neueda

futureproof your workforce

MONITORING AND MANAGING APPLICATIONS

NEUEDA.COM

Objectives

- Command line tools for listing and debugging resources
- Monitoring projects, deployments and the cluster
- Deleting resources with the command line

Monitoring and Managing Applications

- We should now have a good handle on the workings of an OpenShift cluster when everything goes as planned …
- … but things often don’t go as planned.
- OpenShift provides a range of tools for inspecting, debugging and managing resources on your cluster.

Command Line Tools

The action you want to perform
The kind of object you want to do it to
The name of specific object (if any)

- For many monitoring and debugging purposes the command line is the best port of call.
- In particular the oc command line interface is the simplest tool.
- General pattern for addressing a resource:

```
$ oc <verb> <kind> <name>
```

- For example:

```
$ oc get pods
NAME                      READY   STATUS      RESTARTS   AGE
deploy-example-1-bbc2l    1/1     Running     0          93m
deploy-example-1-deploy   0/1     Completed   0          93m
deploy-example-1-q47t6    1/1     Running     0          93m

$ oc get pod deploy-example-1-bbc2l
NAME                      READY   STATUS    RESTARTS   AGE
deploy-example-1-bbc2l    1/1     Running   0          104m
```

Listing and Detailing Resources

- In the previous slide `$ oc get pods` returns a list of all the resources of type pod.
- In our list all the running pods are suffixed with a unique ID to distinguish among replicas.
- You can also use the `--selector` option to filter your list on labels

```
$ oc get pods --selector app=spring-petclinic
NAME                                  READY   STATUS    RESTARTS      AGE
spring-petclinic-7f9796765b-pv2k4     1/1     Running   7 (33m ago)   126m
```

Describing Resources

- The describe verb gives a lot more detail on a resource

```
$ oc describe deployment spring-petclinic
Name:                   spring-petclinic
Namespace:              adm-petclinic
CreationTimestamp:      Wed, 27 Apr 2022 13:25:32 +0100
Labels:                 app=spring-petclinic
Selector:               app=spring-petclinic
Replicas:               1 desired | 1 updated | 1 total | 1 available | 0
unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  50% max unavailable, 50% max surge
Pod Template:
  Labels:  app=spring-petclinic
  Containers:
   spring-petclinic:
>-- SNIP --<
```

Viewing Logs

- When a problem arises, the logs subcommand will retrieve the logs of a specified pod for troubleshooting:

```
$ oc logs spring-petclinic-7f9796765b-pv2k4
Starting the Java application using /opt/jboss/container/java/run/run-java.sh ...
INFO exec  java -Xms25m -Xmx100m -XX:+UseParallelGC -XX:MinHeapFreeRatio=10 -XX:MaxHeapFreeRatio=20 -XX:GCTimeRatio=4 -
XX:AdaptiveSizePolicyWeight=90 -XX:+ExitOnOutOfMemoryError -cp "." -jar /deployments/spring-petclinic-2.3.0.BUILD-SNAPSHOT.jar
              |\      _,,,--,,_
             /,`.-'`'   ._  \-;;,_
  _______ __|,4-  ) )_   .;.(__`'-'__     ___ __    _ ___ _______
 |       | '---''(_/._)-'(_\_)   |   |   |   |  |  | |   |       |
 |    _  |    ___|_     _|       |   |   |   |   |_| |   |       | __ _ _
 |   |_| |   |___  |   | |       |   |   |   |       |   |       | \ \ \ \
 |    ___|    ___| |   | |      _|   |___|   |  _    |   |      _|  \ \ \ \
 |   |   |   |___  |   | |     |_|       |   | | |   |   |     |_     ) ) ) )
 |___|   |_______| |___| |_______|_______|___|_|  |__|___|_______|  / / / /
 ==================================================================/_/_/_/

:: Built with Spring Boot :: 2.3.3.RELEASE

2022-05-05 15:50:11.632  INFO 1 --- [
>-- SNIP --<
```

Debugging Options

- If the problem isn’t in configuration or deployment, you need to debug at the application level.
- The `oc` CLI has a set of commands for running things inside your application’s container

oc Commands for Container Access

`oc rsh`

Sets up a connection to an interactive shell in a Pod

- By default rsh picks the first container in the specified Pod.
- You can specify a different container (which must include an interactive shell)

`oc exec`

Runs a given command inside the given container

- You can `exec` when you can't `rsh`. The `exec` subcommand can directly invoke an executable without needing a shell.

`oc debug`

Starts a new instance running a command shell

- Connects you to a terminal running inside a specified container (like `rsh`)
- But `debug` starts a new instance instead of the entry point specified in the container image.
- Suppose a container is failing to start with its usual server command, run `debug` on the failing container's deployment or pod to run a new instance and bypass the failing server in favour of a shell.

```
$ oc new-app httpd-example                          # Deploy from registry
--> Deploying template "openshift/httpd-example" to project adm-petclinic
>--SNIP--<
$ oc get pods --selector name=httpd-example         # Get the Pod name
NAME                    READY   STATUS    RESTARTS   AGE
httpd-example-1-ztb8s   1/1     Running   0          3m43s
$ oc rsh httpd-example-1-ztb8s                      # rsh into the first container in the Pod
sh-4.4$ ps ax                                       # This is inside the container shell now
    PID TTY      STAT   TIME COMMAND
      1 ?        Ss     0:00 httpd -D FOREGROUND
     33 ?        S      0:00 /usr/bin/coreutils --coreutils-prog-shebang=cat /usr/bin/cat
     34 ?        S      0:00 /usr/bin/coreutils --coreutils-prog-shebang=cat /usr/bin/cat
     35 ?        S      0:00 /usr/bin/coreutils --coreutils-prog-shebang=cat /usr/bin/cat
     36 ?        S      0:00 /usr/bin/coreutils --coreutils-prog-shebang=cat /usr/bin/cat
     37 ?        S      0:00 httpd -D FOREGROUND
     38 ?        Sl     0:00 httpd -D FOREGROUND
     40 ?        Sl     0:00 httpd -D FOREGROUND
     42 ?        Sl     0:00 httpd -D FOREGROUND
    252 pts/0    Ss     0:00 /bin/sh
    257 pts/0    R+     0:00 ps ax
sh-4.4$ env
HTTPD_CONTAINER_SCRIPTS_PATH=/usr/share/container-scripts/httpd/
>--SNIP--<
SUMMARY=Platform for running Apache httpd 2.4 or building httpd-based application
```

oc rsh example

```
$ oc exec httpd-example-1-ztb8s -- ps -ax
    PID TTY      STAT   TIME COMMAND
      1 ?        Ss     0:00 httpd -D FOREGROUND
     33 ?        S      0:00 /usr/bin/coreutils --coreutils-prog-shebang=cat /usr/bin/cat
     34 ?        S      0:00 /usr/bin/coreutils --coreutils-prog-shebang=cat /usr/bin/cat
     35 ?        S      0:00 /usr/bin/coreutils --coreutils-prog-shebang=cat /usr/bin/cat
     36 ?        S      0:00 /usr/bin/coreutils --coreutils-prog-shebang=cat /usr/bin/cat
     37 ?        S      0:00 httpd -D FOREGROUND
     38 ?        Sl     0:00 httpd -D FOREGROUND
     40 ?        Sl     0:00 httpd -D FOREGROUND
     42 ?        Sl     0:00 httpd -D FOREGROUND
    264 ?        Rs     0:00 ps -ax
```

oc exec example

```
$ oc debug httpd-example-1-ztb8s
Starting pod/httpd-example-1-ztb8s-debug, command was: container-entrypoint /bin/sh -c
/usr/libexec/s2i/run
Pod IP: 10.129.2.26
If you don't see a command prompt, try pressing enter.
sh-4.4$ ps -ax                                    # This is inside the container debug shell
    PID TTY      STAT   TIME COMMAND
      1 pts/0    Ss     0:00 /bin/sh
      6 pts/0    R+     0:00 ps -ax
sh-4.4$ exit
exit

Removing debug pod ...
```

oc debug example

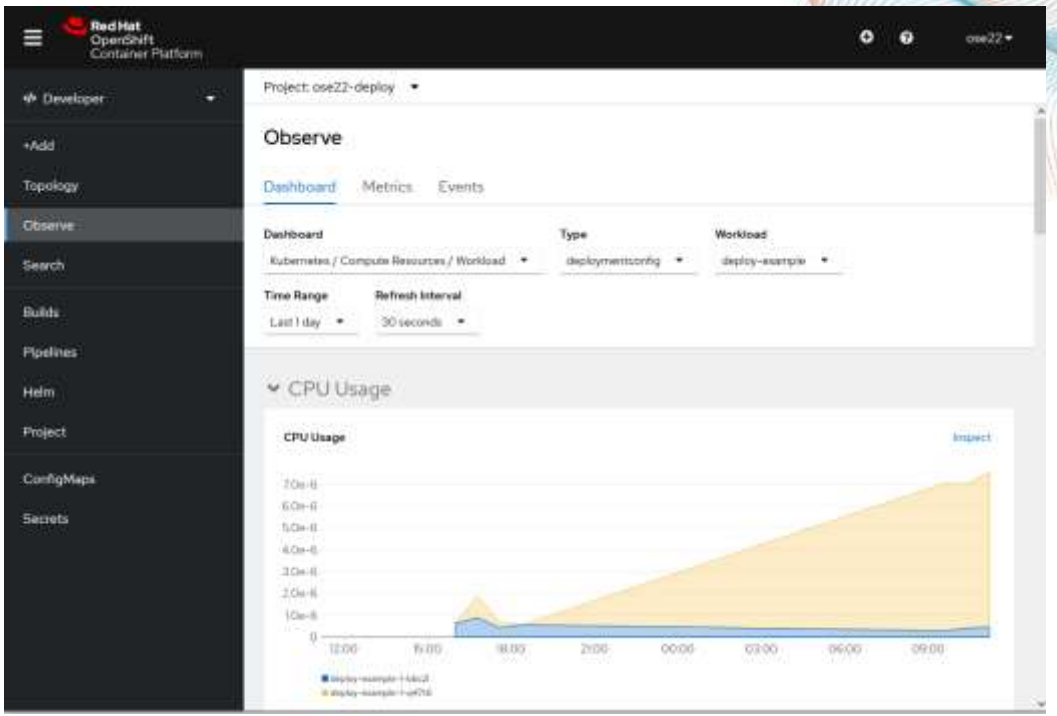
OPENSHIFT MONITORING

OpenShift Monitoring

- OpenShift leverages the Prometheus open source project for its monitoring solution.
  - Cluster resources: CPU and memory on Nodes
  - Control plane pods
  - Platform Services
  - Includes alerts to notify admins of unusual usage

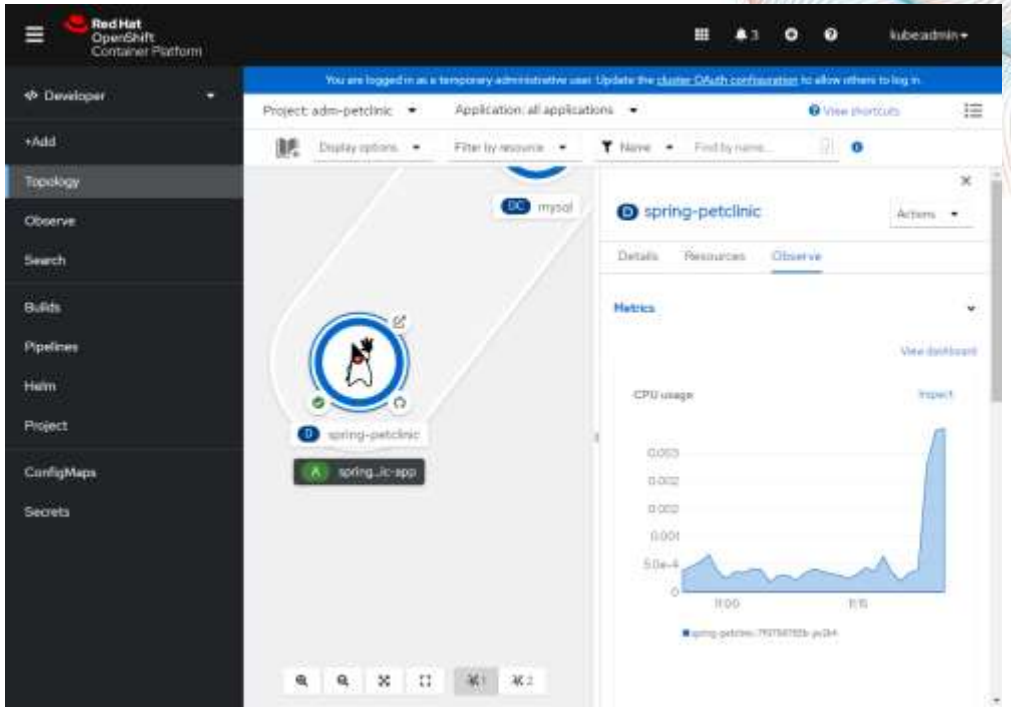
Monitoring in the Developer Perspective

- In the Developer Perspective select the “Observe” option from the left sidebar.



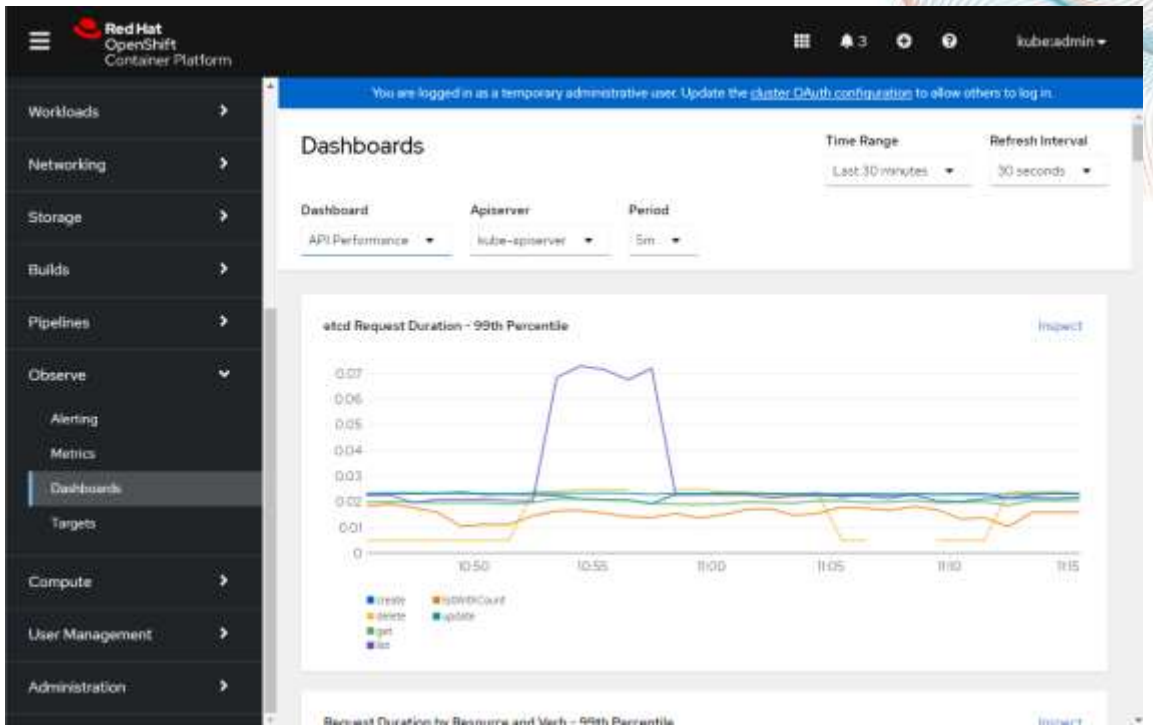
Monitoring a Deployment

- You can also monitor a single deployment by clicking the “Observe” tab in the right sidebar.
- This is only available for Deployments, not DeploymentConfigs.



Monitoring in the Administrator Perspective

- Only on administrator accounts
- Gives dashboards, metrics, alerting and targets for workloads across the cluster.



Deleting Resources with oc

- In the command line, resource deletion is achieved with the `oc delete` command:

```
$ oc delete project depl-proj
project.project.openshift.io "depl-proj" deleted
```

- Sometimes you need to delete selectively - the `all` identifier can be used to access a list of resources with a given label:

```
$ oc delete all --selector app=nodejs-frontend
pod "nodejs-frontend-78f69c6d68-mhv4d" deleted
service "nodejs-frontend" deleted
deployment.apps "nodejs-frontend" deleted
buildconfig.build.openshift.io "nodejs-frontend" deleted
imagestream.image.openshift.io "nodejs-frontend" deleted
route.route.openshift.io "nodejs-frontend" deleted
```

Summary

- Command line tools for listing and debugging resources
- Monitoring projects, deployments and the cluster
- Deleting resources in the command line

Questions and Comments?