Health Probes in Kubernetes

Probing is inspecting and monitoring. **Kubernetes** health probes check if the application is healthy, self-healing, and ready to serve traffic.

We have 3 types of Health probes in **Kubernetes:**

- **Liveness Probe** – Health check after pod is created for every interval of seconds. Monitors the application for every certain interval of time. If the application failed or crashed due to intermittent network error or some application error something that could be fixed by restarting the container, Liveness probe takes the necessary action of restarting.
- **Readiness probe** – Ensures that application is ready before it starts serving the traffic to the user. Health check while creating the pod.
- **Startup probe** – Mostly used for slow or legacy container applications. Used in applications that takes lot of time to startup.

Each health probe performs health checks using Http, TCP or command methods.

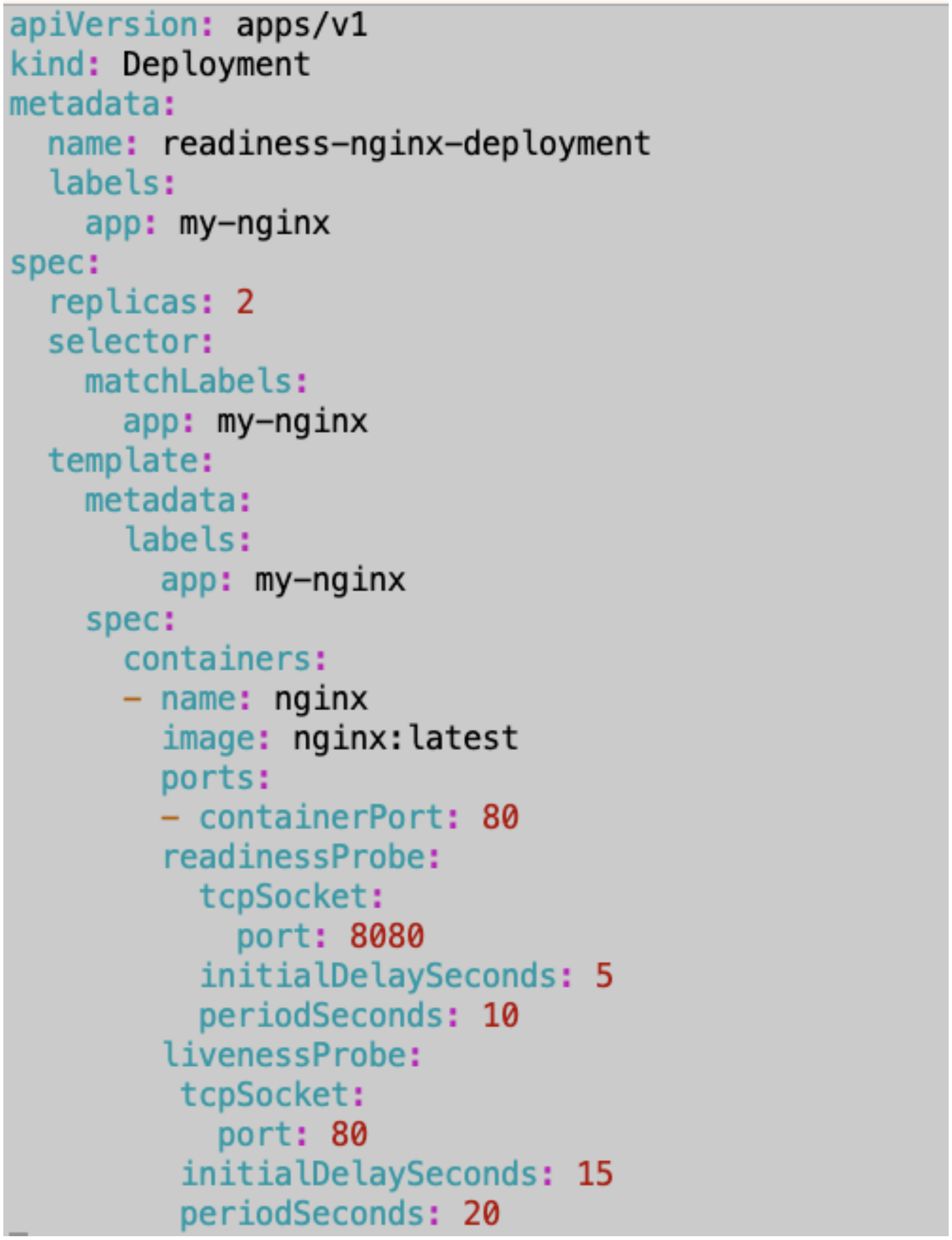
Http method - The probe sends an HTTP GET request to a specified path and port in the container. The container's response determines the health status: Any status code between 200-399 indicates success. Any other status code indicates failure.

TCP method - Kubernetes tries to open a TCP connection to the specified port. For ex: 8080. If the connection is successful, the probe passes. If the connection fails, the probe fails.

Command method - Executes a specified command or script inside the container to determine if it's alive and ready to serve traffic.

Following example shows Readiness and Liveness probes with TCP check

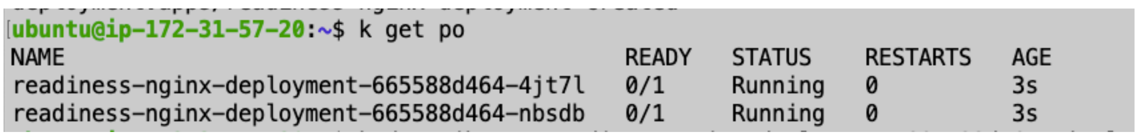
- Nginx Container is deployed at port 80
- Readiness probe is checking at port 8080, which fails



```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: readiness-nginx-deployment
  labels:
    app: my-nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-nginx
  template:
    metadata:
      labels:
        app: my-nginx
    spec:
      containers:
      - name: nginx
        image: nginx:latest
        ports:
        - containerPort: 80
        readinessProbe:
          tcpSocket:
            port: 8080
          initialDelaySeconds: 5
          periodSeconds: 10
        livenessProbe:
         tcpSocket:
           port: 80
         initialDelaySeconds: 15
         periodSeconds: 20
```

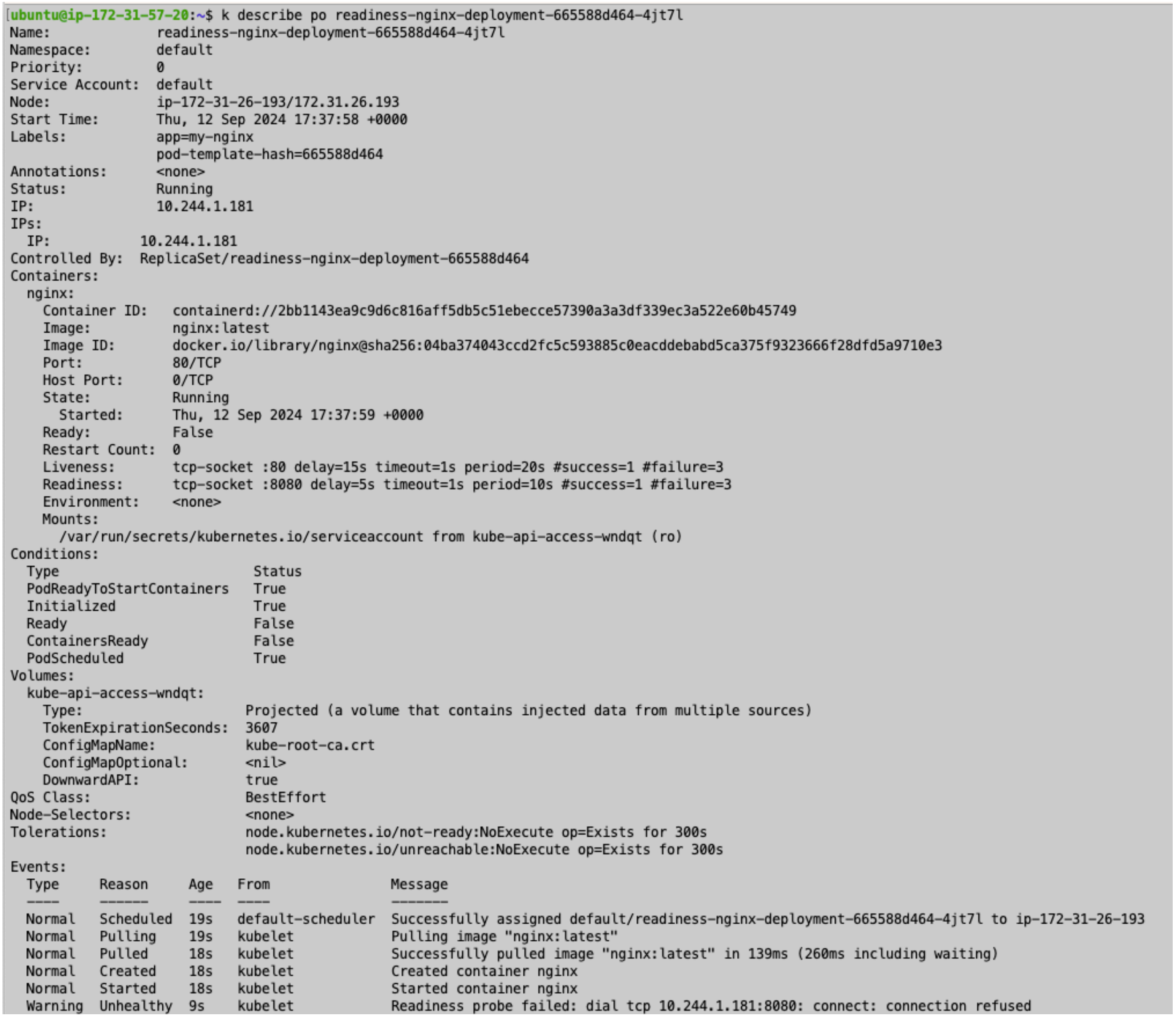
Alias k=kubectl

Getting pods returns the pods are running but not ready as 0/1. initialDelaySeconds: 5, probe runs 5 seconds after deployment and keeps checking every 10 secs as described in periodSeconds: 10

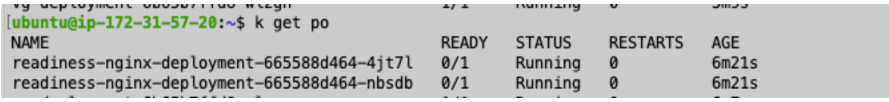


```
[ubuntu@ip-172-31-57-20:~$ k get po
NAME                                          READY   STATUS    RESTARTS   AGE
readiness-nginx-deployment-665588d464-4jt7l   0/1     Running   0          3s
readiness-nginx-deployment-665588d464-nbsdb   0/1     Running   0          3s
```

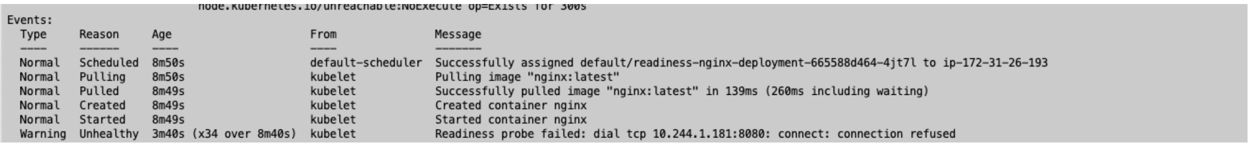
Describing the pod shows that the readiness probe is failed



```
[ubuntu@ip-172-31-57-20:~$ k describe po readiness-nginx-deployment-665588d464-4jt7l
Name:             readiness-nginx-deployment-665588d464-4jt7l
Namespace:        default
Priority:         0
Service Account:  default
Node:             ip-172-31-26-193/172.31.26.193
Start Time:       Thu, 12 Sep 2024 17:37:58 +0000
Labels:           app=my-nginx
                  pod-template-hash=665588d464
Annotations:      <none>
Status:           Running
IP:               10.244.1.181
IPs:
  IP:           10.244.1.181
Controlled By:  ReplicaSet/readiness-nginx-deployment-665588d464
Containers:
  nginx:
    Container ID:   containerd://2bb1143ea9c9d6c816aff5db5c51ebecce57390a3a3df339ec3a522e60b45749
    Image:          nginx:latest
    Image ID:       docker.io/library/nginx@sha256:04ba374043ccd2fc5c593885c0eacddebabd5ca375f9323666f28dfd5a9710e3
    Port:           80/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Thu, 12 Sep 2024 17:37:59 +0000
    Ready:          False
    Restart Count:  0
    Liveness:       tcp-socket :80 delay=15s timeout=1s period=20s #success=1 #failure=3
    Readiness:      tcp-socket :8080 delay=5s timeout=1s period=10s #success=1 #failure=3
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-wndqt (ro)
Conditions:
  Type                        Status
  PodReadyToStartContainers   True
  Initialized                 True
  Ready                       False
  ContainersReady             False
  PodScheduled                True
Volumes:
  kube-api-access-wndqt:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              <none>
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason     Age   From               Message
  ----     ------     ----  ----               -------
  Normal   Scheduled  19s   default-scheduler  Successfully assigned default/readiness-nginx-deployment-665588d464-4jt7l to ip-172-31-26-193
  Normal   Pulling    19s   kubelet            Pulling image "nginx:latest"
  Normal   Pulled     18s   kubelet            Successfully pulled image "nginx:latest" in 139ms (260ms including waiting)
  Normal   Created    18s   kubelet            Created container nginx
  Normal   Started    18s   kubelet            Started container nginx
  Warning  Unhealthy  9s    kubelet            Readiness probe failed: dial tcp 10.244.1.181:8080: connect: connection refused
```



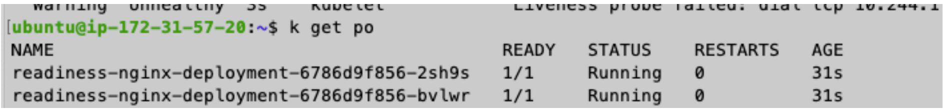
```
[ubuntu@ip-172-31-57-20:~$ k get po
NAME                                          READY   STATUS    RESTARTS   AGE
readiness-nginx-deployment-665588d464-4jt7l   0/1     Running   0          6m21s
readiness-nginx-deployment-665588d464-nbsdb   0/1     Running   0          6m21s
```



```
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason     Age                   From               Message
  ----     ------     ----                  ----               -------
  Normal   Scheduled  8m50s                 default-scheduler  Successfully assigned default/readiness-nginx-deployment-665588d464-4jt7l to ip-172-31-26-193
  Normal   Pulling    8m50s                 kubelet            Pulling image "nginx:latest"
  Normal   Pulled     8m49s                 kubelet            Successfully pulled image "nginx:latest" in 139ms (260ms including waiting)
  Normal   Created    8m49s                 kubelet            Created container nginx
  Normal   Started    8m49s                 kubelet            Started container nginx
  Warning  Unhealthy  3m40s (x34 over 8m40s)  kubelet            Readiness probe failed: dial tcp 10.244.1.181:8080: connect: connection refused
```

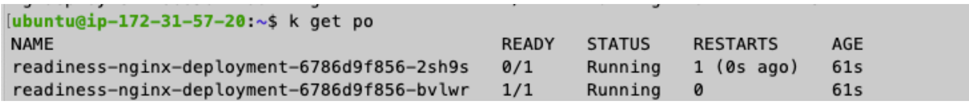
Readiness probe keeps running and the pods are not ready.

To test the Liveness probe, updated the readinessProbe: port to 80 same as container port and kept the Livenessprobe: port as 8080. Now the pods are running.

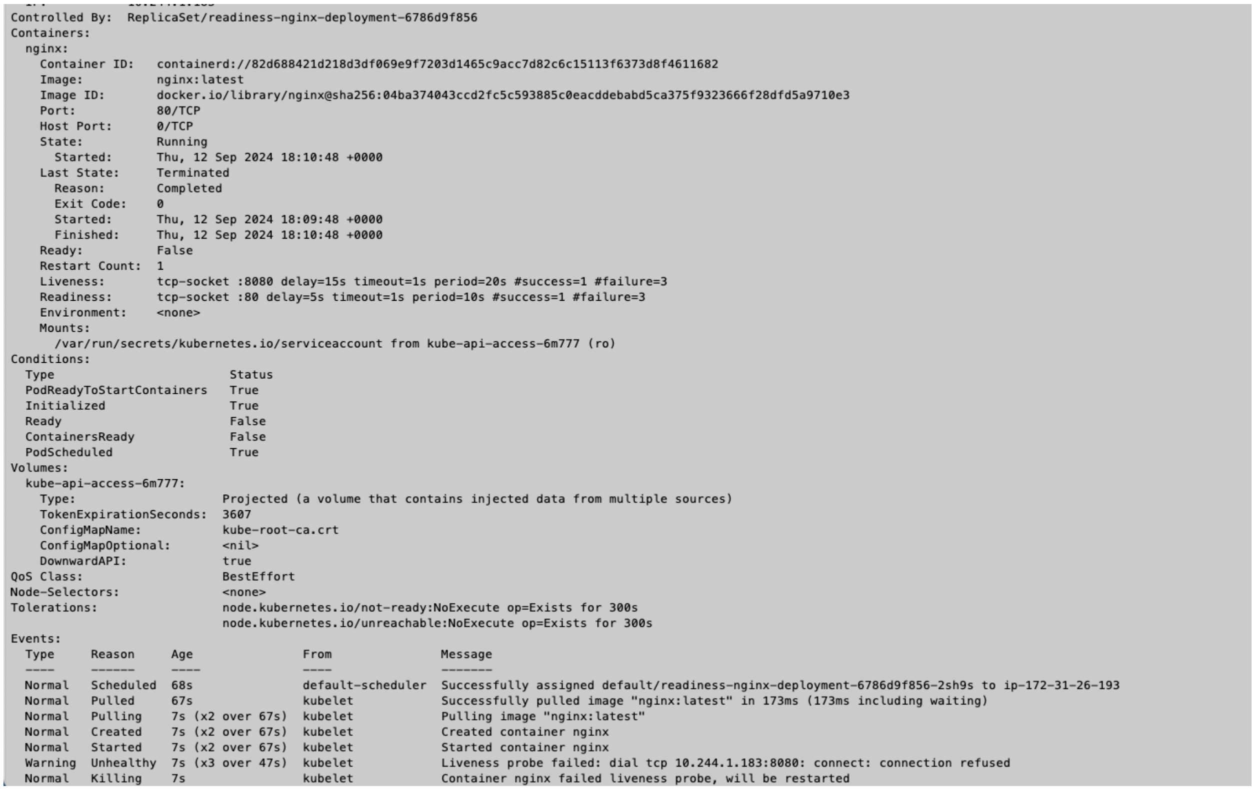


```
  Warning  Unhealthy  3s    kubelet           Liveness probe failed: dial tcp 10.244.1
ubuntu@ip-172-31-57-20:~$ k get po
NAME                                     READY   STATUS    RESTARTS   AGE
readiness-nginx-deployment-6786d9f856-2sh9s   1/1     Running   0          31s
readiness-nginx-deployment-6786d9f856-bvlwr   1/1     Running   0          31s
```

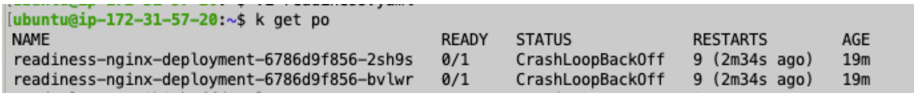
When the Liveness probe is failing, it restarts the pods.



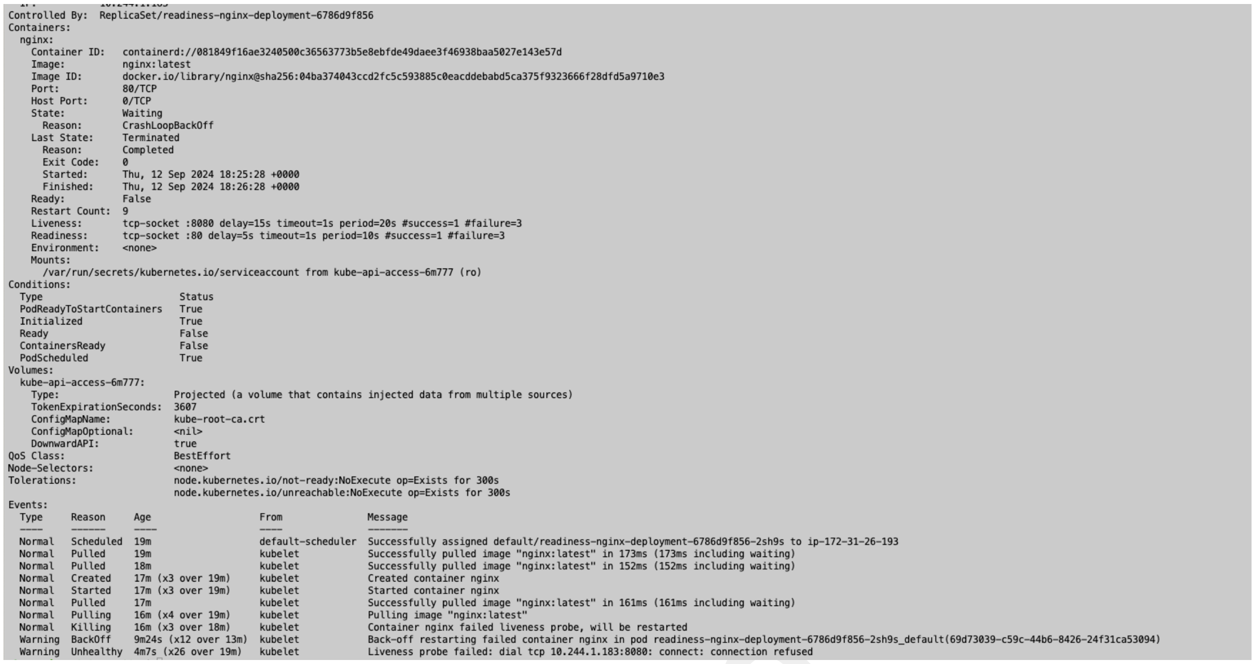
```
ubuntu@ip-172-31-57-20:~$ k get po
NAME                                     READY   STATUS    RESTARTS     AGE
readiness-nginx-deployment-6786d9f856-2sh9s   0/1     Running   1 (0s ago)   61s
readiness-nginx-deployment-6786d9f856-bvlwr   1/1     Running   0            61s
```



```
Controlled By:  ReplicaSet/readiness-nginx-deployment-6786d9f856
Containers:
  nginx:
    Container ID:   containerd://82d688421d218d3df069e9f7203d1465c9acc7d82c6c15113f6373d8f4611682
    Image:          nginx:latest
    Image ID:       docker.io/library/nginx@sha256:04ba374043ccd2fc5c593885c0eacddebabd5ca375f9323666f28dfd5a9710e3
    Port:           80/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Thu, 12 Sep 2024 18:10:48 +0000
    Last State:     Terminated
      Reason:       Completed
      Exit Code:    0
      Started:      Thu, 12 Sep 2024 18:09:48 +0000
      Finished:     Thu, 12 Sep 2024 18:10:48 +0000
    Ready:          False
    Restart Count:  1
    Liveness:       tcp-socket :8080 delay=15s timeout=1s period=20s #success=1 #failure=3
    Readiness:      tcp-socket :80 delay=5s timeout=1s period=10s #success=1 #failure=3
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-6m777 (ro)
Conditions:
  Type                        Status
  PodReadyToStartContainers   True
  Initialized                 True
  Ready                       False
  ContainersReady             False
  PodScheduled                True
Volumes:
  kube-api-access-6m777:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              <none>
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason     Age                From               Message
  ----     ------     ----               ----               -------
  Normal   Scheduled  68s                default-scheduler  Successfully assigned default/readiness-nginx-deployment-6786d9f856-2sh9s to ip-172-31-26-193
  Normal   Pulled     67s                kubelet            Successfully pulled image "nginx:latest" in 173ms (173ms including waiting)
  Normal   Pulling    7s (x2 over 67s)   kubelet            Pulling image "nginx:latest"
  Normal   Created    7s (x2 over 67s)   kubelet            Created container nginx
  Normal   Started    7s (x2 over 67s)   kubelet            Started container nginx
  Warning  Unhealthy  7s (x3 over 47s)   kubelet            Liveness probe failed: dial tcp 10.244.1.183:8080: connect: connection refused
  Normal   Killing    7s                 kubelet            Container nginx failed liveness probe, will be restarted
```

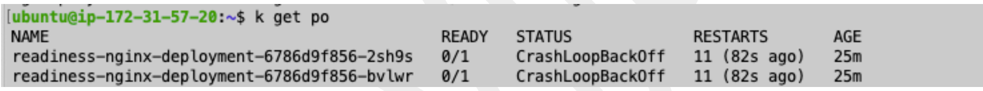


```
ubuntu@ip-172-31-57-20:~$ k get po
NAME                                     READY   STATUS             RESTARTS        AGE
readiness-nginx-deployment-6786d9f856-2sh9s   0/1     CrashLoopBackOff   9 (2m34s ago)   19m
readiness-nginx-deployment-6786d9f856-bvlwr   0/1     CrashLoopBackOff   9 (2m34s ago)   19m
```



```
IP:           10.244.1.183
Controlled By:  ReplicaSet/readiness-nginx-deployment-6786d9f856
Containers:
  nginx:
    Container ID:   containerd://081849f16ae3240500c36563773b5e8ebfde49daee3f46938baa5027e143e57d
    Image:          nginx:latest
    Image ID:       docker.io/library/nginx@sha256:04ba374043ccd2fc5c593885c0eacddebabd5ca375f9323666f28dfd5a9710e3
    Port:           80/TCP
    Host Port:      0/TCP
    State:          Waiting
      Reason:       CrashLoopBackOff
    Last State:     Terminated
      Reason:       Completed
      Exit Code:    0
      Started:      Thu, 12 Sep 2024 18:25:28 +0000
      Finished:     Thu, 12 Sep 2024 18:26:28 +0000
    Ready:          False
    Restart Count:  9
    Liveness:       tcp-socket :8080 delay=15s timeout=1s period=20s #success=1 #failure=3
    Readiness:      tcp-socket :80 delay=5s timeout=1s period=10s #success=1 #failure=3
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-6m777 (ro)
Conditions:
  Type                        Status
  PodReadyToStartContainers   True
  Initialized                 True
  Ready                       False
  ContainersReady             False
  PodScheduled                True
Volumes:
  kube-api-access-6m777:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              <none>
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason     Age                   From               Message
  ----     ------     ----                  ----               -------
  Normal   Scheduled  19m                   default-scheduler  Successfully assigned default/readiness-nginx-deployment-6786d9f856-2sh9s to ip-172-31-26-193
  Normal   Pulled     19m                   kubelet            Successfully pulled image "nginx:latest" in 173ms (173ms including waiting)
  Normal   Pulled     18m                   kubelet            Successfully pulled image "nginx:latest" in 152ms (152ms including waiting)
  Normal   Created    17m (x3 over 19m)     kubelet            Created container nginx
  Normal   Started    17m (x3 over 19m)     kubelet            Started container nginx
  Normal   Pulled     17m                   kubelet            Successfully pulled image "nginx:latest" in 161ms (161ms including waiting)
  Normal   Pulling    16m (x4 over 19m)     kubelet            Pulling image "nginx:latest"
  Normal   Killing    16m (x3 over 18m)     kubelet            Container nginx failed liveness probe, will be restarted
  Warning  BackOff    9m24s (x12 over 13m)  kubelet            Back-off restarting failed container nginx in pod readiness-nginx-deployment-6786d9f856-2sh9s_default(69d73039-c59c-44b6-8426-24f31ca53094)
  Warning  Unhealthy  4m7s (x26 over 19m)   kubelet            Liveness probe failed: dial tcp 10.244.1.183:8080: connect: connection refused
```

Every time the pod is restarted, Kubernetes waits for a longer and longer time, known as a “backoff delay”. The delay between restarts is exponential (10s, 20s, 40s, …) and is capped at five minutes. During this process, Kubernetes displays the CrashLoopBackOff error.

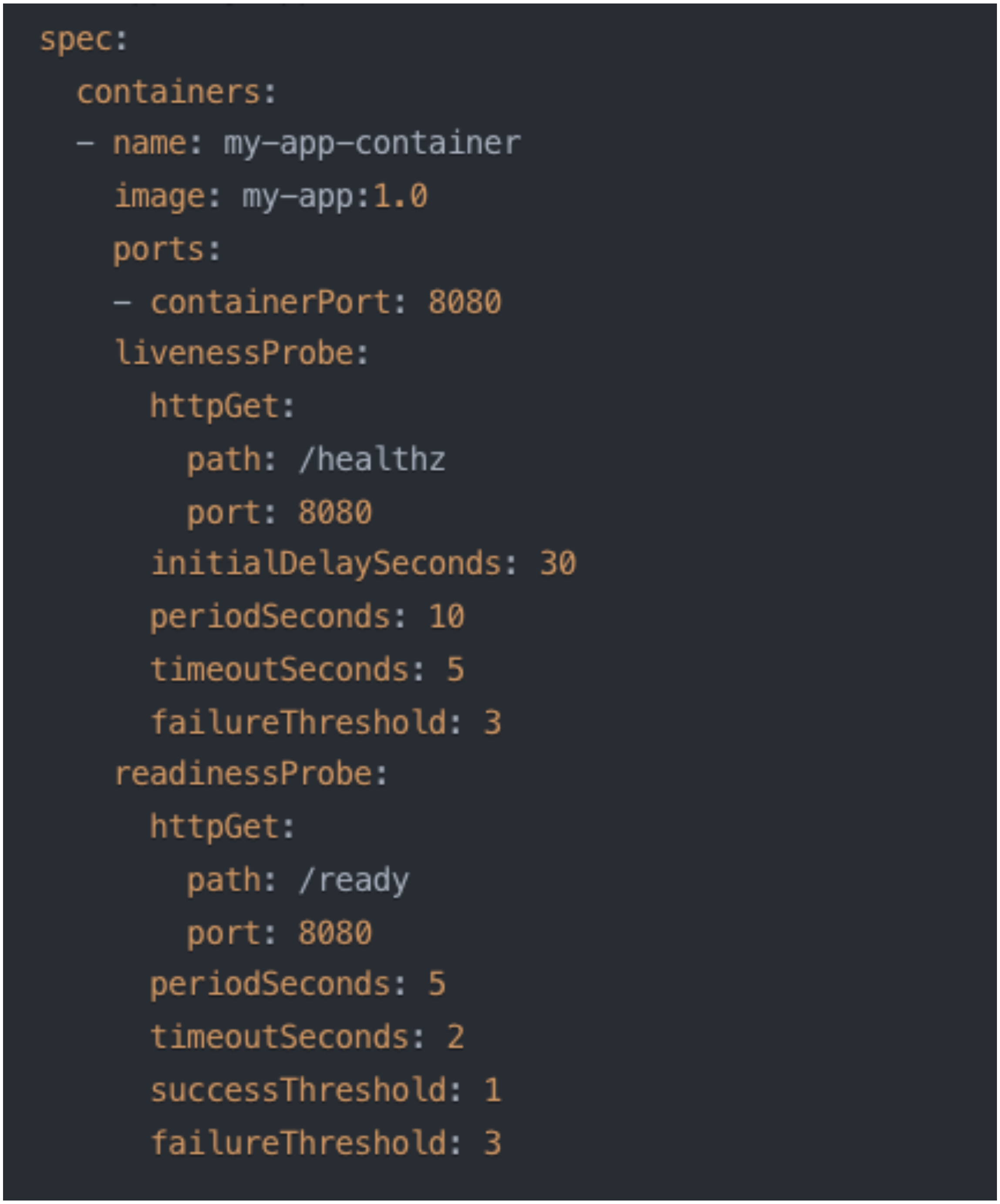


```
ubuntu@ip-172-31-57-20:~$ k get po
NAME                                        READY   STATUS             RESTARTS       AGE
readiness-nginx-deployment-6786d9f856-2sh9s   0/1     CrashLoopBackOff   11 (82s ago)   25m
readiness-nginx-deployment-6786d9f856-bvlwr   0/1     CrashLoopBackOff   11 (82s ago)   25m
```

Example code for Http check

In the pod or deployment specification, you can define:

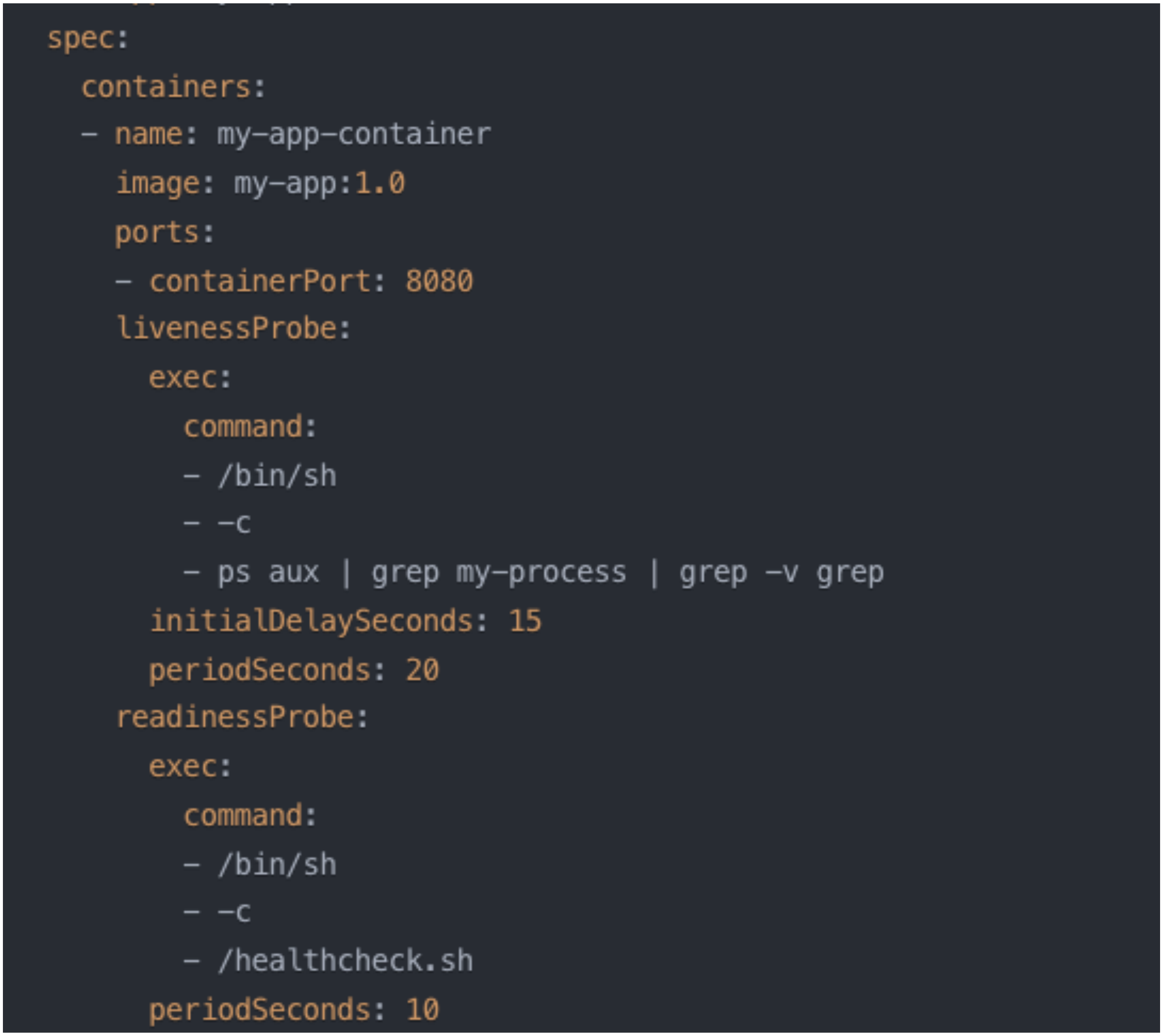
- The endpoint path (e.g., '/healthz')
- The port to send the request to
- Various timing parameters (initial delay, frequency, timeout, etc.)



```yaml
spec:
  containers:
  - name: my-app-container
    image: my-app:1.0
    ports:
    - containerPort: 8080
    livenessProbe:
      httpGet:
        path: /healthz
        port: 8080
      initialDelaySeconds: 30
      periodSeconds: 10
      timeoutSeconds: 5
      failureThreshold: 3
    readinessProbe:
      httpGet:
        path: /ready
        port: 8080
      periodSeconds: 5
      timeoutSeconds: 2
      successThreshold: 1
      failureThreshold: 3
```

Example code for Command check

- Kubernetes runs the specified command inside the container.
- If the command exits with a status code of 0, the probe passes.
- If the command exits with any other status code, the probe fails.



```yaml
spec:
  containers:
  - name: my-app-container
    image: my-app:1.0
    ports:
    - containerPort: 8080
    livenessProbe:
      exec:
        command:
        - /bin/sh
        - -c
        - ps aux | grep my-process | grep -v grep
      initialDelaySeconds: 15
      periodSeconds: 20
    readinessProbe:
      exec:
        command:
        - /bin/sh
        - -c
        - /healthcheck.sh
      periodSeconds: 10
```