

Pod

Pod , , Docker .

Pod cgroup , IF

pod .

Pod

```
$ kubectl get pods
```

NAME	READY	STATUS	RESTARTS	AGE
kubernetes-bootcamp-5c69669756-jbgqh	1/1	Running	0	1m

Pod

```
# kubectl run kuard --image=gcr.io/kuar-demo/kuard-amd64:1
# kubectl apply -f kuard-pod.yaml
# kubectl delete deployments/kuard
# kubectl get pods
```

Pod

 $(IP, \quad) .$

```
# kubectl describe pods {{pod name}}
```

```
//
Name:          kuard
Namespace:     default
Node:          <node name>
Start Time:    Mon, 16 Jul 2018 10:45:07 +0900
Labels:        <none>
Annotations:   kubectl.kubernetes.io/last-applied-configuration=...
Status:        Running
IP:            10.36.0.0

//
Containers:
  kuard:
    Container ID:  docker://47da8387bfb2ac1...
    Image:         gcr.io/kuar-demo/kuard-amd64:1
    Image ID:      docker-pullable://gcr.io/kuar-demo/kuard-amd64...
    Port:          8080/TCP
    Host Port:     0/TCP
    State:         Running
      Started:     Mon, 16 Jul 2018 10:47:52 +0900
    Ready:         True
    Restart Count:  0
    Environment:   <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-prsng (ro)
Conditions:
  Type            Status
  Initialized      True
  Ready            True
  PodScheduled     True
Volumes:
  default-token-prsng:
    Type:          Secret (a volume populated by a Secret)
    SecretName:    default-token-prsng
    Optional:      false
QoS Class:        BestEffort
Node-Selectors:   <none>
Tolerations:      node.kubernetes.io/not-ready:NoExecute for 300s
                  node.kubernetes.io/unreachable:NoExecute for 300s

//( , , , )
Events:
  Type            Reason              Age             From              Message
  ----            -
  ...
```

Pod

.
30 . .

```
# kubectl delete pods/{{pod name}}
# kubectl delete -f {{pod name}}.yaml
```

Pod

.
, , .

API .

http://localhost:8080 .

```
# kubectl port-forward {{pod name}} 8080:8080
```

--previous

```
# kubectl logs {{pod name}}
# kubectl logs -f {{pod name}}
```

```
# kubectl exec {{pod name}} date
# kubectl exec -it {{pod name}} ash
```

```
# kubectl cp {{pod name}}:/captures/capture1.txt ./capture1.txt
# kubectl cp $HOME/config.txt {{pod name}}:/config.txt
```

Pod (Process Health Check)

process health check

```
...
spec:
  containers:
    - image : {{docker registry}}
      name: {{name}}
      livenessProbe:
        httpGet:
          path: /healthy
          port: 8080
          initialDelaySeconds: 5 // 5
          timeoutSeconds: 1 // 1
          periodSeconds: 10 // 10
          failureThreshold: 3 // 3
      ports:
        - containerPort: 8080
          name: http
          protocol: TCP
```

Pod

, .
0.5 CPU 128, 1.0CPU 256 .

```
...
spec:
  containers:
  - image: {{docker registry}}
    name: {{name}}
    resources:
      requests:
        cpu: "500m"
        memory: "128Mi"
      limits:
        cpu: "1000m"
        memory: "256Mi"
...
```

Pod

.
, .

spec.volumes

.
.

volumeMounts

volumeMounts .

```
...
spec:
  volumes:
  - name: "{{app name}}"
    hostPath:
      path: "{{path}}"
  containers:
  - image: gcr.io/kusr-demo/kuard-amd64:1
    name: {{name}}
    volumeMounts:
    - mountPath: "{{path}}"
      name: "{{app name}}"
...
```

.
.
pod .

NFS .

```
...
volumes:
- name: "{{app name}}"
  nfs:
    server: my.nfs.server.local
    path: "/exports"
...
```

Pod (yaml)

```
apiVersion: v1
kind: Pod
metadata:
  name: {{app name}}
spec:
  volumes:
    - name: "kuard-data"
      nfs:
        server: my.nfs.server.local
        path: "/exports"
  containers:
    - image: gcr.io/kuar-demo/kuard-amd64:1
      name: {{app name}}
      ports:
        - containerPort: 8080
          name: http
          protocol: TCP
  resources:
    requests:
      cpu: "500m"
      memory: "128Mi"
    limits:
      cpu: "1000m"
      memory: "256Mi"
  volumeMounts:
    - mountPath: "/data"
      name: "kuard-data"
  livenessProbe:
    httpGet:
      path: /healthy
      port: 8080
    initialDelaySeconds: 5
    timeoutSeconds: 1
    periodSeconds: 10
    failureThreshold: 3
  readinessProbe:
    httpGet:
      path: /ready
      port: 8080
    initialDelaySeconds: 30
    timeoutSeconds: 1
    periodSeconds: 10
    failureThreshold: 3
```