationController (old kubernetes features now ideally we should use Deployment)

Lab - Deploying Angular application from OpenShift Webconsole using Developer context


+Add · Red Hat OpenShift - Google Chrome
Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/add/ns/jegan
Red Hat OpenShift
kube:admin
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan

Add

Details on

Getting started resources

Create applications using samples

Choose a code sample to get started creating an application with.

- Basic Quarkus →
- Basic Spring Boot →

View all samples

Build with guided documentation

Follow guided documentation to build applications and familiarize yourself with key features.

- Exploring Serverless applications →
- Get started with Quarkus using s2i →

View all quick starts

Explore new developer features

Explore new features and resources within the developer perspective.

- Discover certified Helm Charts →
- Start building your application quickly in topology →

What's new in OpenShift 4.15

Developer Catalog

All services

Browse the catalog to discover, deploy and connect to services

Database

Browse the catalog to discover database services to add to your application

Eventing

Event Source

Create an Event source to register interest in a class of events from a particular system

Broker

Create a Broker to define an event mesh for collecting a pool of events and route

Serverless function

Import from git

Create and deploy stateless, Serverless functions

Samples

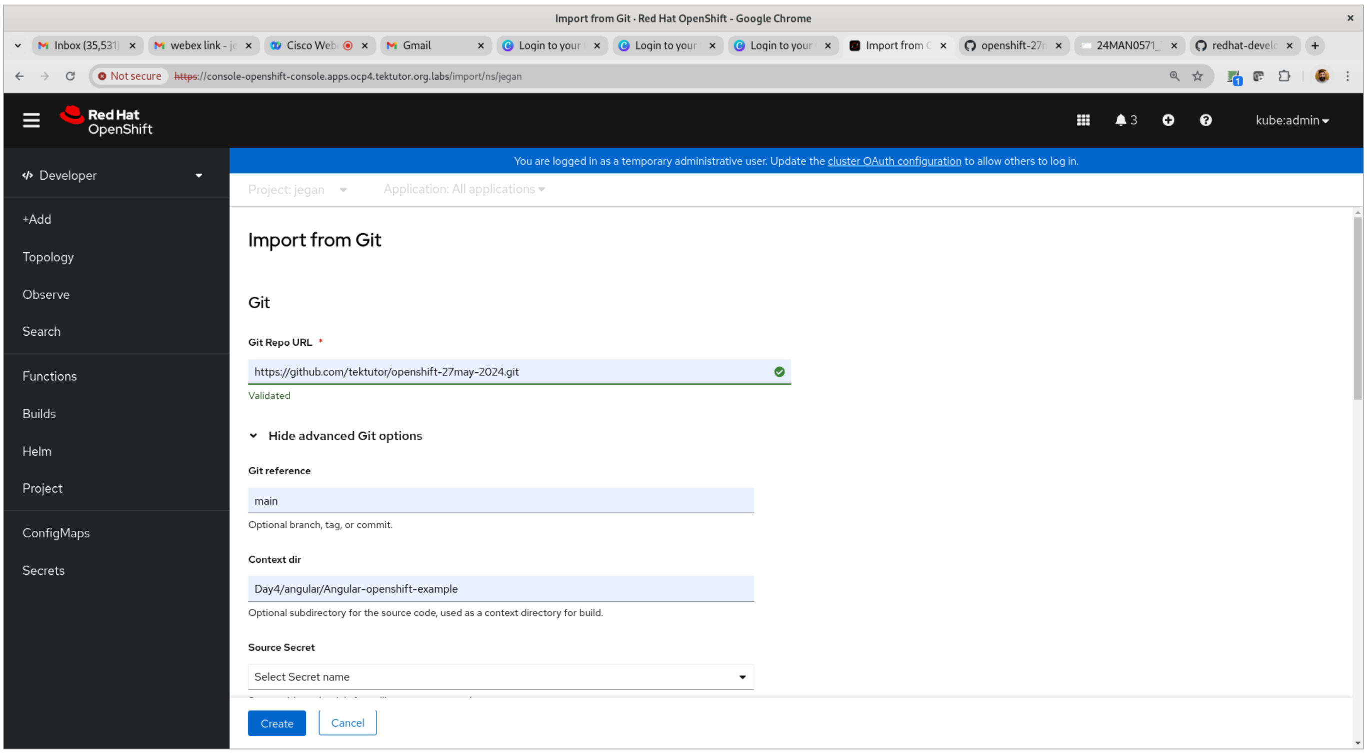
Build from pre-built function sample code

Git Repository

Import from Git

Import code from your Git repository to be built and deployed

Container images


Import from Git · Red Hat OpenShift - Google Chrome
Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/import/ns/jegan
Red Hat OpenShift
kube:admin
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan Application: All applications

Import from Git

Git

Git Repo URL *

https://github.com/tektutor/openshift-27may-2024.git

Validated

Hide advanced Git options

Git reference

main

Optional branch, tag, or commit.

Context dir

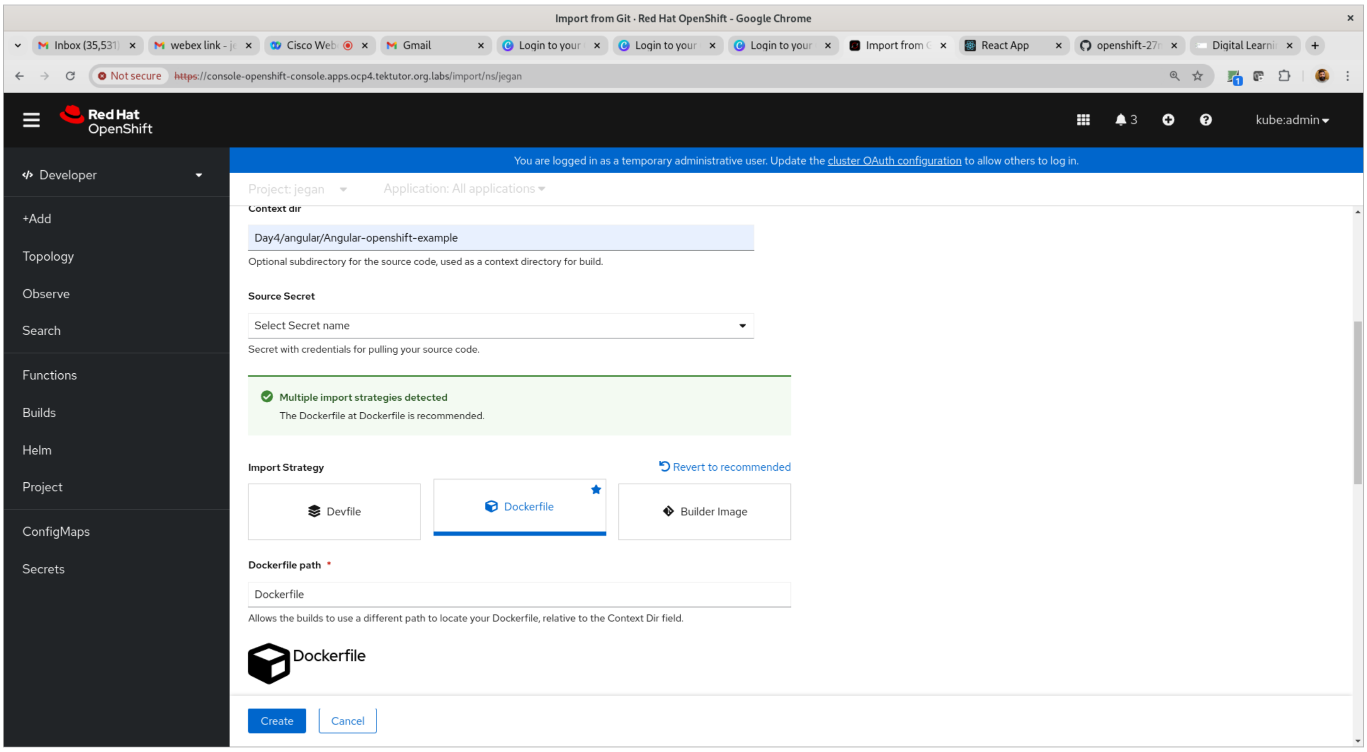
Day4/angular/Angular-openshift-example

Optional subdirectory for the source code, used as a context directory for build.

Source Secret

Select Secret name

Create Cancel


Import from Git · Red Hat OpenShift - Google Chrome
Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/import/ns/jegan
Red Hat OpenShift
kube:admin
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets
Project: jegan
Application: All applications

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Context dir

Day4/angular/Angular-openshift-example

Optional subdirectory for the source code, used as a context directory for build.

Source Secret

Select Secret name

Secret with credentials for pulling your source code.

Multiple import strategies detected

The Dockerfile at Dockerfile is recommended.

Import Strategy — Revert to recommended

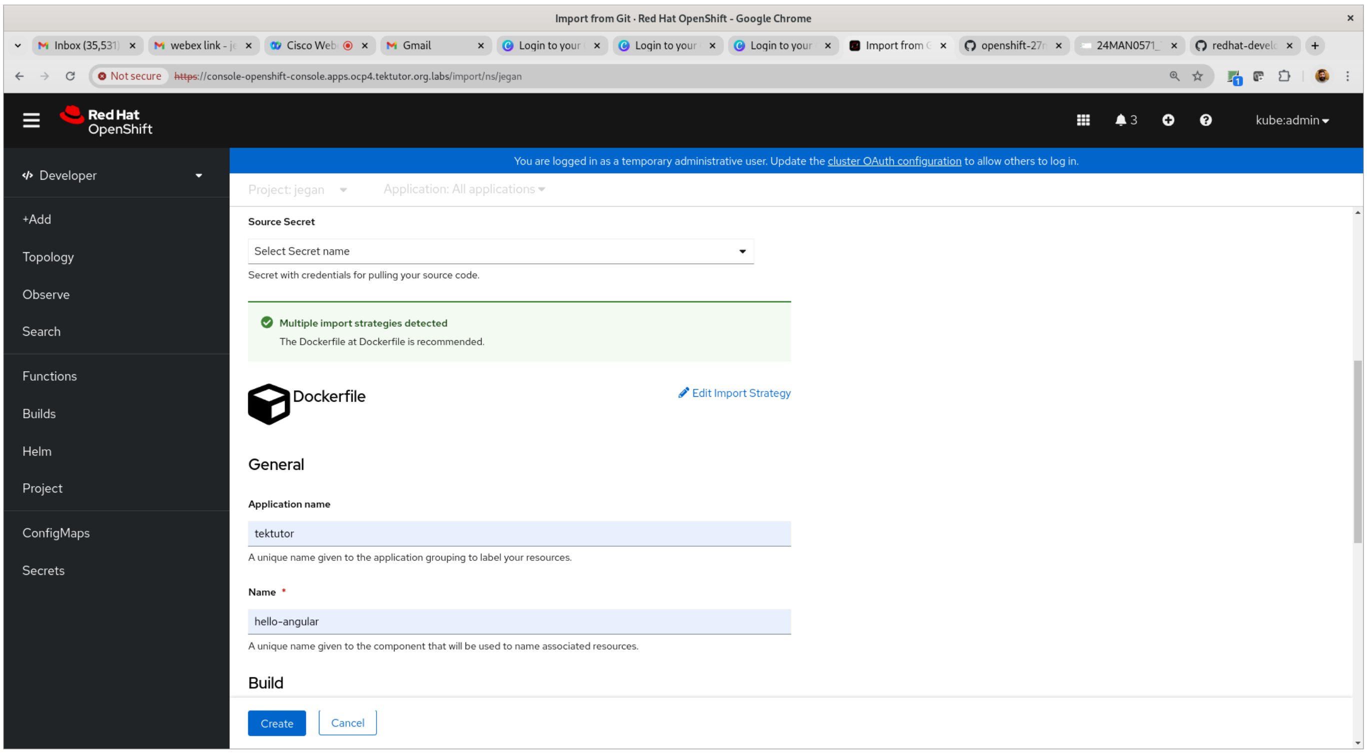
Devfile | Dockerfile | Builder Image

Dockerfile path *

Dockerfile

Allows the builds to use a different path to locate your Dockerfile, relative to the Context Dir field.

Dockerfile


Import from Git · Red Hat OpenShift - Google Chrome
Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/import/ns/jegan
Red Hat OpenShift
kube:admin
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets
Project: jegan
Application: All applications

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Source Secret

Select Secret name

Secret with credentials for pulling your source code.

Multiple import strategies detected

The Dockerfile at Dockerfile is recommended.

Dockerfile — Edit Import Strategy

General

Application name

tektutor

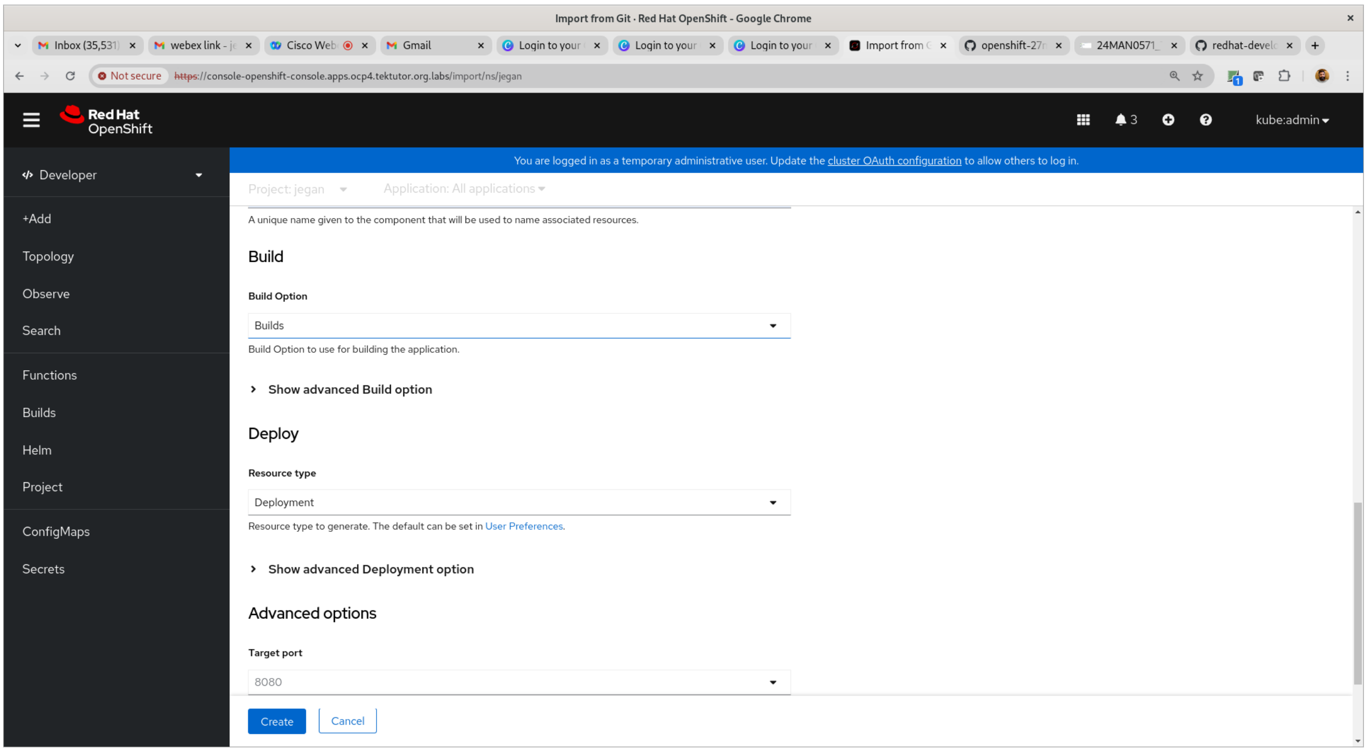
A unique name given to the application grouping to label your resources.

Name *

hello-angular

A unique name given to the component that will be used to name associated resources.

Build



Import from Git · Red Hat OpenShift - Google Chrome

Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/import/ns/jegan

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan Application: All applications

A unique name given to the component that will be used to name associated resources.

Build

Build Option: Builds

Build Option to use for building the application.

Show advanced Build option

Deploy

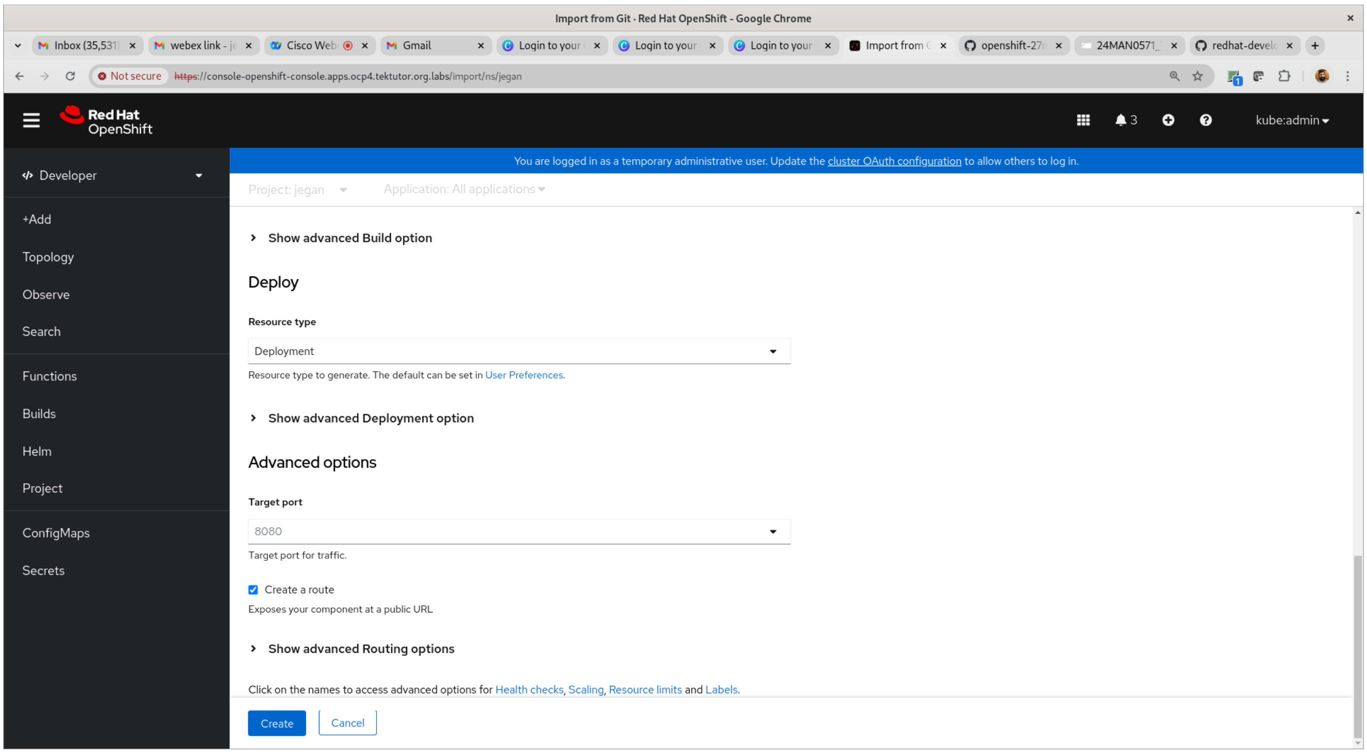
Resource type: Deployment

Resource type to generate. The default can be set in User Preferences.

Show advanced Deployment option

Advanced options

Target port: 8080



Import from Git · Red Hat OpenShift - Google Chrome

Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/import/ns/jegan

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan Application: All applications

Show advanced Build option

Deploy

Resource type: Deployment

Resource type to generate. The default can be set in User Preferences.

Show advanced Deployment option

Advanced options

Target port: 8080

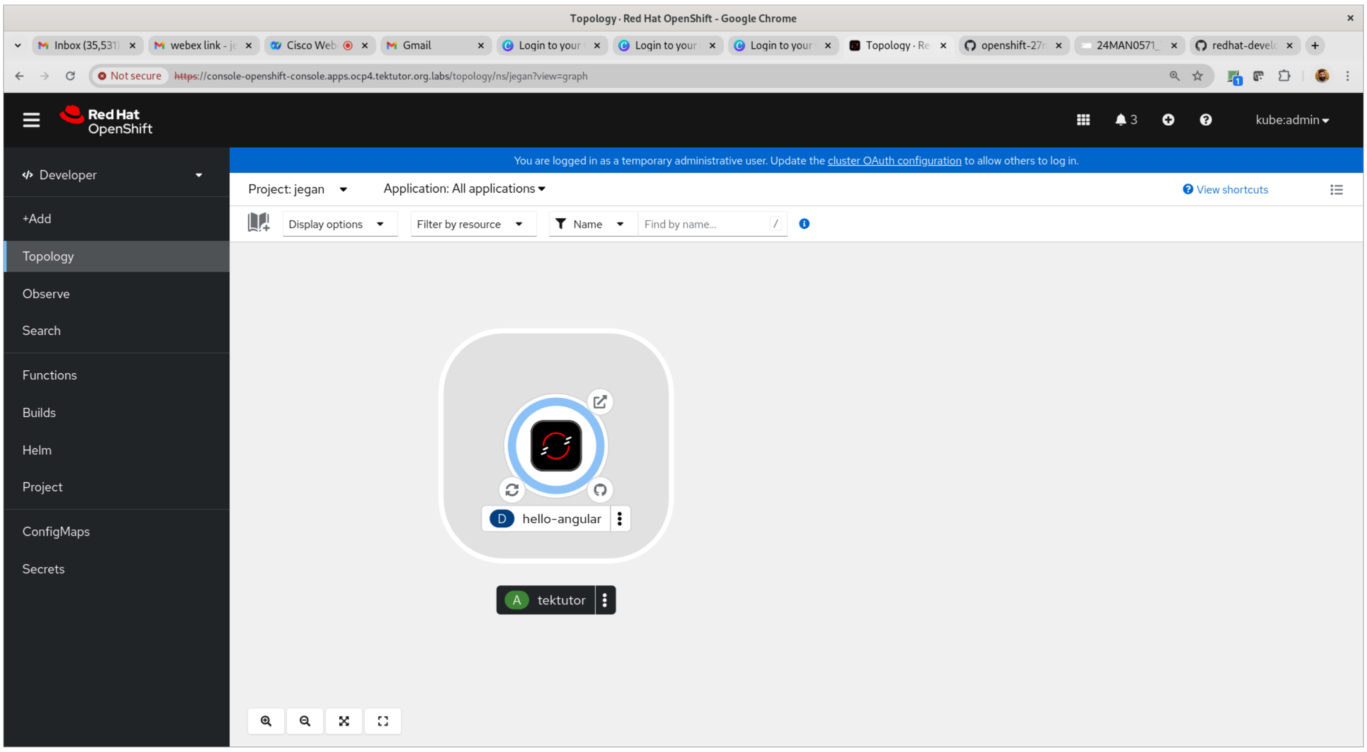
Target port for traffic.

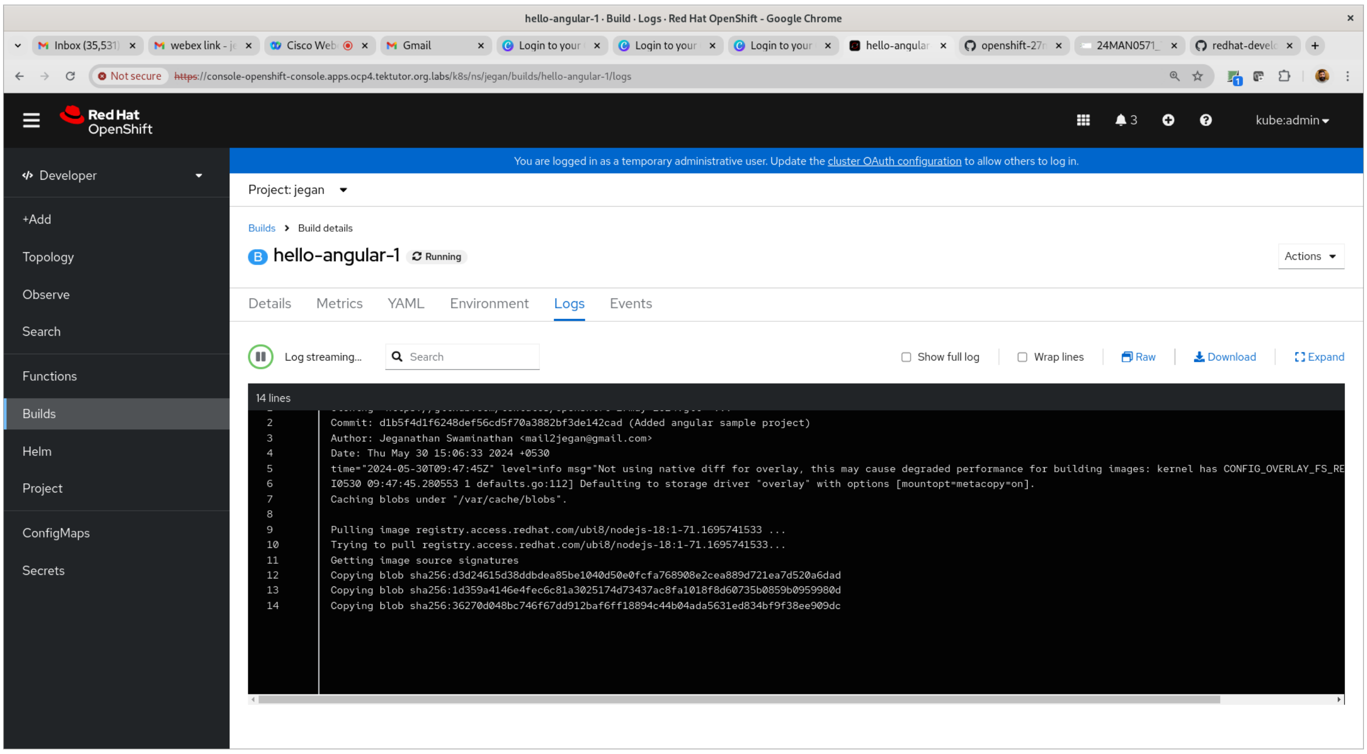
☑ Create a route

Exposes your component at a public URL

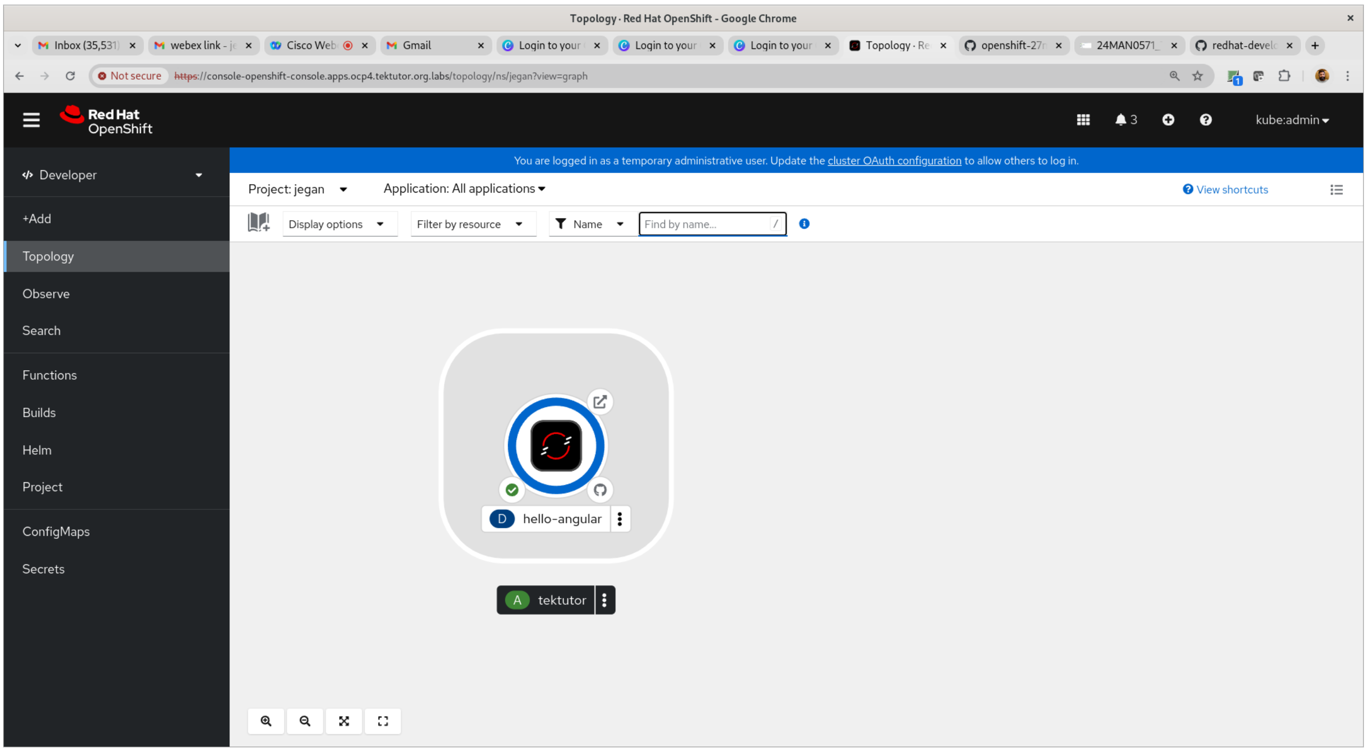
Show advanced Routing options

Click on the names to access advanced options for Health checks, Scaling, Resource limits and Labels.

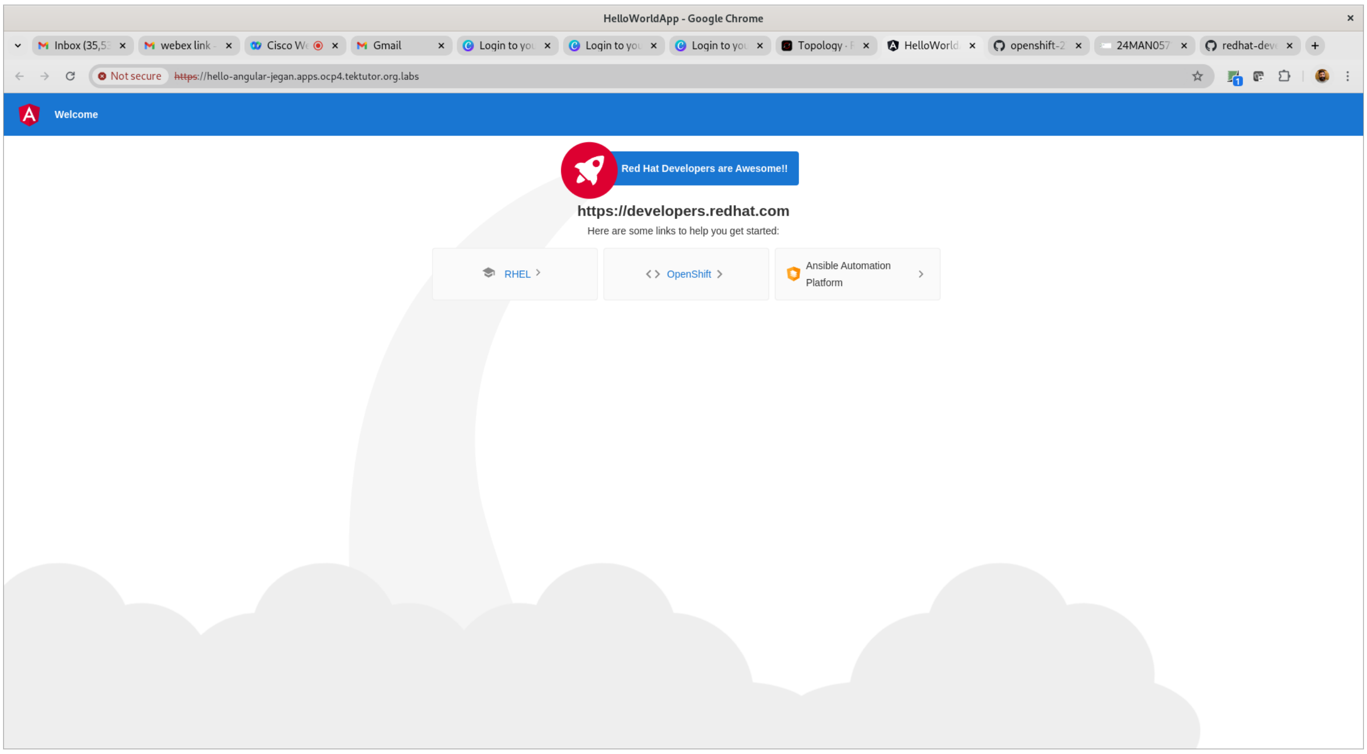

Topology · Red Hat OpenShift - Google Chrome
Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/topology/ns/jegan?view=graph
Red Hat OpenShift
kube:admin
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets
You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.
Project: jegan
Application: All applications
View shortcuts
Display options
Filter by resource
Name
Find by name...
D hello-angular
A tektutor


hello-angular-1 · Build · Logs · Red Hat OpenShift - Google Chrome
Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/k8s/ns/jegan/builds/hello-angular-1/logs
Red Hat OpenShift
kube:admin
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets
You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.
Project: jegan
Builds › Build details
B hello-angular-1 Running
Actions
Details
Metrics
YAML
Environment
Logs
Events
Log streaming...
Search
Show full log
Wrap lines
Raw
Download
Expand
14 lines

```
Commit: d1b5f4d1f6248def56cd5f70a3882bf3de142cad (Added angular sample project)
Author: Jeganathan Swaminathan <mail2jegan@gmail.com>
Date: Thu May 30 15:06:33 2024 +0530
time="2024-05-30T09:47:45Z" level=info msg="Not using native diff for overlay, this may cause degraded performance for building images: kernel has CONFIG_OVERLAY_FS_RE
I0530 09:47:45.280553 1 defaults.go:112] Defaulting to storage driver "overlay" with options [mountopt=metacopy=on].
Caching blobs under "/var/cache/blobs".

Pulling image registry.access.redhat.com/ubi8/nodejs-18:1-71.1695741533 ...
Trying to pull registry.access.redhat.com/ubi8/nodejs-18:1-71.1695741533...
Getting image source signatures
Copying blob sha256:d3d24615d38ddbdea85be1040d50e0fcfa768908e2cea889d721ea7d520a6dad
Copying blob sha256:1d359a4146e4fec6c81a3025174d73437ac8fa1018f8d60735b0859b0959980d
Copying blob sha256:36270d048bc746f67dd912baf6ff18894c44b04ada5631ed834bf9f38ee909dc
```

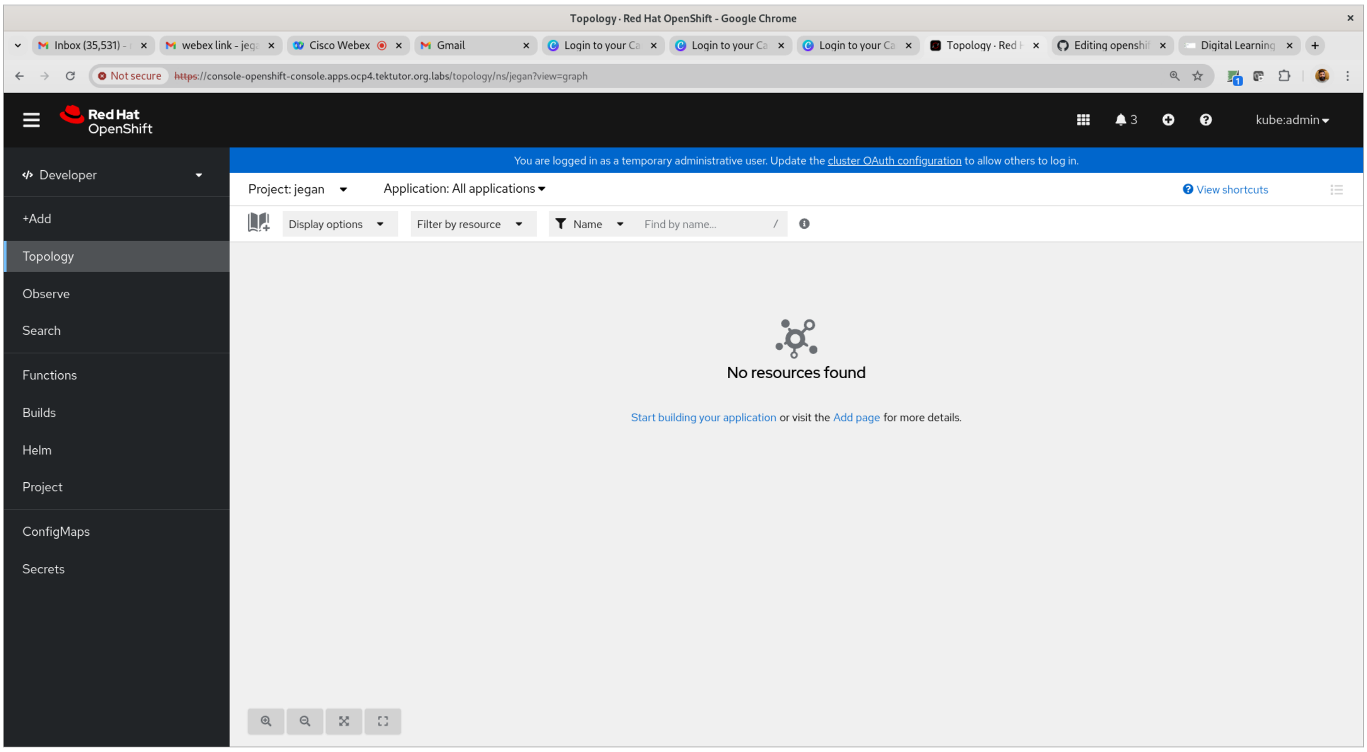


Topology · Red Hat OpenShift - Google Chrome
Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/topology/ns/jegan?view=graph
Red Hat OpenShift
kube:admin
You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets
Project: jegan
Application: All applications
View shortcuts
Display options
Filter by resource
Name
Find by name...
D hello-angular
A tektutor



HelloWorldApp - Google Chrome
Not secure https://hello-angular-jegan.apps.ocp4.tektutor.org.labs
Welcome
Red Hat Developers are Awesome!!
https://developers.redhat.com
Here are some links to help you get started:
RHEL
OpenShift
Ansible Automation Platform

Lab - Deploying ReactJS application in Openshift from webconsole



Topology · Red Hat OpenShift - Google Chrome

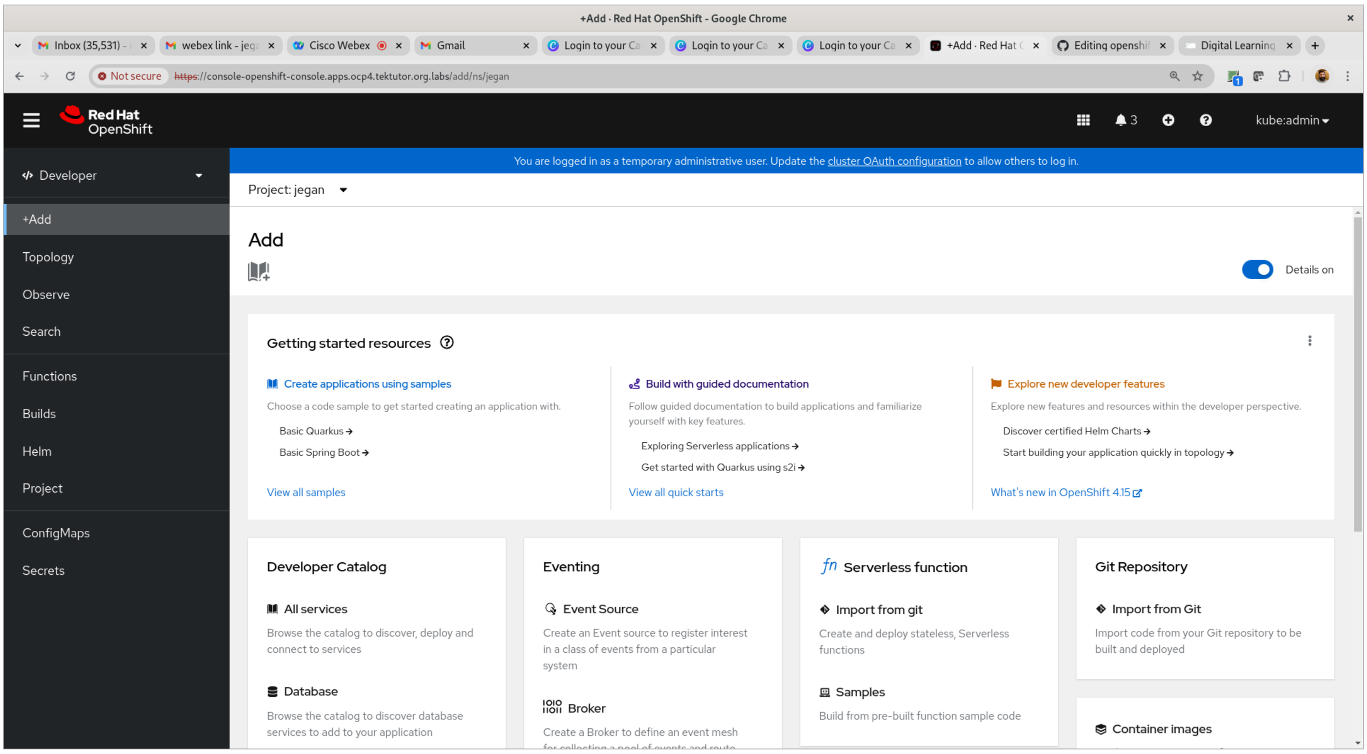
Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/topology/ns/jegan?view=graph

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan   Application: All applications

No resources found

Start building your application or visit the Add page for more details.



+Add · Red Hat OpenShift - Google Chrome

Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/add/ns/jegan

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan

Add

Details on

Getting started resources

Create applications using samples

Choose a code sample to get started creating an application with.

- Basic Quarkus →
- Basic Spring Boot →

View all samples

Build with guided documentation

Follow guided documentation to build applications and familiarize yourself with key features.

- Exploring Serverless applications →
- Get started with Quarkus using s2i →

View all quick starts

Explore new developer features

Explore new features and resources within the developer perspective.

- Discover certified Helm Charts →
- Start building your application quickly in topology →

What's new in OpenShift 4.15

Developer Catalog

All services

Browse the catalog to discover, deploy and connect to services

Database

Browse the catalog to discover database services to add to your application

Eventing

Event Source

Create an Event source to register interest in a class of events from a particular system

Broker

Create a Broker to define an event mesh for collecting a pool of events and route

Serverless function

Import from git

Create and deploy stateless, Serverless functions

Samples

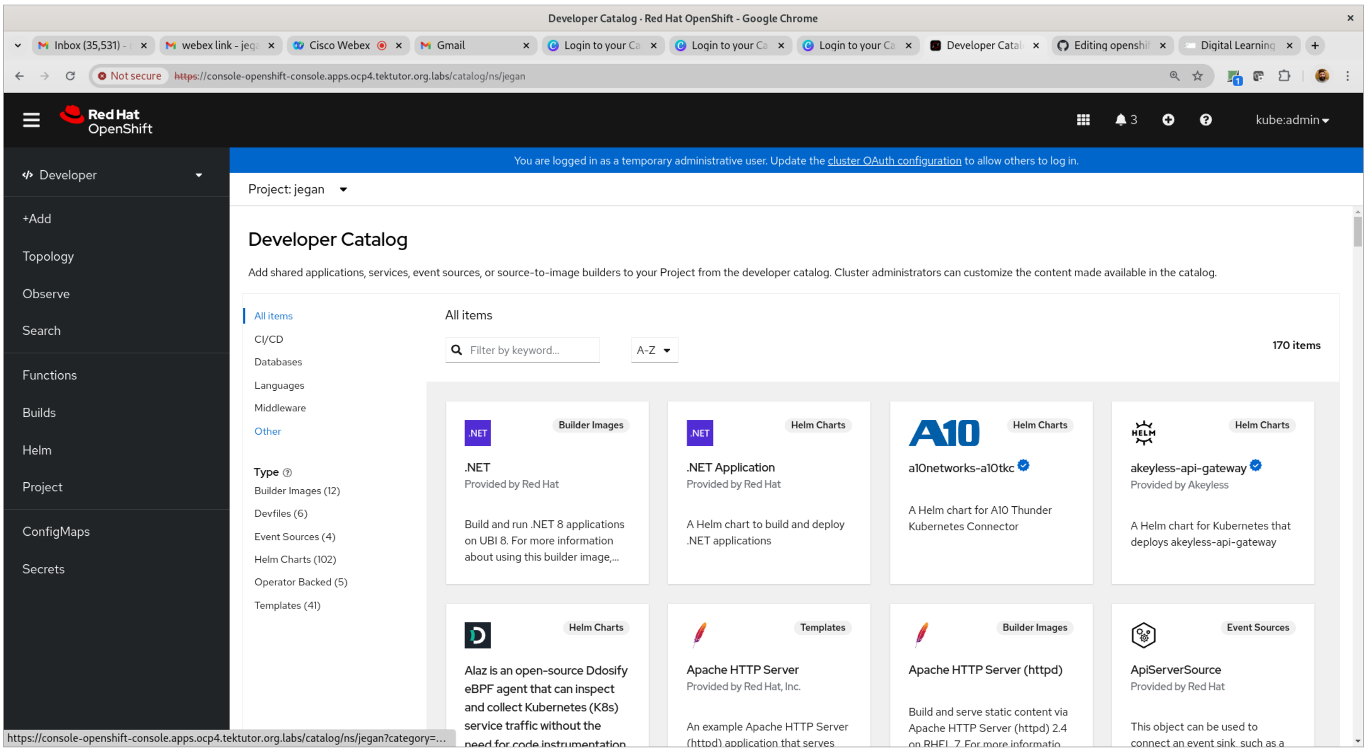
Build from pre-built function sample code

Git Repository

Import from Git

Import code from your Git repository to be built and deployed

Container images



Developer Catalog - Red Hat OpenShift - Google Chrome
Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/catalog/ns/jegan
Red Hat OpenShift
kube:admin
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan

Developer Catalog

Add shared applications, services, event sources, or source-to-image builders to your Project from the developer catalog. Cluster administrators can customize the content made available in the catalog.

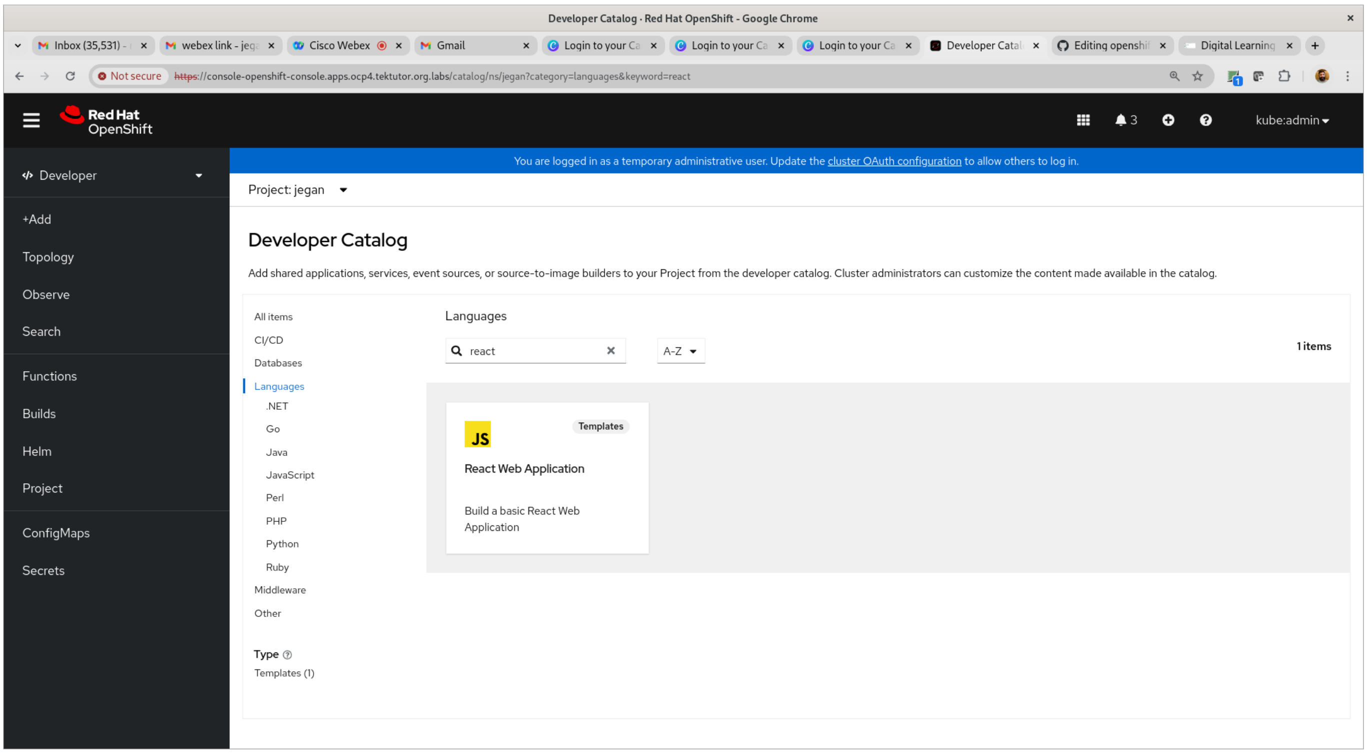
Type
Builder Images (12)
Devfiles (6)
Event Sources (4)
Helm Charts (102)
Operator Backed (5)
Templates (41)

All items

Filter by keyword... A-Z 170 items

Builder Images .NET Provided by Red Hat Build and run .NET 8 applications on UBI 8. For more information about using this builder image,...	Helm Charts .NET Application Provided by Red Hat A Helm chart to build and deploy .NET applications	Helm Charts a10networks-a10tkc A Helm chart for A10 Thunder Kubernetes Connector	Helm Charts akeyless-api-gateway Provided by Akeyless A Helm chart for Kubernetes that deploys akeyless-api-gateway
Helm Charts Alaz is an open-source Ddosify eBPF agent that can inspect and collect Kubernetes (K8s) service traffic without the need for code instrumentation	Templates Apache HTTP Server Provided by Red Hat, Inc. An example Apache HTTP Server (httpd) application that serves	Builder Images Apache HTTP Server (httpd) Build and serve static content via Apache HTTP Server (httpd) 2.4 on RHEL 7. For more informatio	Event Sources ApiServerSource Provided by Red Hat This object can be used to connect an event sink, such as a

https://console-openshift-console.apps.ocp4.tektutor.org.labs/catalog/ns/jegan?category=...



Developer Catalog - Red Hat OpenShift - Google Chrome
Not secure https://console-openshift-console.apps.ocp4.tektutor.org.labs/catalog/ns/jegan?category=languages&keyword=react
Red Hat OpenShift
kube:admin
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan

Developer Catalog

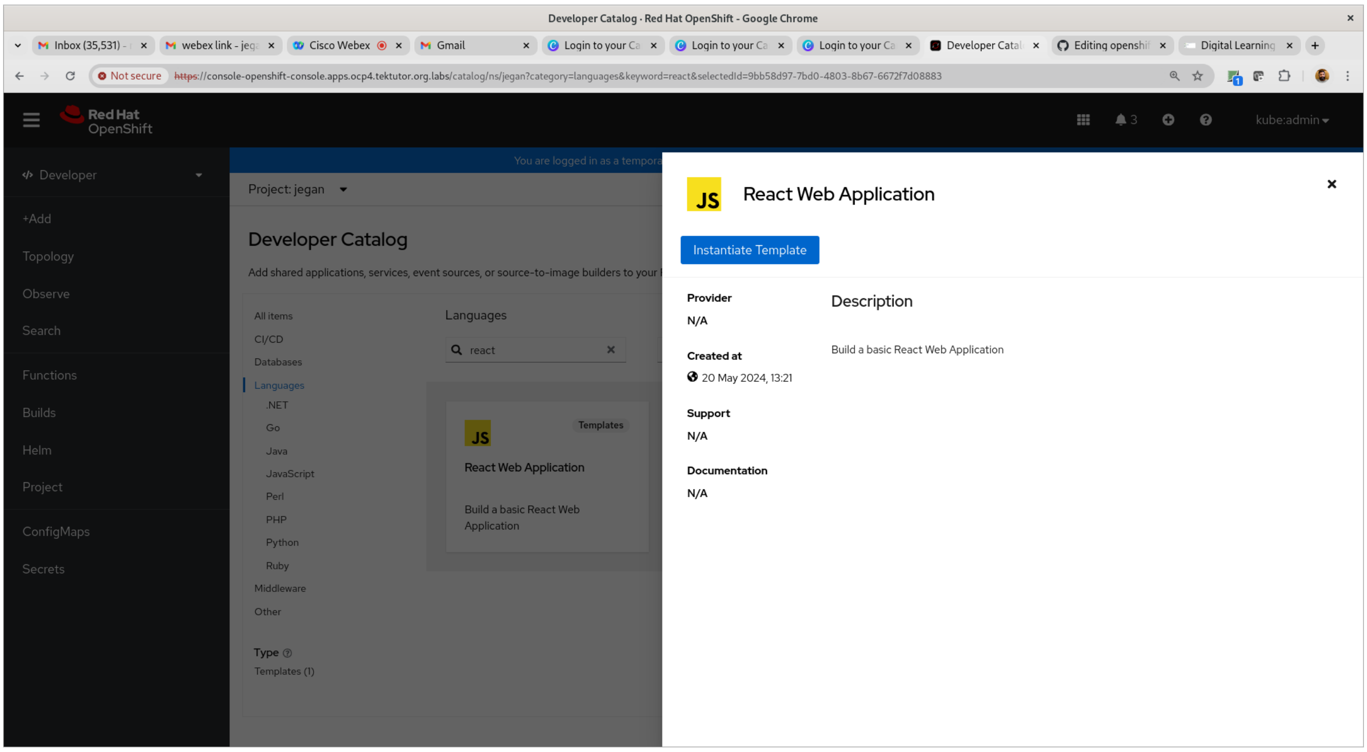
Add shared applications, services, event sources, or source-to-image builders to your Project from the developer catalog. Cluster administrators can customize the content made available in the catalog.

Type
Templates (1)

Languages

react A-Z 1 items

Templates
React Web Application
Build a basic React Web Application


Developer Catalog · Red Hat OpenShift - Google Chrome
Not secure
https://console-openshift-console.apps.ocp4.tektutor.org.labs/catalog/ns/jegan?category=languages&keyword=react&selectedId=9bb58d97-7bd0-4803-8b67-6672f7d08883
Red Hat OpenShift
kube:admin
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets
Project: jegan
Developer Catalog
Add shared applications, services, event sources, or source-to-image builders to your
All items
CI/CD
Databases
Languages
.NET
Go
Java
JavaScript
Perl
PHP
Python
Ruby
Middleware
Other
Type
Templates (1)
react
Templates
Instantiate Template

React Web Application

Build a basic React Web Application

Provider
N/A

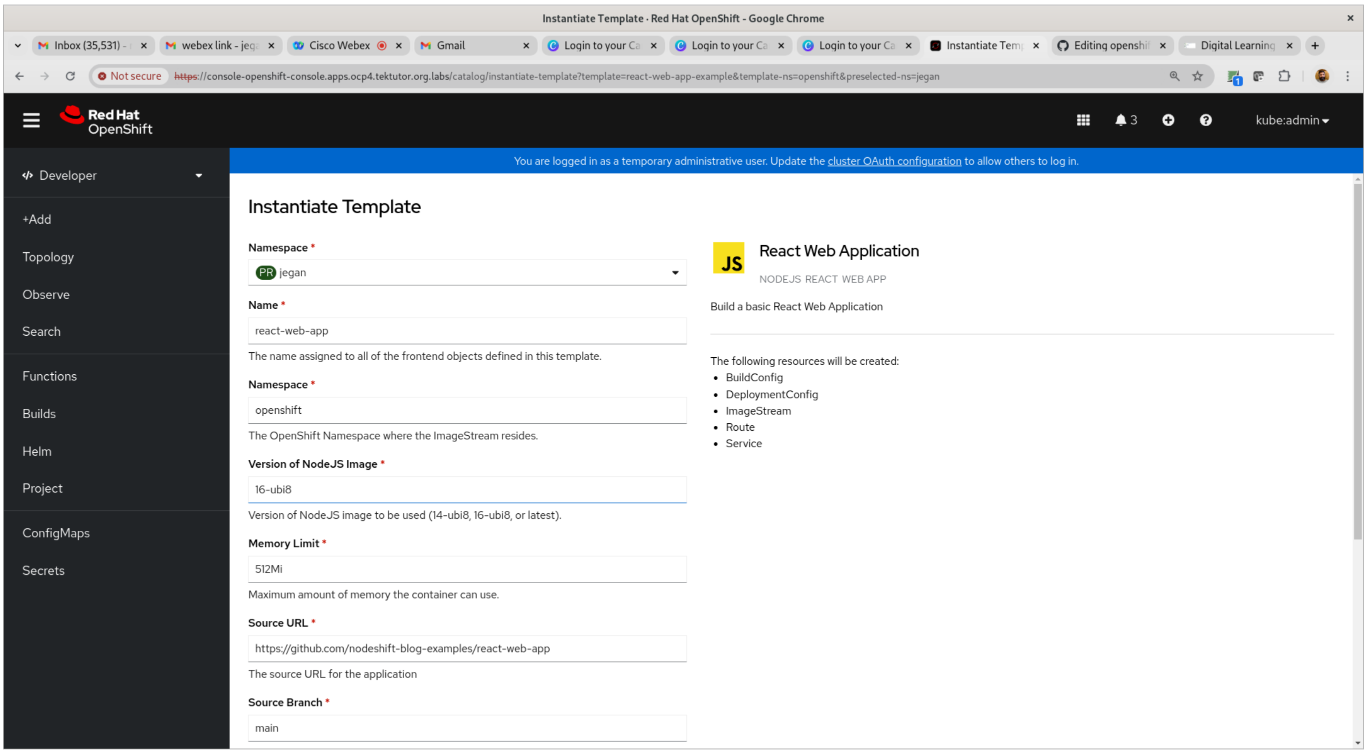
Created at
20 May 2024, 13:21

Support
N/A

Documentation
N/A

Description

Build a basic React Web Application


Instantiate Template · Red Hat OpenShift - Google Chrome
Not secure
https://console-openshift-console.apps.ocp4.tektutor.org.labs/catalog/instantiate-template?template=react-web-app-example&template-ns=openshift&preselected-ns=jegan
Red Hat OpenShift
kube:admin
Developer
+Add
Topology
Observe
Search
Functions
Builds
Helm
Project
ConfigMaps
Secrets

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Instantiate Template

Namespace *
PR jegan

Name *
react-web-app
The name assigned to all of the frontend objects defined in this template.

Namespace *
openshift
The OpenShift Namespace where the ImageStream resides.

Version of NodeJS Image *
16-ubi8
Version of NodeJS image to be used (14-ubi8, 16-ubi8, or latest).

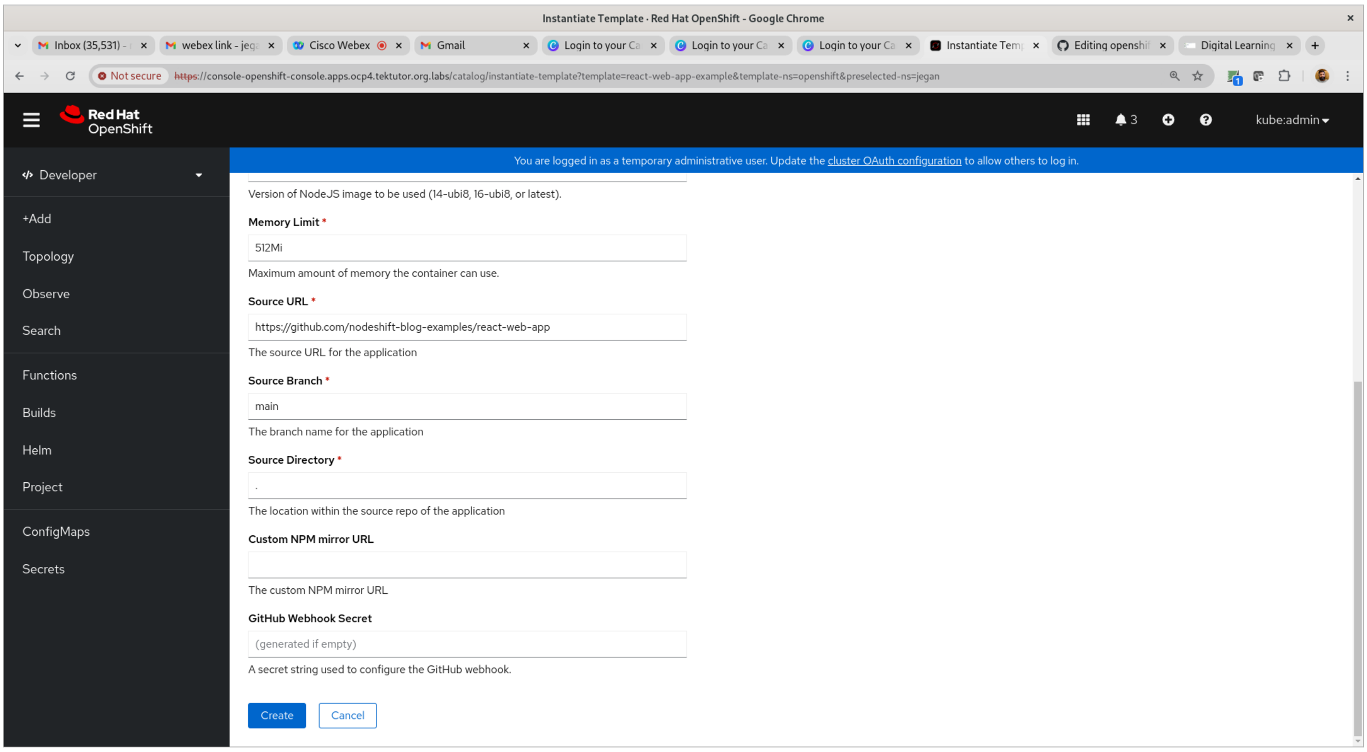
Memory Limit *
512Mi
Maximum amount of memory the container can use.

Source URL *
https://github.com/nodeshift-blog-examples/react-web-app
The source URL for the application

Source Branch *
main

React Web Application
NODEJS REACT WEB APP
Build a basic React Web Application

The following resources will be created:
- BuildConfig
- DeploymentConfig
- ImageStream
- Route
- Service



https://console-openshift-console.apps.ocp4.tektutor.org.labs/catalog/instantiate-template?template=react-web-app-example&template-ns=openshift&preselected-ns=jegan

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Version of NodeJS image to be used (14-ubi8, 16-ubi8, or latest).

Memory Limit *

512Mi

Maximum amount of memory the container can use.

Source URL *

https://github.com/nodeshift-blog-examples/react-web-app

The source URL for the application

Source Branch *

main

The branch name for the application

Source Directory *

.

The location within the source repo of the application

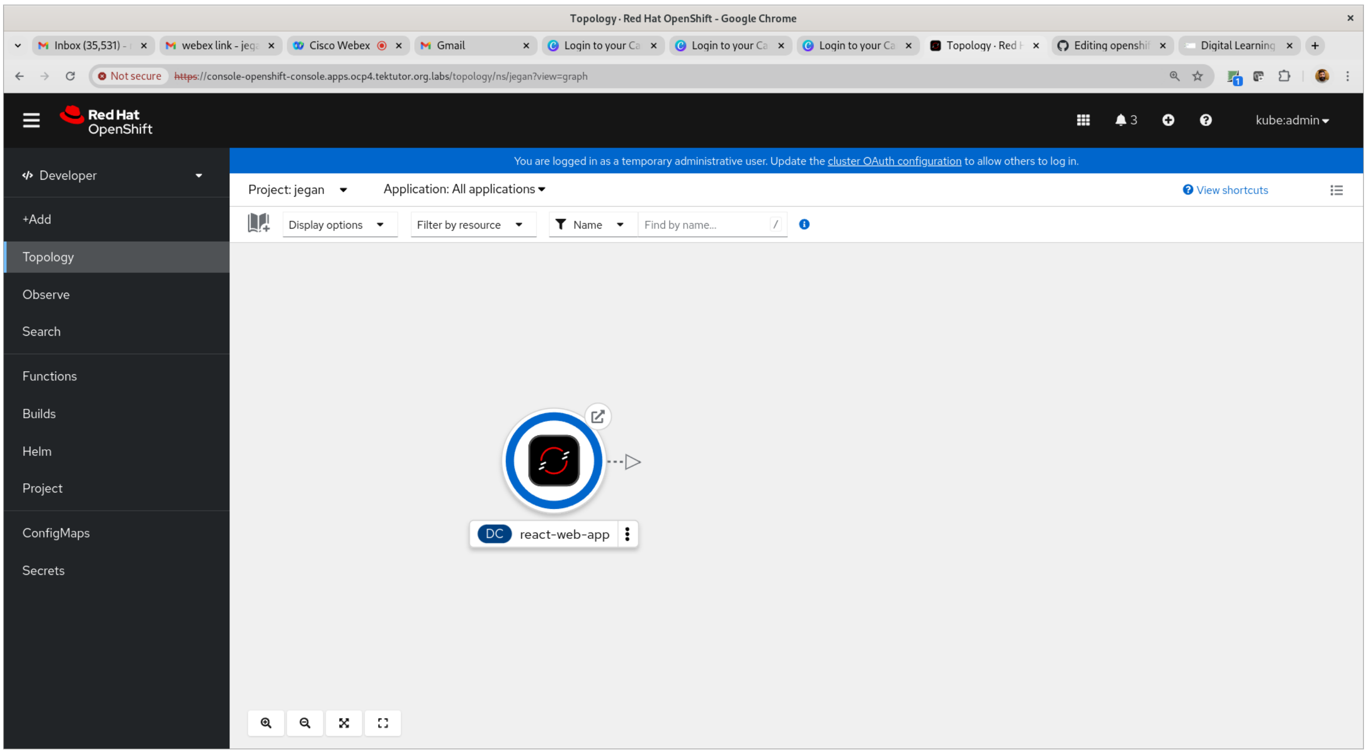
Custom NPM mirror URL

The custom NPM mirror URL

GitHub Webhook Secret

(generated if empty)

A secret string used to configure the GitHub webhook.

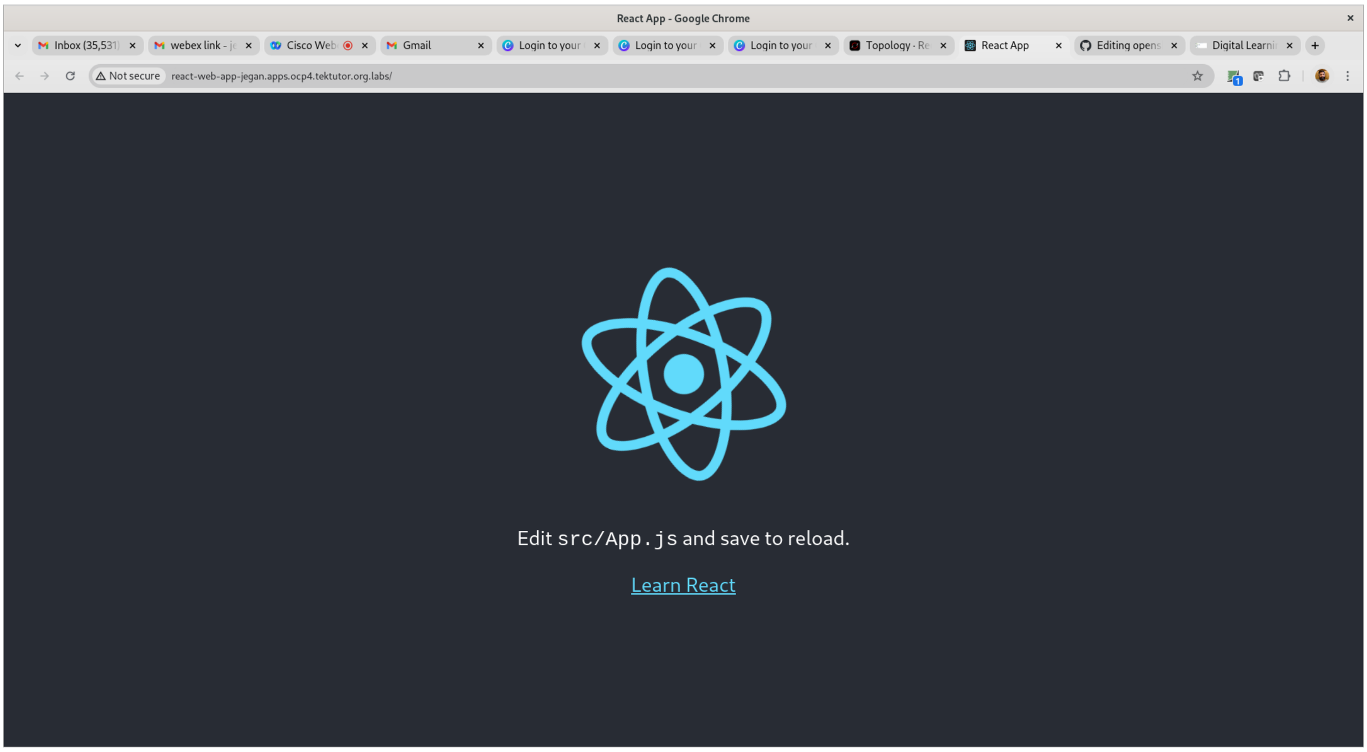


https://console-openshift-console.apps.ocp4.tektutor.org.labs/topology/ns/jegan?view=graph

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan

Application: All applications



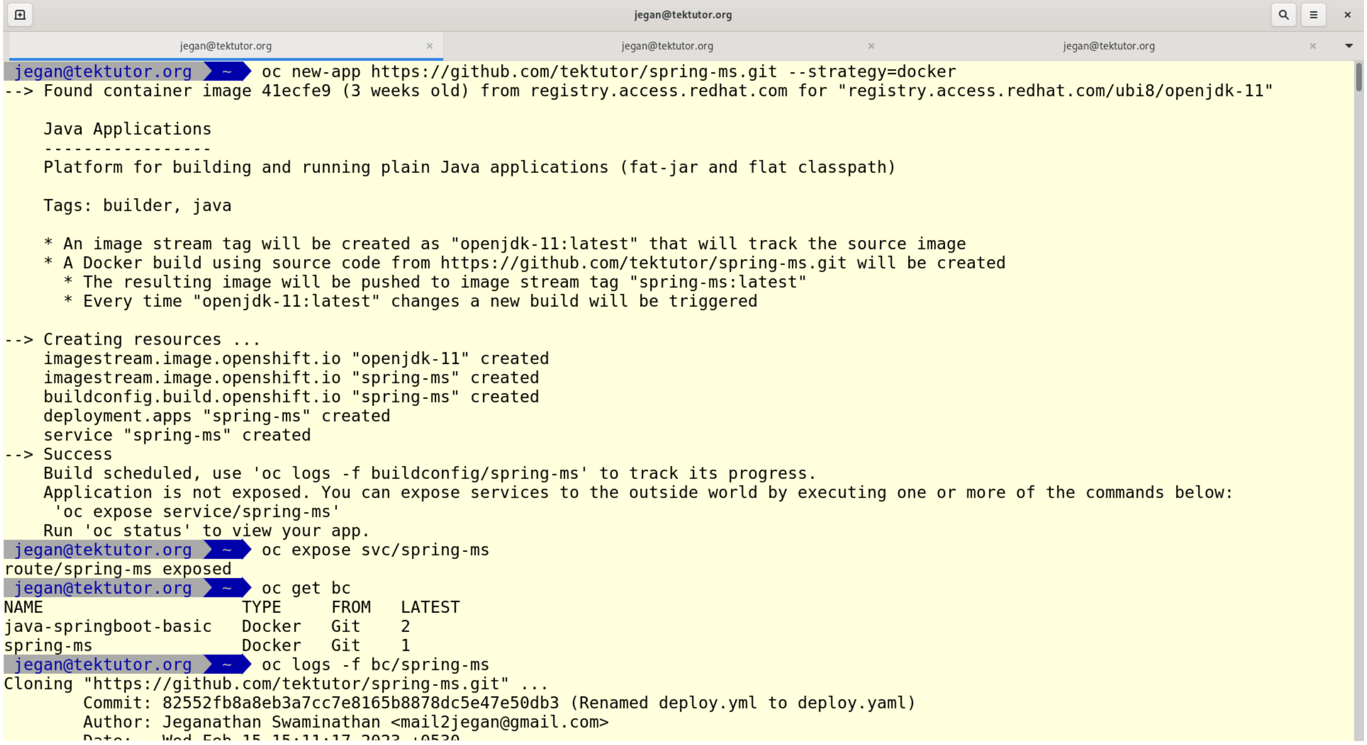
Edit src/App.js and save to reload.

Learn React

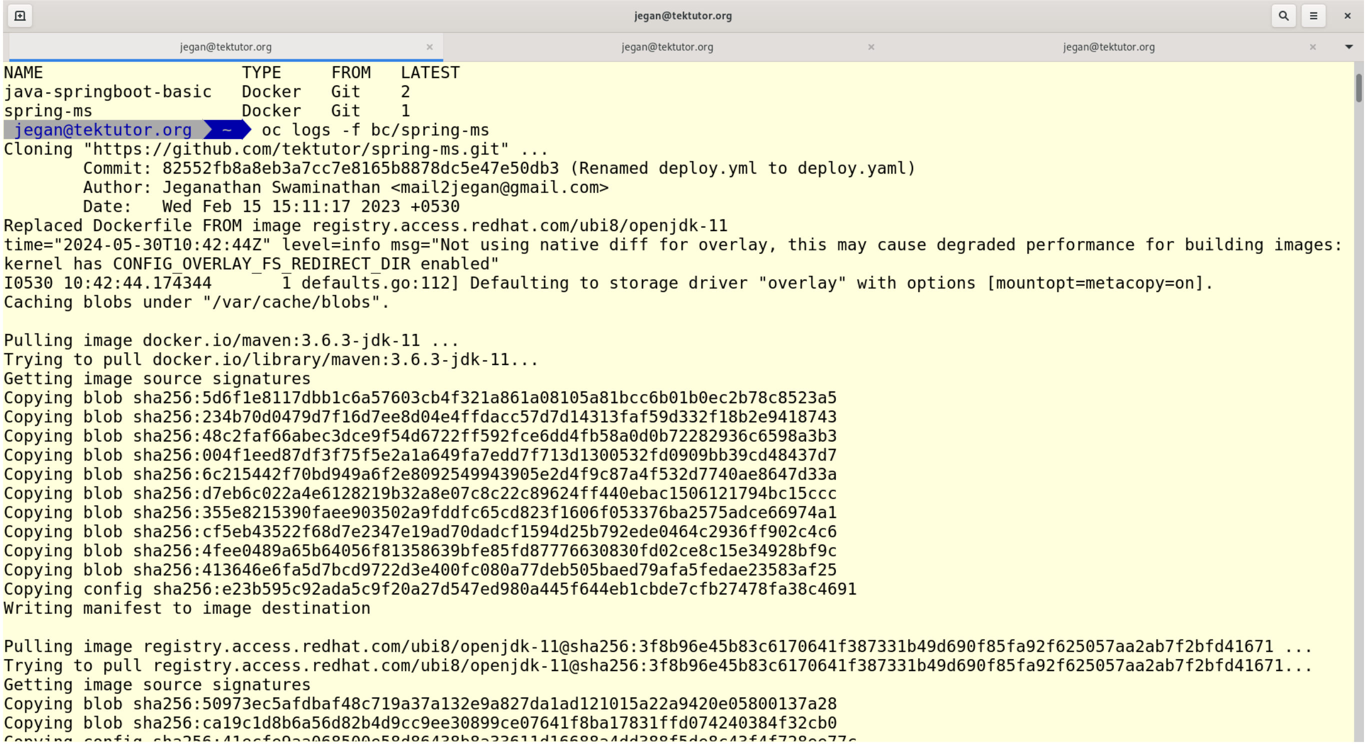
Lab - Deploying a Java springboot application from GitHub source code into Openshift

```
oc new-app https://github.com/tektutor/spring-ms.git --strategy=docker
oc expose svc/spring-ms
oc get bc
oc logs -f bc/spring-ms
```

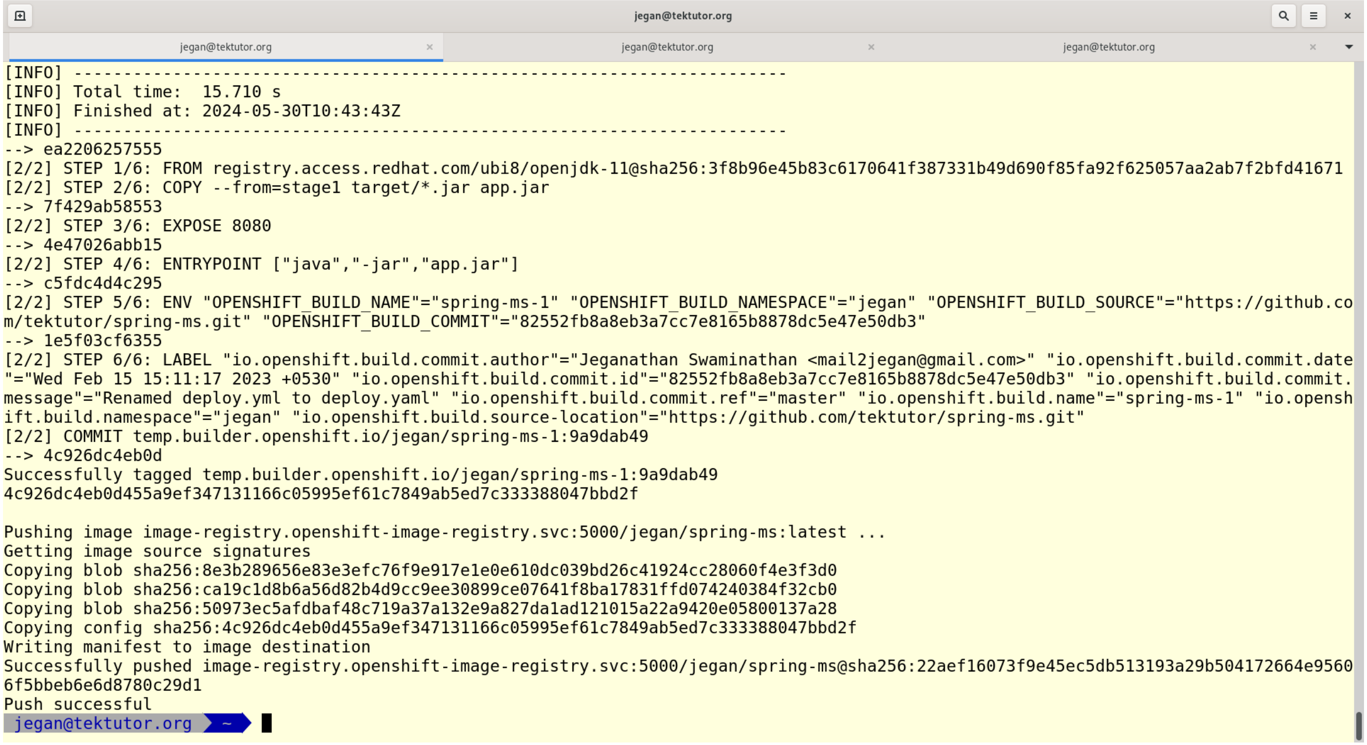
Expected output



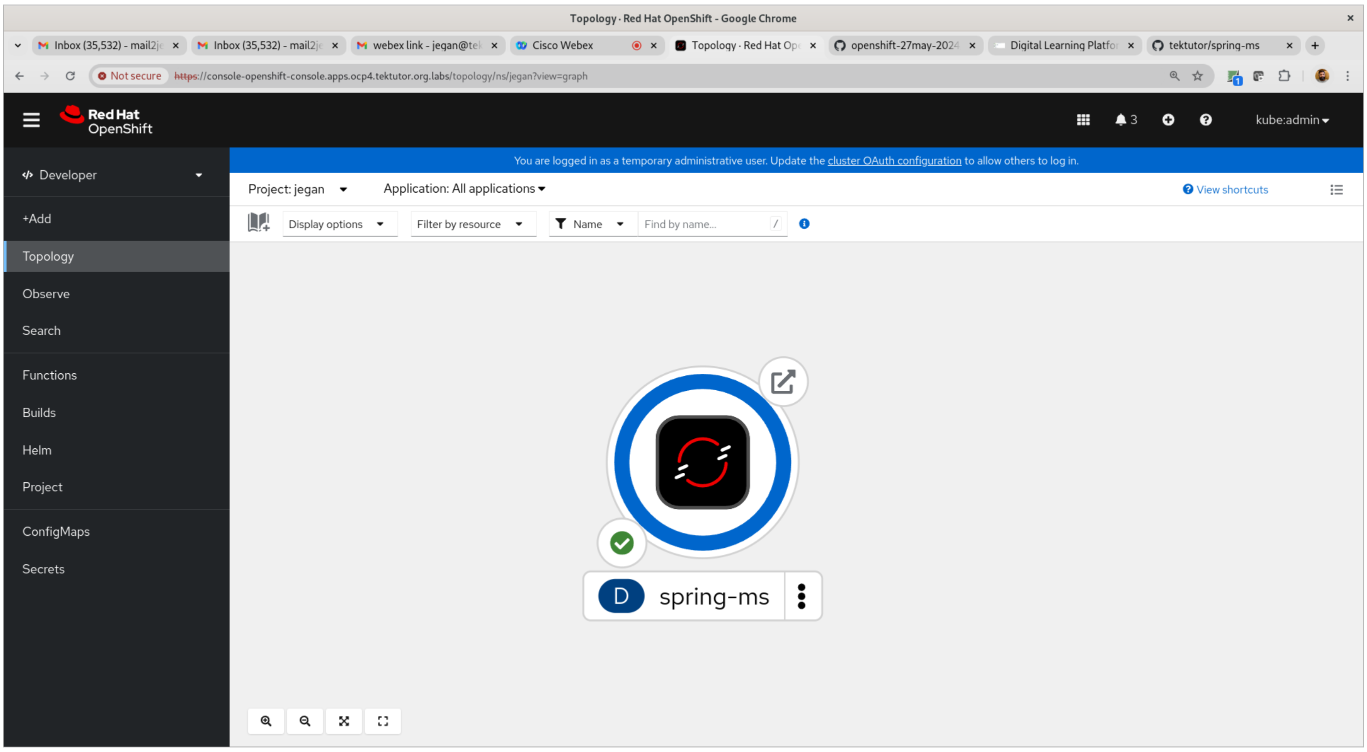
```
 jegan@tektutor.org  ~  oc new-app https://github.com/tektutor/spring-ms.git --strategy=docker
--> Found container image 41ecfe9 (3 weeks old) from registry.access.redhat.com for "registry.access.redhat.com/ubi8/openjdk-11"

    Java Applications
    -----------------
    Platform for building and running plain Java applications (fat-jar and flat classpath)

    Tags: builder, java

    * An image stream tag will be created as "openjdk-11:latest" that will track the source image
    * A Docker build using source code from https://github.com/tektutor/spring-ms.git will be created
      * The resulting image will be pushed to image stream tag "spring-ms:latest"
      * Every time "openjdk-11:latest" changes a new build will be triggered

--> Creating resources ...
    imagestream.image.openshift.io "openjdk-11" created
    imagestream.image.openshift.io "spring-ms" created
    buildconfig.build.openshift.io "spring-ms" created
    deployment.apps "spring-ms" created
    service "spring-ms" created
--> Success
    Build scheduled, use 'oc logs -f buildconfig/spring-ms' to track its progress.
    Application is not exposed. You can expose services to the outside world by executing one or more of the commands below:
     'oc expose service/spring-ms'
    Run 'oc status' to view your app.
 jegan@tektutor.org  ~  oc expose svc/spring-ms
route/spring-ms exposed
 jegan@tektutor.org  ~  oc get bc
NAME                    TYPE     FROM   LATEST
java-springboot-basic   Docker   Git    2
spring-ms               Docker   Git    1
 jegan@tektutor.org  ~  oc logs -f bc/spring-ms
Cloning "https://github.com/tektutor/spring-ms.git" ...
        Commit: 82552fb8a8eb3a7cc7e8165b8878dc5e47e50db3 (Renamed deploy.yml to deploy.yaml)
        Author: Jeganathan Swaminathan <mail2jegan@gmail.com>
        Date:   Wed Feb 15 15:11:17 2023 +0530
```



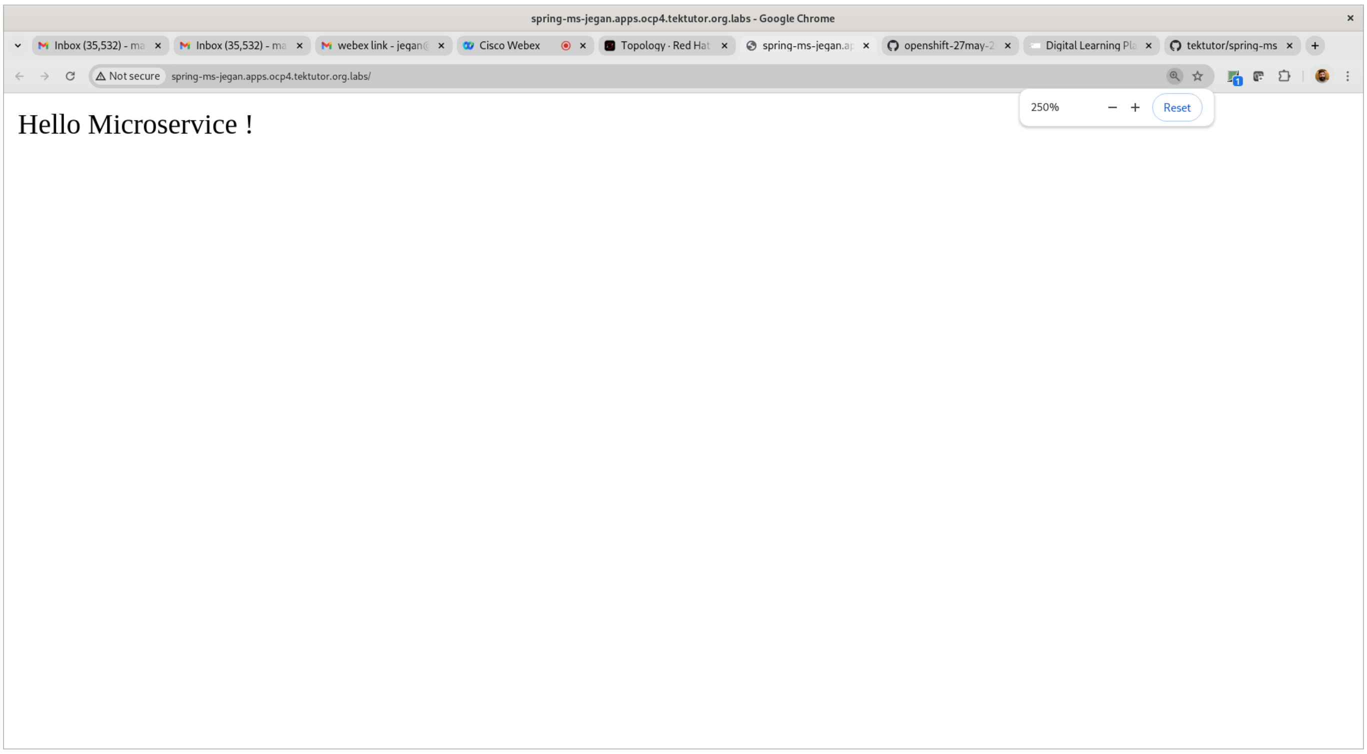
```
NAME                    TYPE     FROM   LATEST
java-springboot-basic   Docker   Git    2
spring-ms               Docker   Git    1
 jegan@tektutor.org  ~  oc logs -f bc/spring-ms
Cloning "https://github.com/tektutor/spring-ms.git" ...
        Commit: 82552fb8a8eb3a7cc7e8165b8878dc5e47e50db3 (Renamed deploy.yml to deploy.yaml)
        Author: Jeganathan Swaminathan <mail2jegan@gmail.com>
        Date:   Wed Feb 15 15:11:17 2023 +0530
Replaced Dockerfile FROM image registry.access.redhat.com/ubi8/openjdk-11
time="2024-05-30T10:42:44Z" level=info msg="Not using native diff for overlay, this may cause degraded performance for building images:
kernel has CONFIG_OVERLAY_FS_REDIRECT_DIR enabled"
I0530 10:42:44.174344       1 defaults.go:112] Defaulting to storage driver "overlay" with options [mountopt=metacopy=on].
Caching blobs under "/var/cache/blobs".

Pulling image docker.io/maven:3.6.3-jdk-11 ...
Trying to pull docker.io/library/maven:3.6.3-jdk-11...
Getting image source signatures
Copying blob sha256:5d6f1e8117dbb1c6a57603cb4f321a861a08105a81bcc6b01b0ec2b78c8523a5
Copying blob sha256:234b70d0479d7f16d7ee8d04e4ffdacc57d7d14313faf59d332f18b2e9418743
Copying blob sha256:48c2faf66abec3dce9f54d6722ff592fce6dd4fb58a0d0b72282936c6598a3b3
Copying blob sha256:004f1eed87df3f75f5e2a1a649fa7edd7f713d1300532fd0909bb39cd48437d7
Copying blob sha256:6c215442f70bd949a6f2e8092549943905e2d4f9c87a4f532d7740ae8647d33a
Copying blob sha256:d7eb6c022a4e6128219b32a8e07c8c22c89624ff440ebac1506121794bc15ccc
Copying blob sha256:355e8215390faee903502a9fddfc65cd823f1606f053376ba2575adce66974a1
Copying blob sha256:cf5eb43522f68d7e2347e19ad70dadcf1594d25b792ede0464c2936ff902c4c6
Copying blob sha256:4fee0489a65b64056f81358639bfe85fd87776630830fd02ce8c15e34928bf9c
Copying blob sha256:413646e6fa5d7bcd9722d3e400fc080a77deb505baed79afa5fedae23583af25
Copying config sha256:e23b595c92ada5c9f20a27d547ed980a445f644eb1cbde7cfb27478fa38c4691
Writing manifest to image destination

Pulling image registry.access.redhat.com/ubi8/openjdk-11@sha256:3f8b96e45b83c6170641f387331b49d690f85fa92f625057aa2ab7f2bfd41671 ...
Trying to pull registry.access.redhat.com/ubi8/openjdk-11@sha256:3f8b96e45b83c6170641f387331b49d690f85fa92f625057aa2ab7f2bfd41671...
Getting image source signatures
Copying blob sha256:50973ec5afdbaf48c719a37a132e9a827da1ad121015a22a9420e05800137a28
Copying blob sha256:ca19c1d8b6a56d82b4d9cc9ee30899ce07641f8ba17831ffd074240384f32cb0
```



```
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  15.710 s
[INFO] Finished at: 2024-05-30T10:43:43Z
[INFO] ------------------------------------------------------------------------
--> ea2206257555
[2/2] STEP 1/6: FROM registry.access.redhat.com/ubi8/openjdk-11@sha256:3f8b96e45b83c6170641f387331b49d690f85fa92f625057aa2ab7f2bfd41671
[2/2] STEP 2/6: COPY --from=stage1 target/*.jar app.jar
--> 7f429ab58553
[2/2] STEP 3/6: EXPOSE 8080
--> 4e47026abb15
[2/2] STEP 4/6: ENTRYPOINT ["java","-jar","app.jar"]
--> c5fdc4d4c295
[2/2] STEP 5/6: ENV "OPENSHIFT_BUILD_NAME"="spring-ms-1" "OPENSHIFT_BUILD_NAMESPACE"="jegan" "OPENSHIFT_BUILD_SOURCE"="https://github.co
m/tektutor/spring-ms.git" "OPENSHIFT_BUILD_COMMIT"="82552fb8a8eb3a7cc7e8165b8878dc5e47e50db3"
--> 1e5f03cf6355
[2/2] STEP 6/6: LABEL "io.openshift.build.commit.author"="Jeganathan Swaminathan <mail2jegan@gmail.com>" "io.openshift.build.commit.date
"="Wed Feb 15 15:11:17 2023 +0530" "io.openshift.build.commit.id"="82552fb8a8eb3a7cc7e8165b8878dc5e47e50db3" "io.openshift.build.commit.
message"="Renamed deploy.yml to deploy.yaml" "io.openshift.build.commit.ref"="master" "io.openshift.build.name"="spring-ms-1" "io.opensh
ift.build.namespace"="jegan" "io.openshift.build.source-location"="https://github.com/tektutor/spring-ms.git"
[2/2] COMMIT temp.builder.openshift.io/jegan/spring-ms-1:9a9dab49
--> 4c926dc4eb0d
Successfully tagged temp.builder.openshift.io/jegan/spring-ms-1:9a9dab49
4c926dc4eb0d455a9ef347131166c05995ef61c7849ab5ed7c333388047bbd2f

Pushing image image-registry.openshift-image-registry.svc:5000/jegan/spring-ms:latest ...
Getting image source signatures
Copying blob sha256:8e3b289656e83e3efc76f9e917e1e0e610dc039bd26c41924cc28060f4e3f3d0
Copying blob sha256:ca19c1d8b6a56d82b4d9cc9ee30899ce07641f8ba17831ffd074240384f32cb0
Copying blob sha256:50973ec5afdbaf48c719a37a132e9a827da1ad121015a22a9420e05800137a28
Copying config sha256:4c926dc4eb0d455a9ef347131166c05995ef61c7849ab5ed7c333388047bbd2f
Writing manifest to image destination
Successfully pushed image-registry.openshift-image-registry.svc:5000/jegan/spring-ms@sha256:22aef16073f9e45ec5db513193a29b504172664e9560
6f5bbeb6e6d8780c29d1
Push successful
 jegan@tektutor.org  ~  
```



You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.


spring-ms-jegan.apps.ocp4.tektutor.org.labs - Google Chrome
Not secure spring-ms-jegan.apps.ocp4.tektutor.org.labs/
250%
Reset

Hello Microservice !

Info - Installing openssl (is already installed in our lab - just for your future reference)

Installing openssl from source code (Already installed on Lab machines, so kindly skip this installation)

```
sudo yum -y remove openssl openssl-devel
sudo yum groupinstall 'Development Tools'
sudo yum install perl-IPC-Cmd perl-Test-Simple -y
cd /usr/src
wget https://www.openssl.org/source/openssl-3.0.0.tar.gz
tar -zxf openssl-3.0.0.tar.gz
rm openssl-3.0.0.tar.gz
```

```
cd /usr/src/openssl-3.0.0
./config
make
make test
make install

sudo ln -s /usr/local/lib64/libssl.so.3 /usr/lib64/libssl.so.3
sudo ln -s /usr/local/lib64/libcrypto.so.3 /usr/lib64/libcrypto.so.3

sudo ldconfig
sudo tee /etc/profile.d/openssl.sh<<EOF
export PATH=/usr/local/bin:$PATH
export
LD_LIBRARY_PATH=/usr/local/openssl/lib:/usr/local/openssl/lib64:$LD_LIBRARY
_PATH
EOF

which openssl
openssl version
```

Lab - Create an edge route (https url)

You can secure your routes with https(secured) as url as opposed to http(unsecured).

Lab - Create an edge route (https based public route url)

Find your base domain of your openshift cluster

```
oc get ingresses.config/cluster -o jsonpath={.spec.domain}
```

Expected output

```
[root@tektutor.org auth]# oc get ingresses.config/cluster -o jsonpath=
{.spec.domain}
apps.ocp.tektutor.org.labs
```

Let's deploy a microservice and create an edge route as shown below.

First, let's generate a private key

```
openssl genrsa -out key.key
```

We need to create a public key using the private key with specific with your organization domain

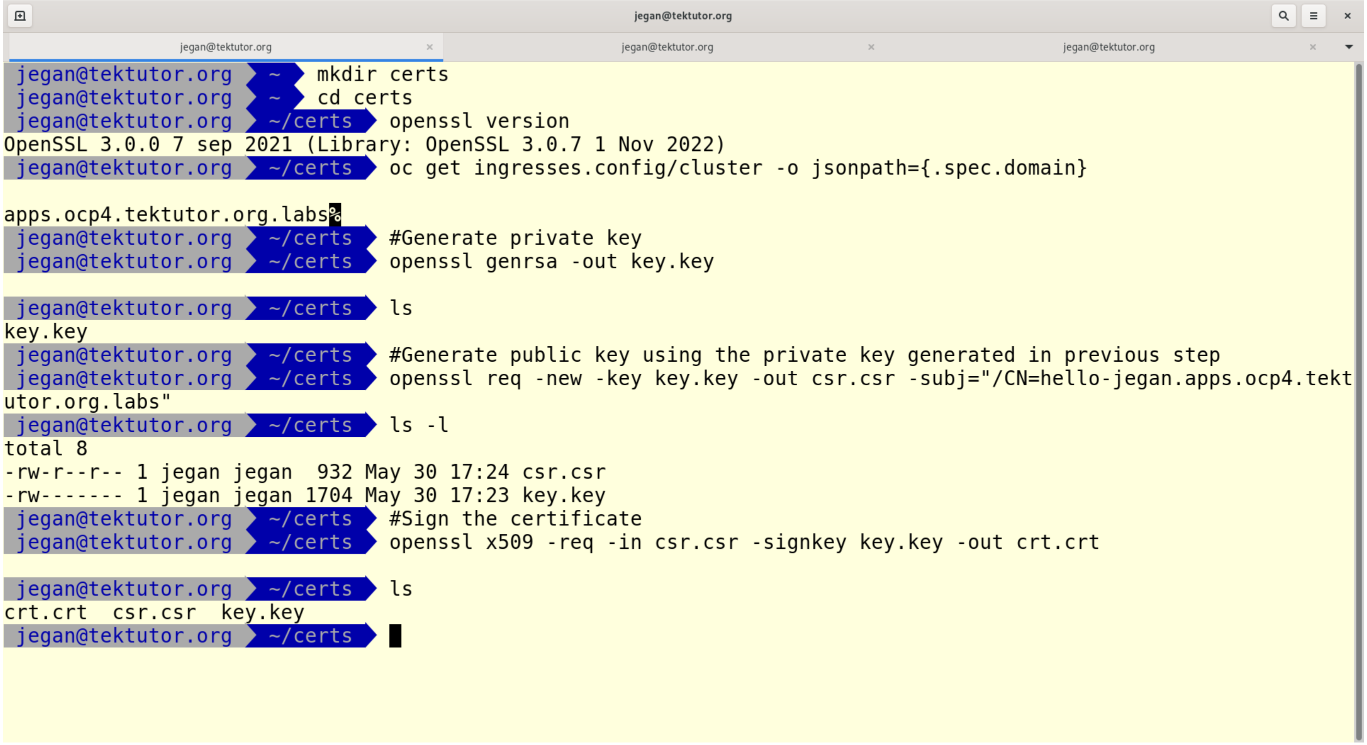
```
openssl req -new -key key.key -out csr.csr -subj="/CN=hello-
jegan.apps.ocp.tektutor.org.labs"
```

Sign the public key using the private key and generate certificate(.crt)

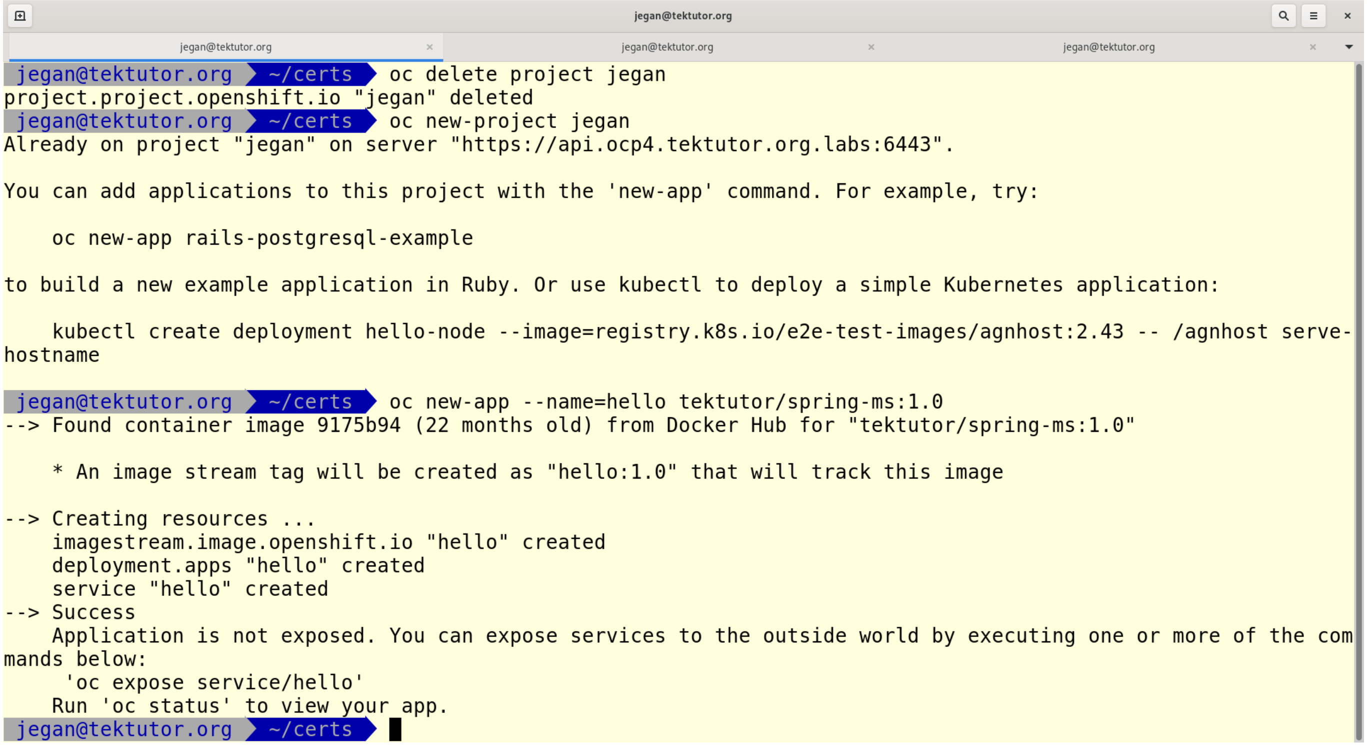
```
openssl x509 -req -in csr.csr -signkey key.key -out crt.crt
oc create route edge --service spring-ms --hostname hello-
jegan.apps.ocp4.tektutor.org.labs --key key.key --cert crt.crt
```

Expected output

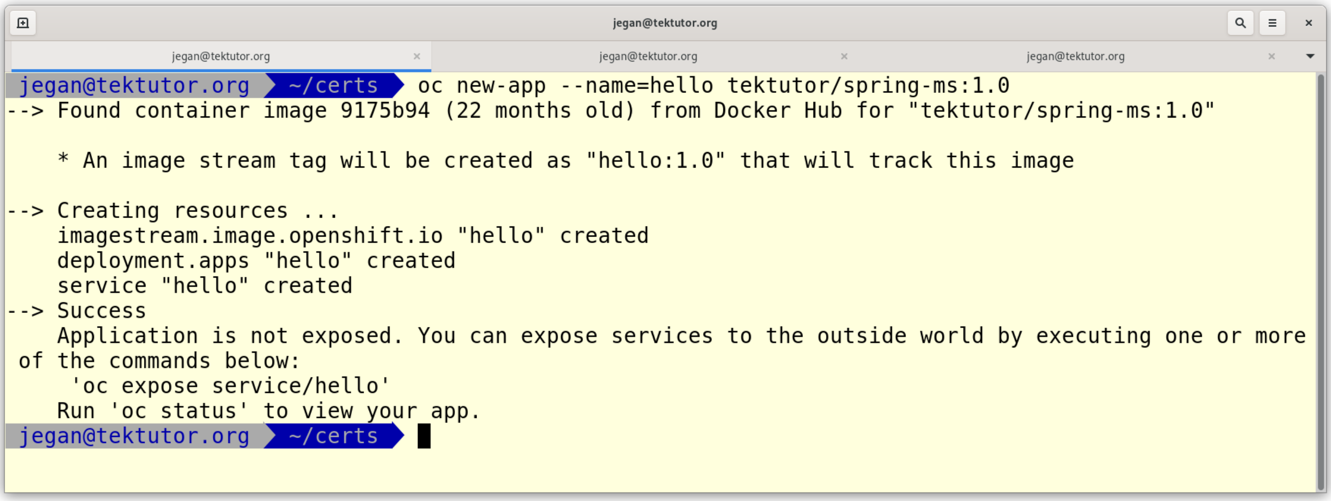
```
[jegan@tektutor.org edge-route]$ oc get svc
NAME        TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)    AGE
spring-ms   ClusterIP   172.30.208.33           8080/TCP   87m
[jegan@tektutor.org edge-route]$ oc expose deploy/nginx --port=8080
service/nginx exposed

[jegan@tektutor.org edge-route]$ oc get svc
NAME        TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)    AGE
nginx       ClusterIP   172.30.16.165           8080/TCP   4s
spring-ms   ClusterIP   172.30.208.33           8080/TCP   87m

[jegan@tektutor.org edge-route]$ oc get ingresses.config/cluster -o
jsonpath={.spec.domain}
apps.ocp4.tektutor.org.labs

[jegan@tektutor.org edge-route]$ oc project
Using project "jegan-devops" on server
"https://api.ocp4.tektutor.org.labs:6443".

[jegan@tektutor.org edge-route]$ openssl req -new -key key.key -out csr.csr
-subj="/CN=nginx-jegan-devops.apps.ocp4.tektutor.org.labs"

[jegan@tektutor.org edge-route]$ openssl x509 -req -in csr.csr -signkey
key.key -out crt.crt

[jegan@tektutor.org edge-route]$ oc create route edge --service nginx --
hostname nginx-jegan-devops.apps.ocp4.tektutor.org.labs --key key.key --
cert crt.crt
route.route.openshift.io/nginx created

[jegan@tektutor.org edge-route]$ oc get route
NAME    HOST/PORT                                        PATH   SERVICES
PORT    TERMINATION   WILDCARD
nginx   nginx-jegan-devops.apps.ocp4.tektutor.org.labs   nginx         edge
None
```



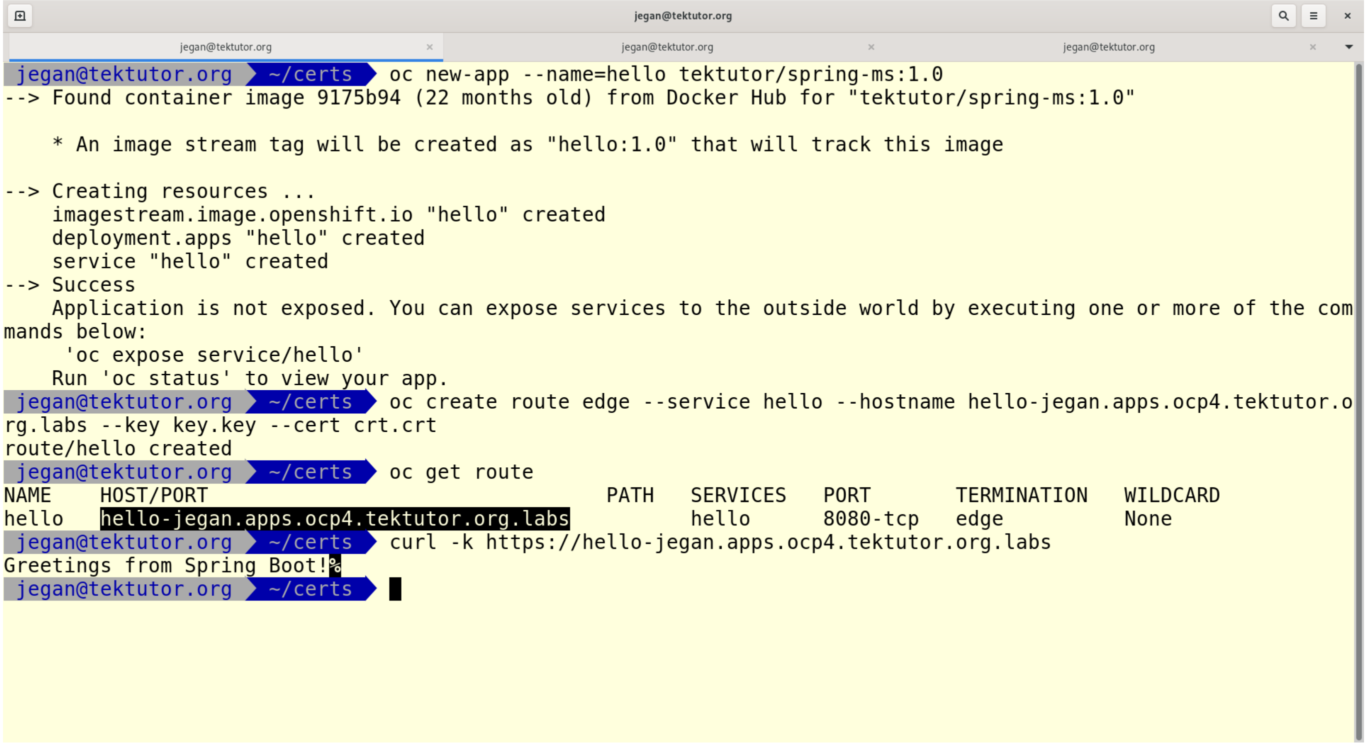
```
 jegan@tektutor.org  ~  mkdir certs
 jegan@tektutor.org  ~  cd certs
 jegan@tektutor.org  ~/certs  openssl version
OpenSSL 3.0.0 7 sep 2021 (Library: OpenSSL 3.0.7 1 Nov 2022)
 jegan@tektutor.org  ~/certs  oc get ingresses.config/cluster -o jsonpath={.spec.domain}

apps.ocp4.tektutor.org.labs%
 jegan@tektutor.org  ~/certs  #Generate private key
 jegan@tektutor.org  ~/certs  openssl genrsa -out key.key

 jegan@tektutor.org  ~/certs  ls
key.key
 jegan@tektutor.org  ~/certs  #Generate public key using the private key generated in previous step
 jegan@tektutor.org  ~/certs  openssl req -new -key key.key -out csr.csr -subj="/CN=hello-jegan.apps.ocp4.tekt
utor.org.labs"
 jegan@tektutor.org  ~/certs  ls -l
total 8
-rw-r--r-- 1 jegan jegan  932 May 30 17:24 csr.csr
-rw------- 1 jegan jegan 1704 May 30 17:23 key.key
 jegan@tektutor.org  ~/certs  #Sign the certificate
 jegan@tektutor.org  ~/certs  openssl x509 -req -in csr.csr -signkey key.key -out crt.crt

 jegan@tektutor.org  ~/certs  ls
crt.crt  csr.csr  key.key
 jegan@tektutor.org  ~/certs  
```



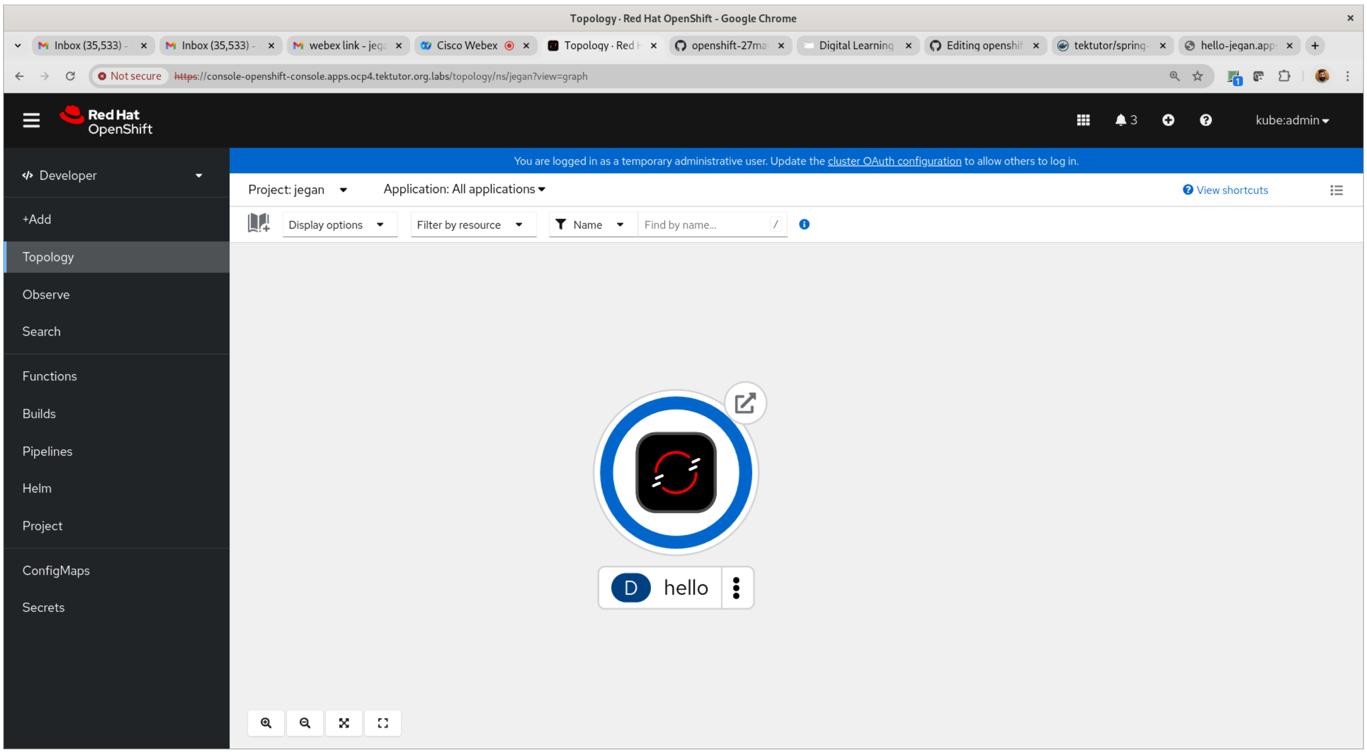
```
 jegan@tektutor.org  ~/certs  oc delete project jegan
project.project.openshift.io "jegan" deleted
 jegan@tektutor.org  ~/certs  oc new-project jegan
Already on project "jegan" on server "https://api.ocp4.tektutor.org.labs:6443".

You can add applications to this project with the 'new-app' command. For example, try:

    oc new-app rails-postgresql-example

to build a new example application in Ruby. Or use kubectl to deploy a simple Kubernetes application:

    kubectl create deployment hello-node --image=registry.k8s.io/e2e-test-images/agnhost:2.43 -- /agnhost serve-
hostname

 jegan@tektutor.org  ~/certs  oc new-app --name=hello tektutor/spring-ms:1.0
--> Found container image 9175b94 (22 months old) from Docker Hub for "tektutor/spring-ms:1.0"

    * An image stream tag will be created as "hello:1.0" that will track this image

--> Creating resources ...
    imagestream.image.openshift.io "hello" created
    deployment.apps "hello" created
    service "hello" created
--> Success
    Application is not exposed. You can expose services to the outside world by executing one or more of the com
mands below:
     'oc expose service/hello'
    Run 'oc status' to view your app.
 jegan@tektutor.org  ~/certs  
```



```
 jegan@tektutor.org  ~/certs  oc new-app --name=hello tektutor/spring-ms:1.0
--> Found container image 9175b94 (22 months old) from Docker Hub for "tektutor/spring-ms:1.0"

    * An image stream tag will be created as "hello:1.0" that will track this image

--> Creating resources ...
    imagestream.image.openshift.io "hello" created
    deployment.apps "hello" created
    service "hello" created
--> Success
    Application is not exposed. You can expose services to the outside world by executing one or more
 of the commands below:
     'oc expose service/hello'
    Run 'oc status' to view your app.
 jegan@tektutor.org  ~/certs  
```



```
 jegan@tektutor.org  ~/certs  oc new-app --name=hello tektutor/spring-ms:1.0
--> Found container image 9175b94 (22 months old) from Docker Hub for "tektutor/spring-ms:1.0"

    * An image stream tag will be created as "hello:1.0" that will track this image

--> Creating resources ...
    imagestream.image.openshift.io "hello" created
    deployment.apps "hello" created
    service "hello" created
--> Success
    Application is not exposed. You can expose services to the outside world by executing one or more of the com
mands below:
     'oc expose service/hello'
    Run 'oc status' to view your app.
 jegan@tektutor.org  ~/certs  oc create route edge --service hello --hostname hello-jegan.apps.ocp4.tektutor.o
rg.labs --key key.key --cert crt.crt
route/hello created
 jegan@tektutor.org  ~/certs  oc get route
NAME    HOST/PORT                               PATH   SERVICES   PORT       TERMINATION   WILDCARD
hello   hello-jegan.apps.ocp4.tektutor.org.labs        hello      8080-tcp   edge          None
 jegan@tektutor.org  ~/certs  curl -k https://hello-jegan.apps.ocp4.tektutor.org.labs
Greetings from Spring Boot!%
 jegan@tektutor.org  ~/certs 
```

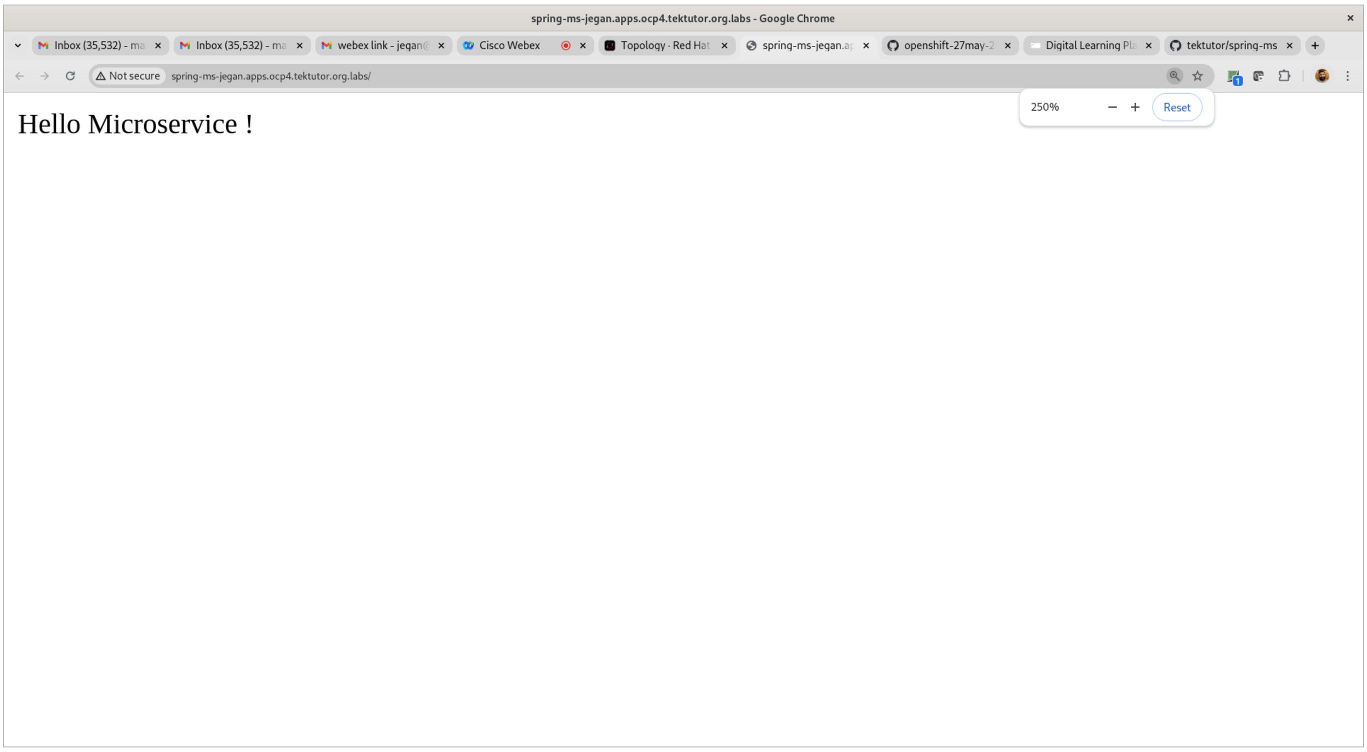


~~https~~://console-openshift-console.apps.ocp4.tektutor.org.labs/topology/ns/jegan?view=graph

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: jegan

Application: All applications


spring-ms-jegan.apps.ocp4.tektutor.org.labs - Google Chrome
Not secure
spring-ms-jegan.apps.ocp4.tektutor.org.labs/
250%
Reset

Hello Microservice !

Lab - Deploying multi-pod PHP application

Info - OpenShift Network Model

What is Flannel?

What is Calico?

What is Weave?

What is edge route?