

# Liveness/ Readiness Probes

Microservice Resiliency 를 지원하는 쿠버네티스의 3 가지 Probe 타입에 대해 이해하고 실습을 통해 확인한다.

Instruction

## Liveness & Readiness

- `pwd` 로 현 위치가 아래의 경로인지 확인  
`/container-orchestration/yaml/liveness#`

### Liveness- Command Probe

```
kubectl apply -f exec-liveness.yaml
```

- 컨테이너가 **Running** 상태로 보이나, **Liveness Probe** 실패로 재시작
- `kubectl describe` 로 실패 메시지 확인

```
kubectl describe po liveness-exec
```

### Liveness- HTTP Probe

- **Liveness Probe** 가 적용된 주문 마이크로서비스를 배포한다.

```
kubectl apply -f
```

```
https://raw.githubusercontent.com/acmexii/demo/master/edu/order-liveness.yaml
```

- 배포된 **Order** 에 대해 서비스를 생성한다.

```
kubectl expose deploy order --type=LoadBalancer --port=8080
```

```
kubectl get svc
```

- **Order Liveness Probe** 를 명시적으로 **Fail** 상태로 전환한다.

```
# Liveness Probe 확인
```

```
http EXTERNAL-IP:8080/actuator/health
```

```
# Liveness Probe Fail 설정 및 확인
```

```
http put EXTERNAL-IP:8080/actuator/down
```

```
http EXTERNAL-IP:8080/actuator/health
```

- **Probe Fail** 에 따른 쿠버네티스 동작확인

```
kubectl get pod
```

```
kubectl describe pod/[ORDER-POD 객체]
```

### Rediness- HTTP Probe

- 배송 마이크로서비스를 배포한다.

```
kubectl apply -f
```

```
https://raw.githubusercontent.com/acmexii/demo/master/edu/delivery-readiness-v1.yaml
```

```
kubectl expose deploy delivery --port=8080
```

- **Siege** (로더제너레이터)를 설치하고 해당 컨테이너로 접속한다.

```
kubectl create deploy siege --image=ghcr.io/acmexii/siege-nginx:latest
```

```
kubectl exec pod/[SIEGE-POD 객체] -it -- /bin/bash
```

- **Delivery** 서비스에 대해 **Siege** 로 부하를 충분히 걸어준다.
- **Siege** 부하가 걸린 상태에서 **Delivery** 서비스를 **v2** 로 **Rollout** 한다.
- **Siege** 테스트 결과를 모니터링한다.

```
kubectl get deploy -o wide
```

```
siege -v -c1 -t80S http://delivery:8080/deliveries
```

```
kubectl apply -f
```

```
https://raw.githubusercontent.com/acmexii/demo/master/edu/delivery-no-rediness-v2.yaml
```

- 신규 버전으로 롤아웃되는 과정에 네트워크 장애가 확인된다. (**Availability 100% 미만**)

```
kubectl get deploy -o wide
```

- **Delivery** 서비스에 대해 **Siege** 부하를 충분히 걸어준다 상태에서 이제는 **Delivery** 서비스를 **v3** 로 **Rollout** 한다.
- **Siege** 테스트 결과를 모니터링한다.

```
siege -v -c1 -t60S http://delivery:8080/deliveries
```

```
kubectl apply -f
```

```
https://raw.githubusercontent.com/acmexii/demo/master/edu/delivery-rediness-v3.yaml
```

- 신규 버전으로 롤아웃되는 과정에 네트워크 오류가 없다. (**Availability 100% 보장**)

```
kubectl get deploy -o wide
```

---

## CheckPoints

1. 모든 요구사항을 만족하는가 ☐