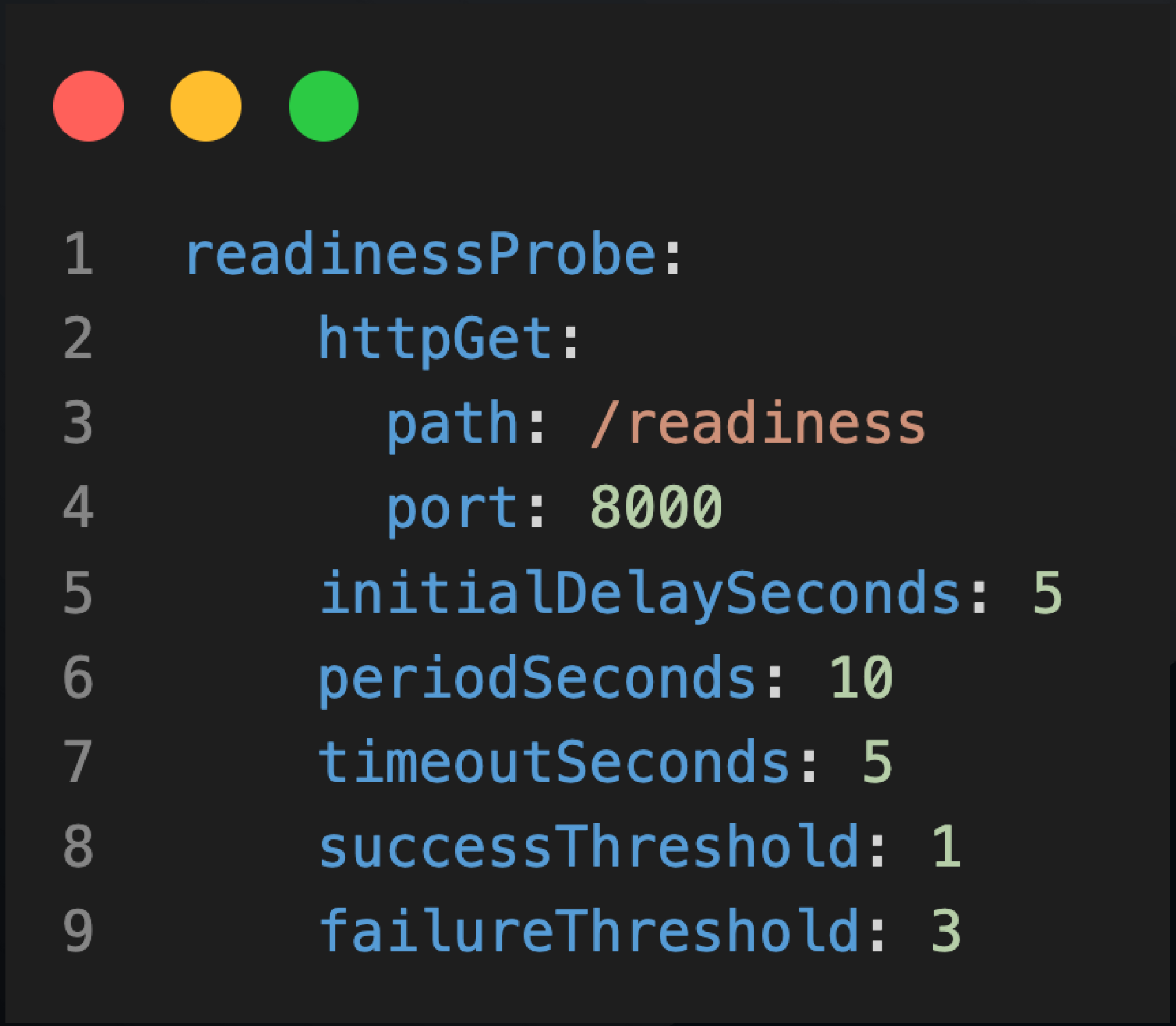
OBSERVABILITY

- **liveness probe** and **readiness probe** are two types of health checks that are used to determine the availability and status of a container running in a pod.
- **Readiness probe**, on the other hand, is used to determine **whether a container is ready to receive incoming requests**
- **Liveness probe** is used to determine **whether a container is still running and responding to requests.**

READINESS PROBE

- **Readiness probe** is used to determine **whether a container is ready to receive incoming requests**
- If a readiness probe fails, Kubernetes **will remove the container from the service endpoints**, preventing incoming requests from being routed to the container
- A readiness probe is typically used to **ensure that a container has completed its startup process and is ready to handle incoming traffic** before being added to a load balancer or service.

READINESS PROBE

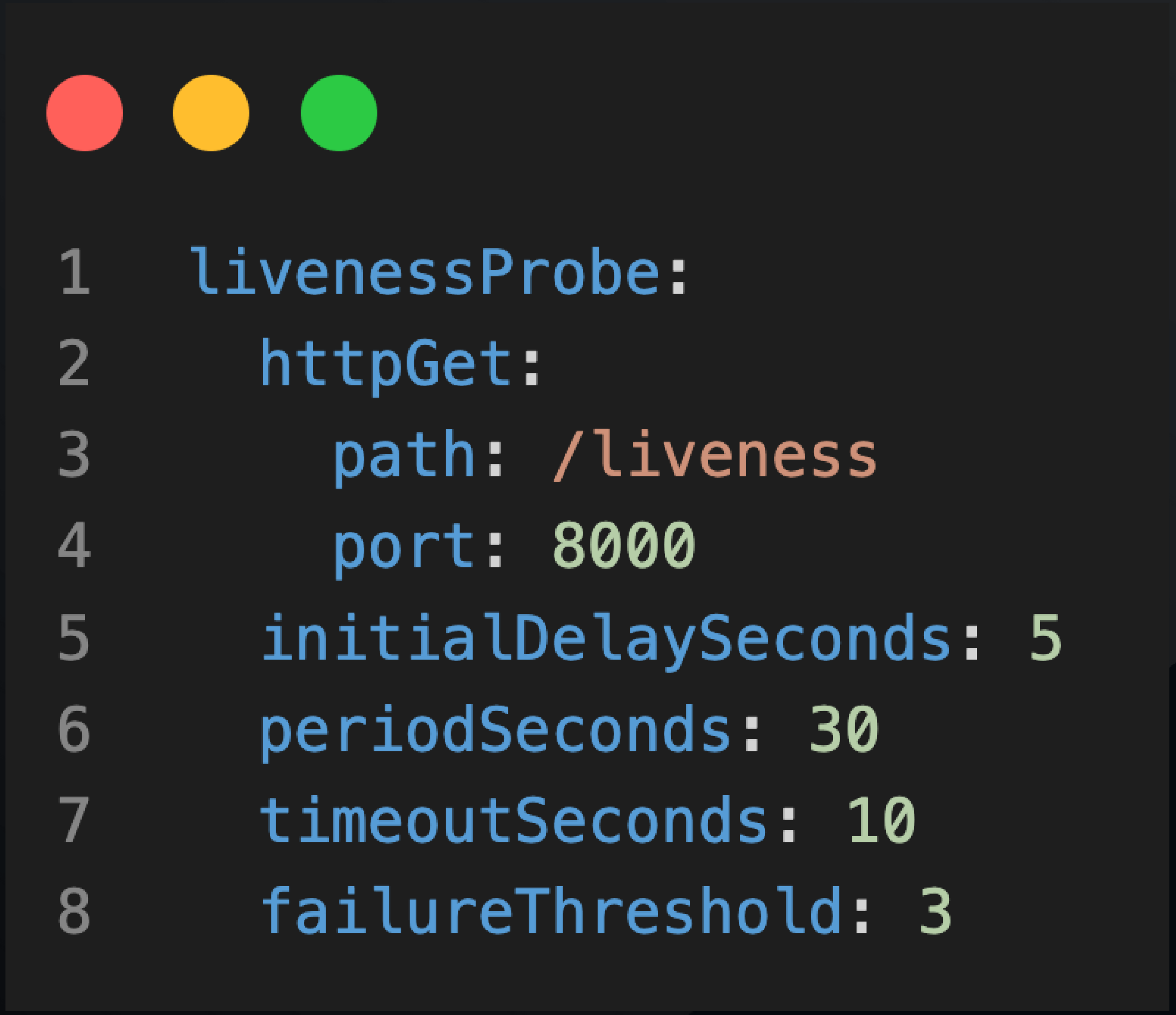


```yaml
readinessProbe:
    httpGet:
      path: /readiness
      port: 8000
    initialDelaySeconds: 5
    periodSeconds: 10
    timeoutSeconds: 5
    successThreshold: 1
    failureThreshold: 3
```

LIVENESS PROBE

- **Liveness probe** is used to determine **whether a container is still running** and responding to requests.
- If a liveness probe fails, Kubernetes will **attempt to restart the container**
- A liveness probe is typically used to **detect and recover from issues** such as application crashes, deadlocks, or resource starvation

LIVENESS PROBE



```yaml
livenessProbe:
  httpGet:
    path: /liveness
    port: 8000
  initialDelaySeconds: 5
  periodSeconds: 30
  timeoutSeconds: 10
  failureThreshold: 3
```