neueda

futureproof your workforce

PRODUCTION DEPLOYMENT AND SCALING

NEUEDA.COM

Objectives

- Application Scaling
  - Manual Scaling
  - Horizontal Pod Autoscaler
- Health Checks: Liveness, Readiness, Startup
- Deployment Strategies

neueda
futureproof your workforce

Production Deployment and Scaling

- Deploying any given app in a production setting requires the ability to scale up to demand (and scale down to budget).
- The app should also be updateable without compromising your user experience (zero downtime).
- This functionality is often a big part of why the app is deployed on a cluster in the first place.
- OpenShift has built in capability for these requirements, along with a robust health checking mechanism.

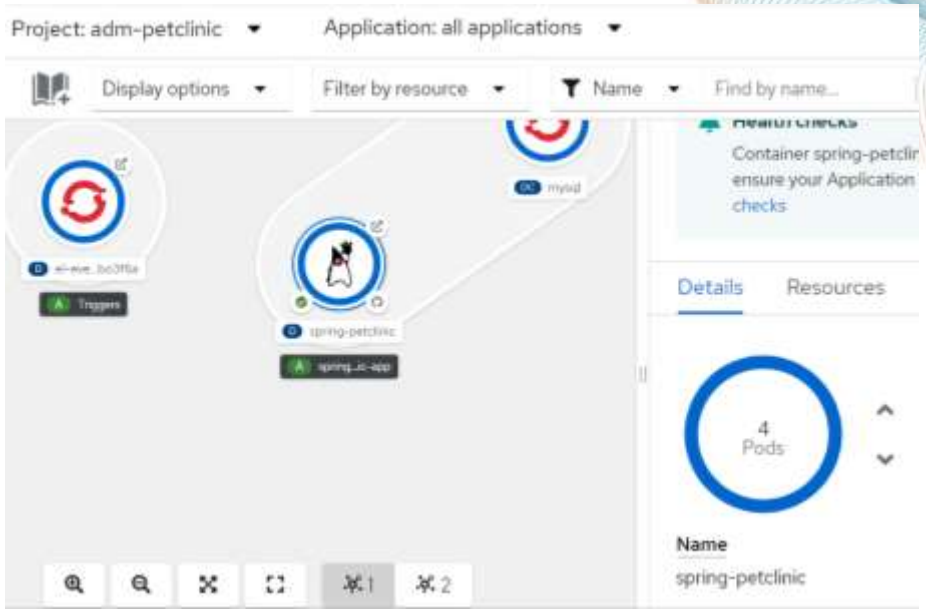
neueda
future-proof your workforce

APPLICATION SCALING

neueda
futureproof your workforce

Application Scaling

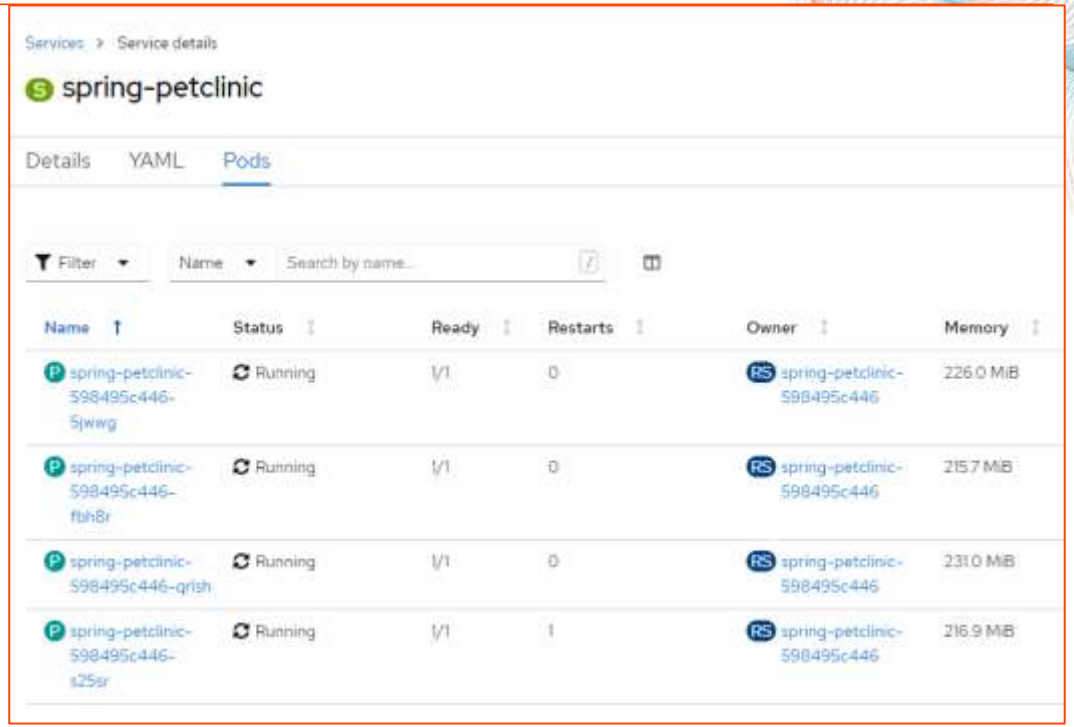
- We have already seen how we can manually scale up an app using the Web Console.
- Here we have scaled the app up to 4 Pods.



neueda
futureproof your workforce

Load Balancing

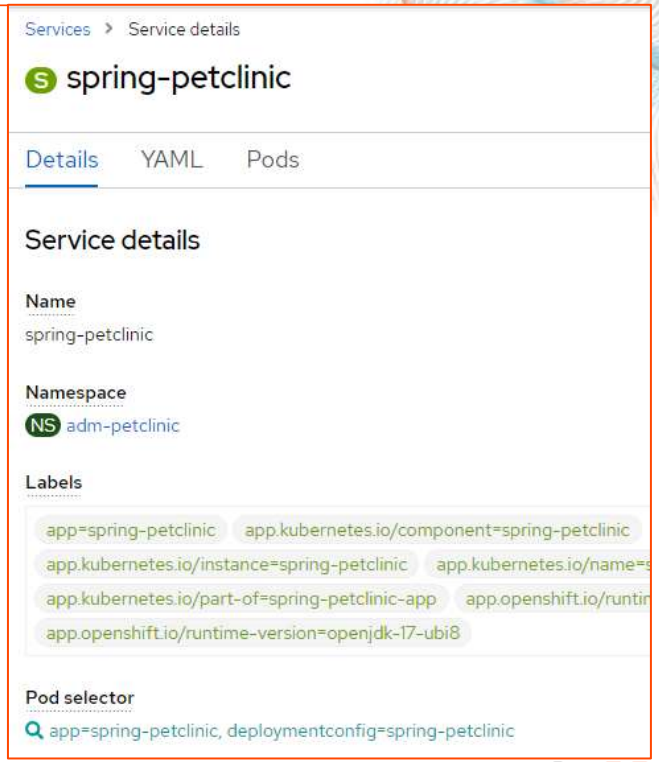
- The Service created to handle incoming requests and distribute them to the Pods also balances the load.
- How does it know which Pods to balance?



neueda
futureproof your workforce

Which Pods to Balance?

- Notice the Service has a section called “Pod selector”. This is defined in the `spec.selector.app` section of the YAML file.
- It looks for Pods that have the same set of tags to balance.



neueda
future proof your workforce

Automatic Scaling

- While manual scaling can be useful, we would generally prefer for OpenShift to react automatically to usage by scaling up and down as needed.
- OpenShift has a built in Horizontal Pod Autoscaler (HPA) that will automatically scale your deployment based on a CPU and memory threshold you specify.

neueda
futureproof your workforce

Demo: Adding HPA to the Pet Clinic

- To configure an HPA the deployment needs to specify memory and CPU limits.
- This is done in the initial deployment configuration
  - Click on the app, then from the **Actions** menu choose `Edit spring-petclinic`
  - Down the bottom of the page, choose the **Resource Limit** link.
  - Set the CPU request to 100 millicores, and Limit to 1 core.
  - Set the memory request to 100 Mi and limit to 200 Mi.

neueda
future proof your workforce

Demo: Adding HPA to the Pet Clinic

- Back in Topology click on the Actions menu again, and select AddHorizontalPodAutoscaler
- Set the Minimum Pods to 1 and the Maximum to 3
- Set the CPU Utilization to 5% and the Memory to 5%
- Save the HPA and note that now the app details says your app has been autoscaled!
- The OpenShift OperatorHub also includes a **VerticalPodAutoscaler** Operator, which allows you to update resource limits based on usage values it learns.

neueda
future proof your workforce

HEALTH CHECKS

neueda
futureproof your workforce

Health Checks

- OpenShift has health-checking functionality
- This automatically polls an application using HTTP, TCP or container commands to determine if it is healthy.
- Based on the poll's response OpenShift can attempt to redeploy and/or notify administrators.
- All health checks require at least some knowledge of how your app is implemented.

neueda
future proof your workforce

Liveness Probes

- A **liveness probe** checks the application pod is running and healthy.
- It can do this in three ways
  - HTTP(S) checks - Checks a given URL endpoint served by the container and checks the return code.
  - Container evaluation check - a command (usually a script) run at intervals to check the container is behaving normally.
  - TCP Socket checks - Checks a TCP connection is possible on a specific TCP port in the application pod.

neueda
future proof your workforce

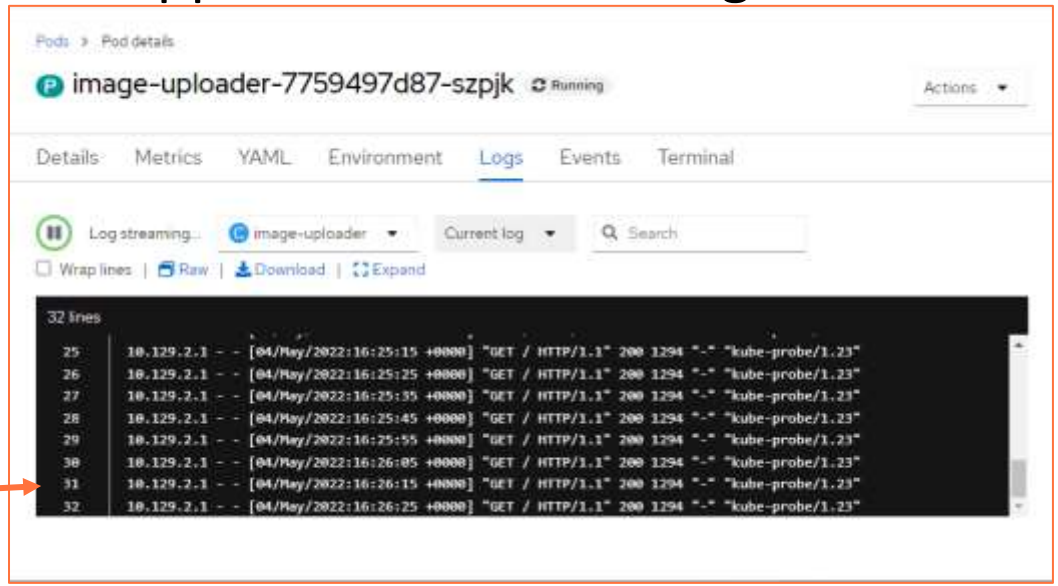
Adding a Liveness Probe to the Image Uploader

- In the Image Uploader project click on the app and choose “Add Health Checks” from the Actions menu.
- Scroll to **Liveness probe** and click on Add liveness probe. Fill out the following:
  - Type: HTTP GET
  - Use HTTPS: unchecked
  - Path: /
  - Port: 8080
  - Initial Delay: 5
  - Timeout: 1
- Check the tick at the bottom and click the “Add” button.

neueda
future proof your workforce

Checking the Liveness Probe

- The easiest way to check the probe is working is to open the Pod details of the app and click on the Logs tab:



Liveness probes

neueda
future proof your workforce

Readiness Probes

- Many applications need to perform a set of initialization tasks before they are able to receive traffic.
- In OpenShift a readiness probe is used to determine if a Pod is ready for traffic.
- It is very similar to a liveness probe, except failing it doesn’t result in a new pod - the pod remains running while not receiving traffic.
- The functionality is wired into the apps, here we will demo a *failed* readiness probe.

neueda
future proof your workforce

Adding a Readiness Probe to the Image Uploader

- Follow the same process as before, selecting Edit Health Checks this time.
- Click on “Add readiness probe”
  - Type: HTTP GET
  - Use HTTPS: unchecked
  - Path: `/notreal`
  - Port: `8080`
  - Initial Delay: 5
  - Timeout: 1
- The `/notreal` endpoint doesn’t exist, so the Pod will fail the probe.

neueda
future-proof your workforce

Checking the Readiness Probe

- Click on the app in the Topology view
- Under the “Observe” tab click on “All events” at the bottom, or click on the Pod in resources and go to Events.
- You can see that readiness probes are failing

neueda
future proof your workforce

Readiness Probe Functionality

- Notice that the application is still working as normal, it’s just the new deployment is not scaling up.
- OpenShift automatically protects your deployment by keeping the previous (working) version up and running while the readiness probe is failing.
- After 10 minutes, the readiness probe should trigger a failed deployment at which point the pod will be deleted and the deployment will roll back to the previous working configuration.

neueda
future-proof your workforce

Startup Probes

- OpenShift offers a third kind of health check - a **Startup Probe**
- This indicates whether the application within a container has started (all other probes are disabled until the startup probe completes).
- If the probe fails to register within the specified time, the Pod is killed and restarted according to the `restartPolicy`.
- Particularly useful if you have an application with a long startup time so you wish to delay liveness and readiness checks.

neueda
futureproof your workforce

DEPLOYMENT STRATEGIES

neueda
futureproof your workforce

OpenShift Deployment Strategies

- Over time deployments need update, failure recovery, scale changes etc
- In a multi-node application this requires a rollout strategy.
- OpenShift has 3 strategies available:
  - Rolling deployment
  - Recreate deployment
  - Custom deployment.

neueda
futureproof your workforce

Rolling and Canary Deployments

- A **rolling deployment** is the default and most common strategy
  - Instances of a previous version are slowly replaced with instances of a new version.
  - If health checking is configured, the deployment waits for a readiness check on new pods before scaling down old one.
  - This is a zero-downtime strategy
- OpenShift rolling deployments are also **canary deployments**
  - Failure of the new (canary) deployment will trigger a rollback.

neueda
futureproof your workforce

Recreate Deployments

- This is used when a new version cannot run alongside the old one (why may that be?)
  - The existing deployment is completely scaled down to zero, then the new deployment is scaled up.
  - Obviously this incurs downtime.

neueda
future-proof your workforce

Custom Deployments

- OpenShift allows you considerable freedom to create custom deployment strategies depending on your application needs
  - See the documentation for more info on this.

neueda
futureproof your workforce

Configuring a Deployment Strategy

- In the Topology view of the Developer perspective, click on the application node and select the Details tab.
- Note that the Update strategy is automatically set to a RollingUpdate.
- You can change this by selecting the Actions menu and selecting “Edit Deployment” or “Edit DeploymentConfig” depending on your deployment type.

neueda
future proof your workforce

About Deployments and DeploymentConfigs

- OpenShift has two main paths to deploying applications.

1. **Deployment**: This is a k8s native resource that is designed to maximise the availability of the application, but might not give optimal consistency.
2. **DeploymentConfig**: This is an additional OpenShift resource that gives finer grained control over your deployment and rollout strategy.

neueda
futureproof your workforce

Deployments and DeploymentConfigs

- For the most part you probably won’t care which one you are using.
- But they do change the options available to you (in the same way as a BuildConfig and a Pipeline change the build path).
- Deployments initiate rollouts by default whenever a suitable change is detected, DeploymentConfigs need to be triggered manually by default.

neueda
future proof your workforce

Demo: Deployment Strategies

neueda
futureproof your workforce

Summary

- Application Scaling
  - Manual Scaling
  - Horizontal Pod Autoscaler
- Health Checks: Liveness, Readiness, Startup
- Deployment Strategies

neueda
futureproof your workforce

Questions and Comments?

neueda
futureproof your workforce