How can you ensure there are at least 3 Pod instances (example) are always available and running at point in time?

srinathchalla@outlook.com

ReplicaSet

Concept

Udemy

Objectives

Concept

a. Overview - ReplicaSet
b. Labels & Selectors
c. Equality-based and Set-based selectors

Review Demo

a. Manifest file – ReplicaSet
b. Deploy application with ReplicaSet
c. Display and validate
d. Test Cases
e. Clean up

srinathchalla@outlook.com

ReplicaSet

- Ensures that a specified number of pods are running at any time
    a. If there are excess Pods, they get killed and vice versa
    b. New Pods are launched when they get fail, get deleted or terminated

- ReplicaSet and Pods are associated with “ labels ”

srinathchalla@outlook.com

ReplicaSet vs. Replication Controller

ReplicaSet is Next-generation Replication Controller

ReplicaSet

↓

Set-based Selectors

Replication Controller

↓

Equality-based Selectors

srinathchalla@outlook.com

Labels & Selectors

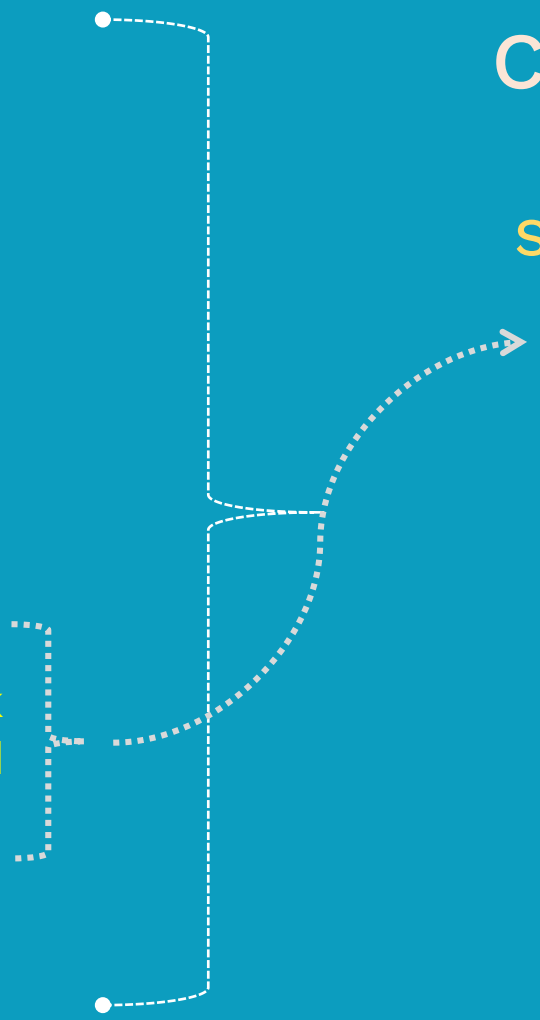
Pods

Labels

```
#Pod-Spec
apiVersion: v1
kind: pod
metadata:
  name: nginx-pod
  labels:
    app: guestbook
    tier: frontend
    env: dev
spec:
  replicas: 5
  . . . . .
```

Controllers & Services

Selectors



srinathchalla@outlook.com

Equality-based

Operators:

= == !=

Examples:

environment = production
tier != frontend

Command line

```
$ kubectl get pods -l environment=production
```

In manifest:

```
...
selector:
  environment: production
  tier: frontend
...
```

⚠ Supports: Services, Replication Controller

Set-based

Operators:

in notin exists

Examples:

environment in (production, qa)
tier notin (frontend, backend)

Command line

```
$ kubectl get pods -l 'environment in (production)
```

In manifest:

```
...
selector:
  matchExpressions:
    - {key: environment, operator: In, values: [prod, qa]}
    - {key: tier, operator: NotIn, values: [frontend, backend]}
. . .
```

⚠ Supports: Job, Deployment, Replica Set, and Daemon Set,

srinathchalla@outlook.com

```
...
selector:
    app: nginx
    tier: frontend
...
```

=

```
...
selector:
  matchLabels:
    app: nginx
     tier: frontend
...
```

Supports on Older Resourses such as:

- ReplicationControllers,
- Services

Supports on newer resourses such as:

- ReplicaSets
- Deployments
- Jobs
- DaemonSet

srinathchalla@outlook.com

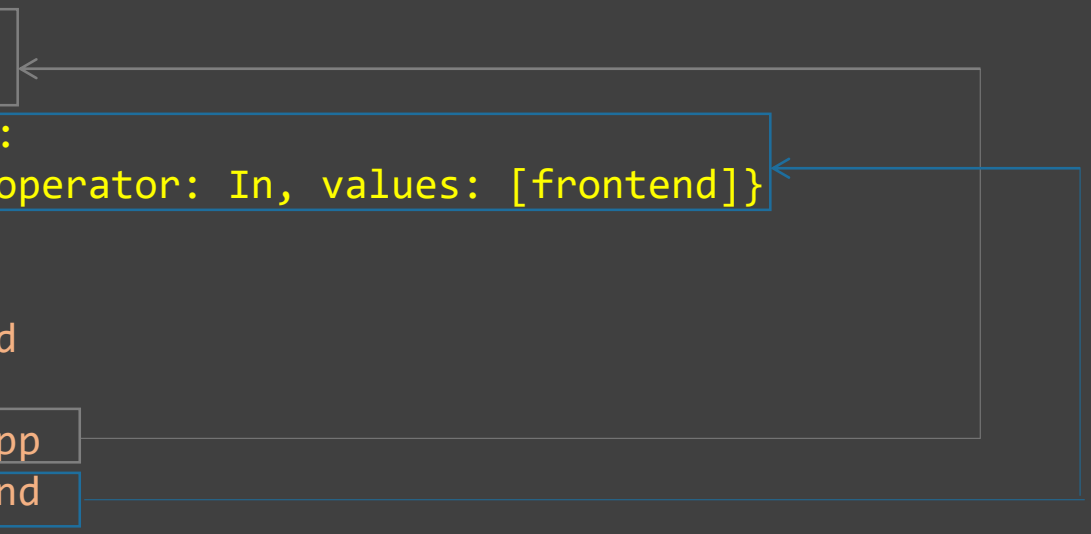
Review
Demo

a. Manifest file
b. Deploy app using RS
c. Display and validate RS
d. Test – Node fails
e. Test – Scale up
f. Test – Scale down

srinathchalla@outlook.com

ReplicaSet– Manifest file

```yaml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: nginx-rs
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx-app
    matchExpressions:
      - {key: tier, operator: In, values: [frontend]}
  template:
    metadata:
      name: nginx-pod
      labels:
        app: nginx-app
        tier: frontend
    spec:
      containers:
      - name: nginx-container
        image: nginx
        ports:
        - containerPort: 80
```



srinathchalla@outlook.com

ReplicaSet– Create & Display

```
[srinath@master ~]$ kubectl create -f nginx-rs.yaml
replicaset/nginx-rs created
```

```
[srinath@master ~]$ kubectl get pods
NAME             READY     STATUS    RESTARTS   AGE
nginx-rs-62vpv   1/1       Running   0          6m
nginx-rs-fk67w   1/1       Running   0          6m
nginx-rs-qk4ph   1/1       Running   0          6m
```

```
[srinath@master ~]$ kubectl get po -l tier=frontend
NAME             READY     STATUS    RESTARTS   AGE
nginx-rs-62vpv   1/1       Running   0          4m
nginx-rs-fk67w   1/1       Running   0          4m
nginx-rs-qk4ph   1/1       Running   0          4m
```

srinathchalla@outlook.com

ReplicaSet– Describe

```
srinath@master:$ kubectl get rs nginx-rs –o wide
NAME       DESIRED   CURRENT   READY     AGE       CONTAINERS          IMAGES    SELECTOR
nginx-rs   3         3         3         14s       nginx-container     nginx     app=nginx-app,tier in (frontend)
```

```
srinath@master:$ kubectl describe rs nginx-rs
Name:         nginx-rs
Namespace:    default
Selector:     app=nginx-app
Labels:       app=nginx-app
Annotations:  <none>
Replicas:     3 current / 3 desired
Pods Status:  3 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=nginx-app
  Containers:
   nginx-container:
    Image:        nginx
    Port:         80/TCP
    Environment:  <none>
    Mounts:       <none>
  Volumes:        <none>
Events:
  Type    Reason            Age   From                   Message
  ----    ------            ----  ----                   -------
  Normal  SuccessfulCreate  8m    replicaset-controller  Created pod: nginx-rs-8vcfq
  Normal  SuccessfulCreate  8m    replicaset-controller  Created pod: nginx-rs-l2t5z
  Normal  SuccessfulCreate  8m    replicaset-controller  Created pod: nginx-rs-br776
```

srinathchalla@outlook.com

ReplicaSet – Scheduling

```
[srinath@master ~]$ kubectl get po -o wide
NAME              READY     STATUS    RESTARTS   AGE       IP              NODE      NOMINATED NODE
nginx-rs-5lb6b    1/1       Running   0          16s       10.240.1.32     node1     <none>
nginx-rs-l64js    1/1       Running   0          16s       10.240.1.33     node1     <none>
nginx-rs-qrqv7    1/1       Running   0          16s       10.240.2.28     node2     <none>
```

```
[srinath@master ~]$ kubectl get nodes
NAME      STATUS     ROLES     AGE       VERSION
master    Ready      master    1d        v1.11.2
node1     Ready      <none>    1d        v1.11.2
node2     NotReady   <none>    1d        v1.11.2
```

```
[srinath@master ~]$ kubectl get po -o wide
NAME              READY     STATUS    RESTARTS   AGE       IP              NODE      NOMINATED NODE
nginx-rs-5lb6b    1/1       Running   0          11m       10.240.1.32     node1     <none>
nginx-rs-l64js    1/1       Running   0          11m       10.240.1.33     node1     <none>
nginx-rs-mtwzs    1/1       Running   0          2m        10.240.1.34     node1     <none>
nginx-rs-qrqv7    1/1       Unknown   0          11m       10.240.2.28     node2     <none>
```

srinathchalla@outlook.com

ReplicaSet – Scaling up

```
[srinath@master ~]$ kubectl scale rs nginx-rs --replicas=5
replicaset/nginx-rs scaled
```

```
[srinath@master ~]$ kubectl get rs nginx-rs
NAME       DESIRED   CURRENT   READY     AGE
nginx-rs   5         5         5         9m
```

```
[srinath@master ~]$ kubectl get po -o wide
NAME             READY     STATUS    RESTARTS   AGE       IP              NODE  NOMINATED NODE
nginx-rs-2x2kg   1/1       Running   0          36m       10.240.1.173    node1 <none>
nginx-rs-7kvhl   1/1       Running   0          36m       10.240.2.3      node2 <none>
nginx-rs-g7mwq   1/1       Running   0          33m       10.240.2.42     node2 <none>
nginx-rs-jgt28   1/1       Running   0          33m       10.240.1.3      node1 <none>
nginx-rs-wvmrx   1/1       Running   0          36m       10.240.2.4      node2 <none>
```

srinathchalla@outlook.com

ReplicaSet – Scaling down

```
[srinath@master ~]$ kubectl scale rs nginx-rs --replicas=3
replicaset/nginx-rs scaled
```

```
[srinath@master ~]$ kubectl get rs nginx-rs
NAME       DESIRED   CURRENT   READY     AGE
nginx-rs   3         3         3         45m
```

```
[srinath@master ~]$ kubectl get po -o wide
NAME             READY     STATUS    RESTARTS   AGE       IP              NODE  NOMINATED NODE
nginx-rs-2x2kg   1/1       Running   0          36m       10.240.1.173    node1 <none>
nginx-rs-jgt28   1/1       Running   0          33m       10.240.1.3      node1 <none>
nginx-rs-wvmrx   1/1       Running   0          36m       10.240.2.4      node2 <none>
```

srinathchalla@outlook.com

ReplicaSet – Delete

```
[srinath@master ~]$ kubectl delete -f nginx-rs.yaml  #kubectl delete rs nginx-rs
replicaset "nginx-rs" deleted
```

```
[srinath@master ~]$ kubectl get po -l app=nginx-app
No resourses found.
```

srinathchalla@outlook.com

Summary

Concept

a. Ensures that a specified number of pods are running at any time
b. Next gen Replication Controller
c. Equality based selector vs. Set-based selector

Review Demo

a. Manifest file
b. Deploy application with Replication Controller
c. Display and validate
d. Scheduling, Scale up and Scale down the application
e. Clean up

Coming up...

Demo

ReplicaSet

srinathchalla@outlook.com