

# 微服务远程调试

庄严谨 / 苟荣 / 周丽莎

# 背景

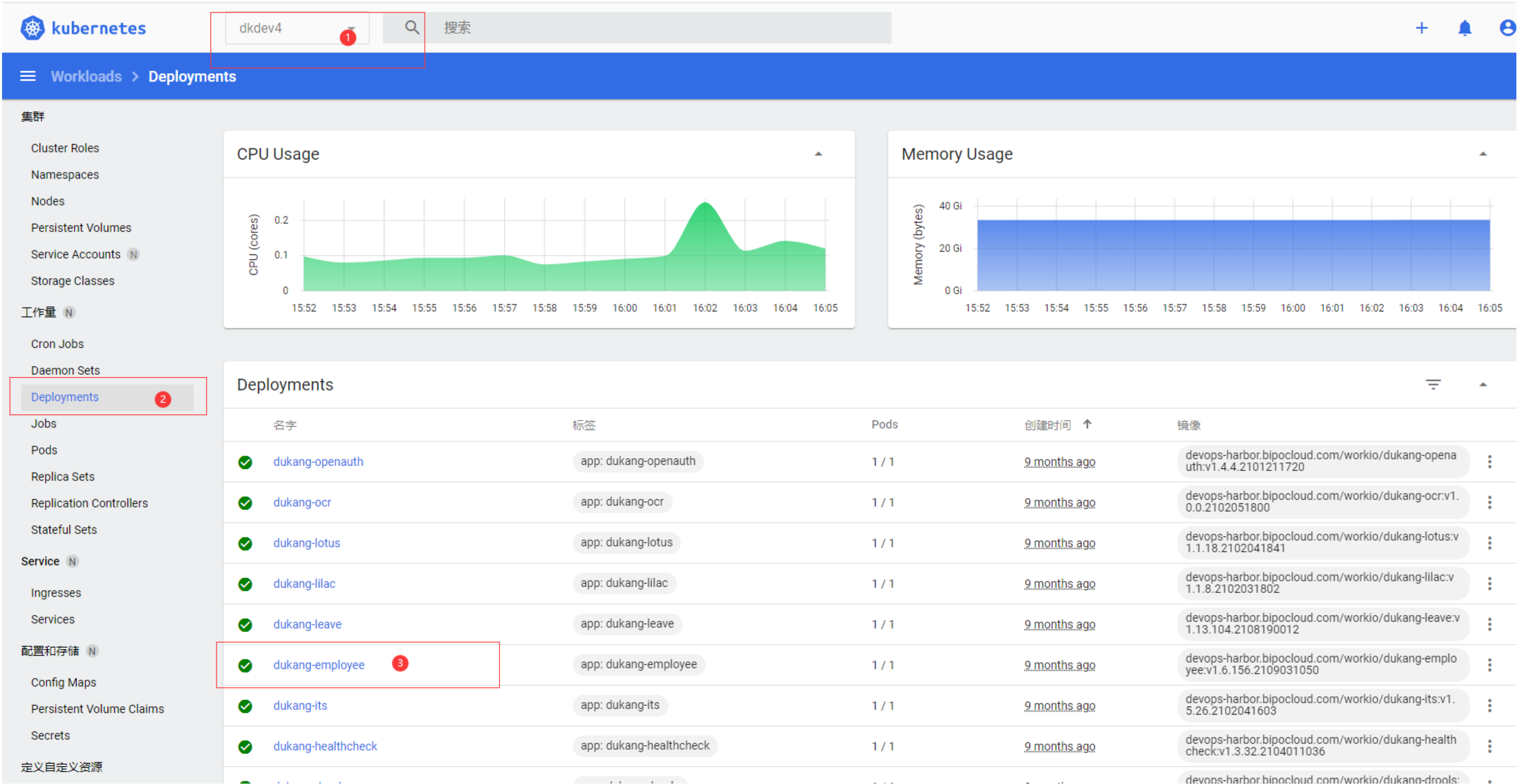
- 本地启动需要的依赖过多，例如：数据库、Eureka、LogStash、Grappa、IdGenerator等
- 调试某个接口需要准备的数据比较复杂
- 调试本地无法重现，线上环境可以重现的问题
- 远程调试可以即时查看运行时变量的值，无需通过日志等方式查看变量等信息
- 远程调试的便捷性：可以在网页上直接进行操作，无需通过Postman等客户端模拟请求内容或者本地启动前端服务进行操作
- 方便测试和开发同学协调复现问题

# 目录

- **k8s开启远程调试**
- **本地通过kubectl命令连接k8s调试端口**
- **通过IDE远程调试**
- **Demo**

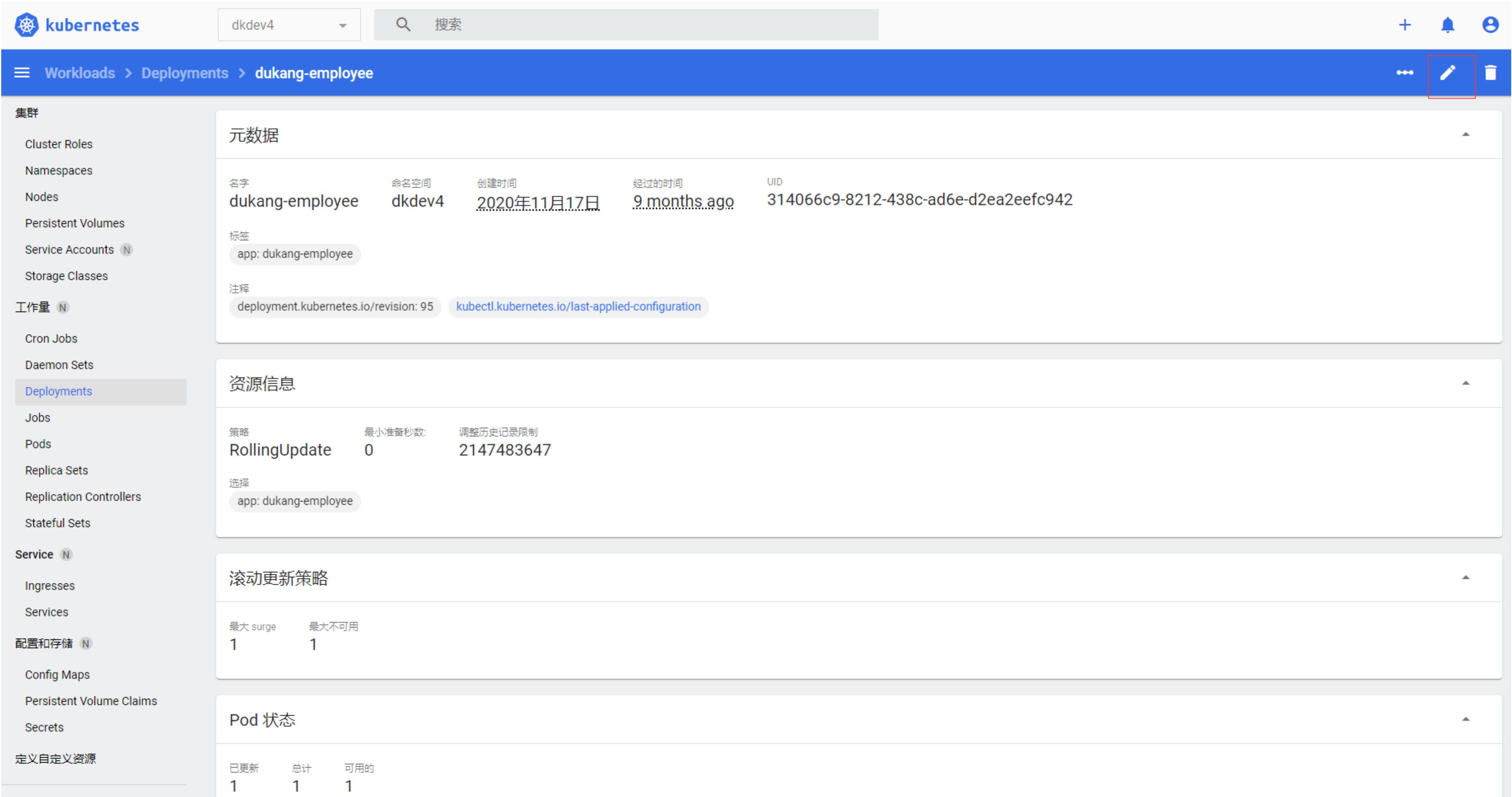
# k8s开启远程调试

1.打开K8S仪表盘， 选择namespace,点击Deployments,找到对应的应用。



# k8s开启远程调试

## 2.通过Kubernetes清单（json / yaml） 设置启动参数并公开调试端口



# k8s开启远程调试

## 3.设置Java启动参数启动远程调试

- '-agentlib:jdwp=transport=dt\_socket,server=y,suspend=n,address=5000'

编辑资源

YAMLJSON

27 spec:

28 volumes:

29 - name: logs-volume

30 hostPath:

31 path: /appdata/dkdev4/logs

32 type: ''

33 containers:

34 - name: dukang-employee

35 image: >-

36 devops-harbor.bipocloud.com/workio/dukang-employee:v1.6.156.2109031050

37 command:

38 - java

39 - '-Xms512m'

40 - '-Xmx512m'

41 - '-jar'

42 - '-agentlib:jdwp=transport=dt\_socket,server=y,suspend=n,address=5000'

43 - dukang-employee.jar

44 ports:

45 - containerPort: 8080

46 protocol: TCP

47 - containerPort: 5000

48 protocol: TCP

49 env:

50 - name: SPRING\_PROFILES\_ACTIVE

51 value: dkdev4

增加虚拟机参数，address为上一步中配置的端口

此操作相当于：kubectl apply -f <spec.yaml>

# k8s开启远程调试

## 4.公开调试端口

- containerPort: 5000  
protocol: TCP

编辑资源

YAMLJSON

34

- name: dukang-employee

35

image: >-

36

devops-harbor.bipocloud.com/workio/dukang-employee:v1.6.156.2109031050

37

command:

38

- java

39

- '-Xms512m'

40

- '-Xmx512m'

41

- '-jar'

42

- '-agentlib:jdwp=transport=dt\_socket,server=y,suspend=n,address=5000'

43

- dukang-employee.jar

44

ports:

45

- containerPort: 8080

46

protocol: TCP

47

- containerPort: 5000

48

protocol: TCP

49

env:

50

- name: SPRING\_PROFILES\_ACTIVE

51

value: dkdev4

52

- name: EUREKA\_USERNAME

53

value: dkdev4

54

- name: EUREKA\_PASSWORD

55

value: dkdev4

56

- name: BOOTSTRAPSERVERS

57

value: '192.168.9.236:9092'

58

resources: {}

web端口

调试端口

i

此操作相当于: kubectl apply -f <spec.yaml>

更新

取消

# 本地通过kubectl命令连接k8s调试端口

执行命令 `kubectl --kubeconfig=configPath port-forward podName localPort:remotePort -n nameSpace`

configPath: kubeconfig文件地址

podName: pod名字

localPort: 本地代理端口

remotePort: 远程调试端口

nameSpace: 命名空间

示例：

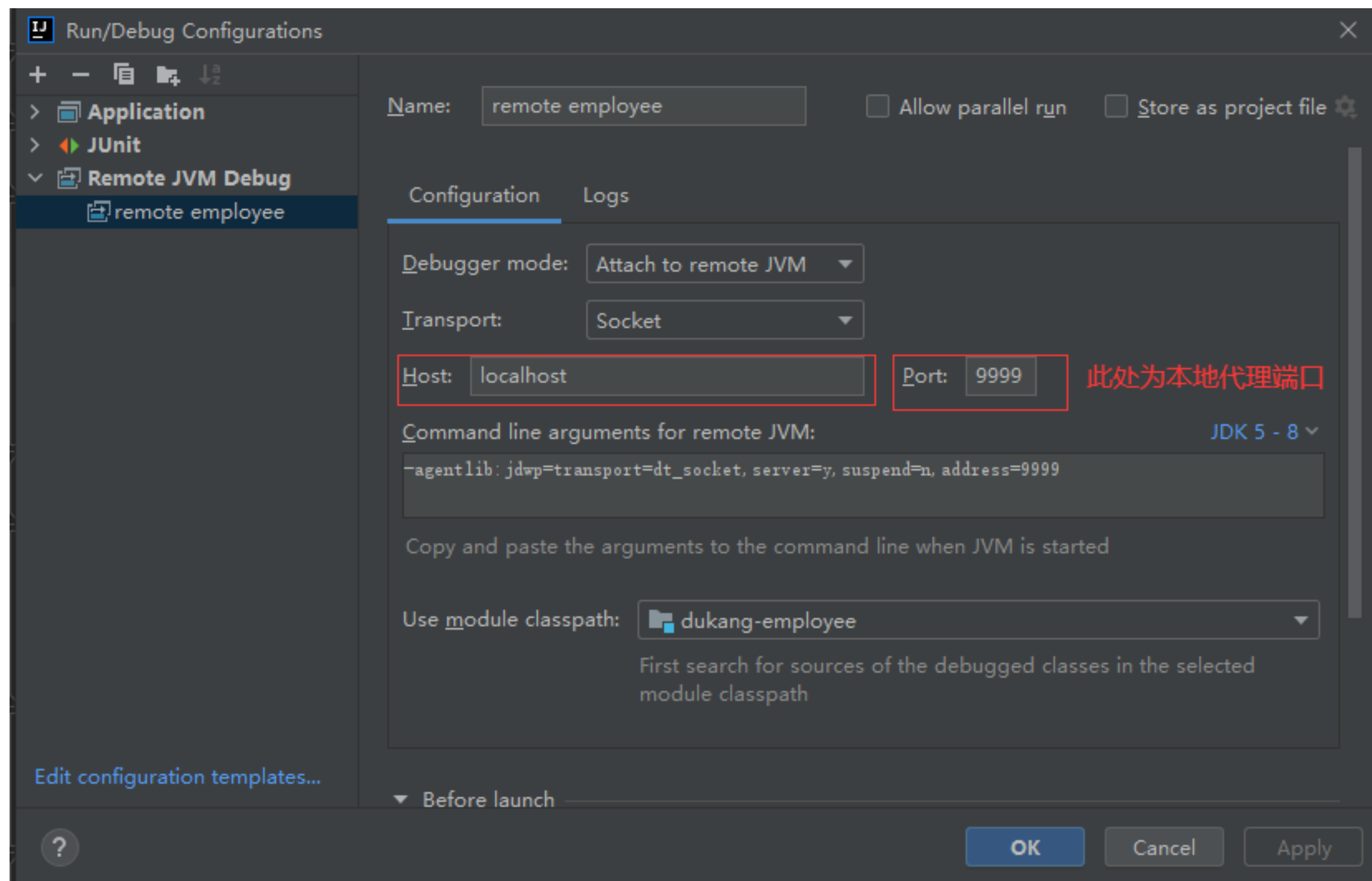
```
kubectl --kubeconfig=./iacdev-k8s.kubeconfig port-forward dukang-employee-578697c5cb-txj7f 9999:5000 -n dkdev4
```

远程调试快捷命令：`sh remote.sh dkdev2 dukang-commission 9997`

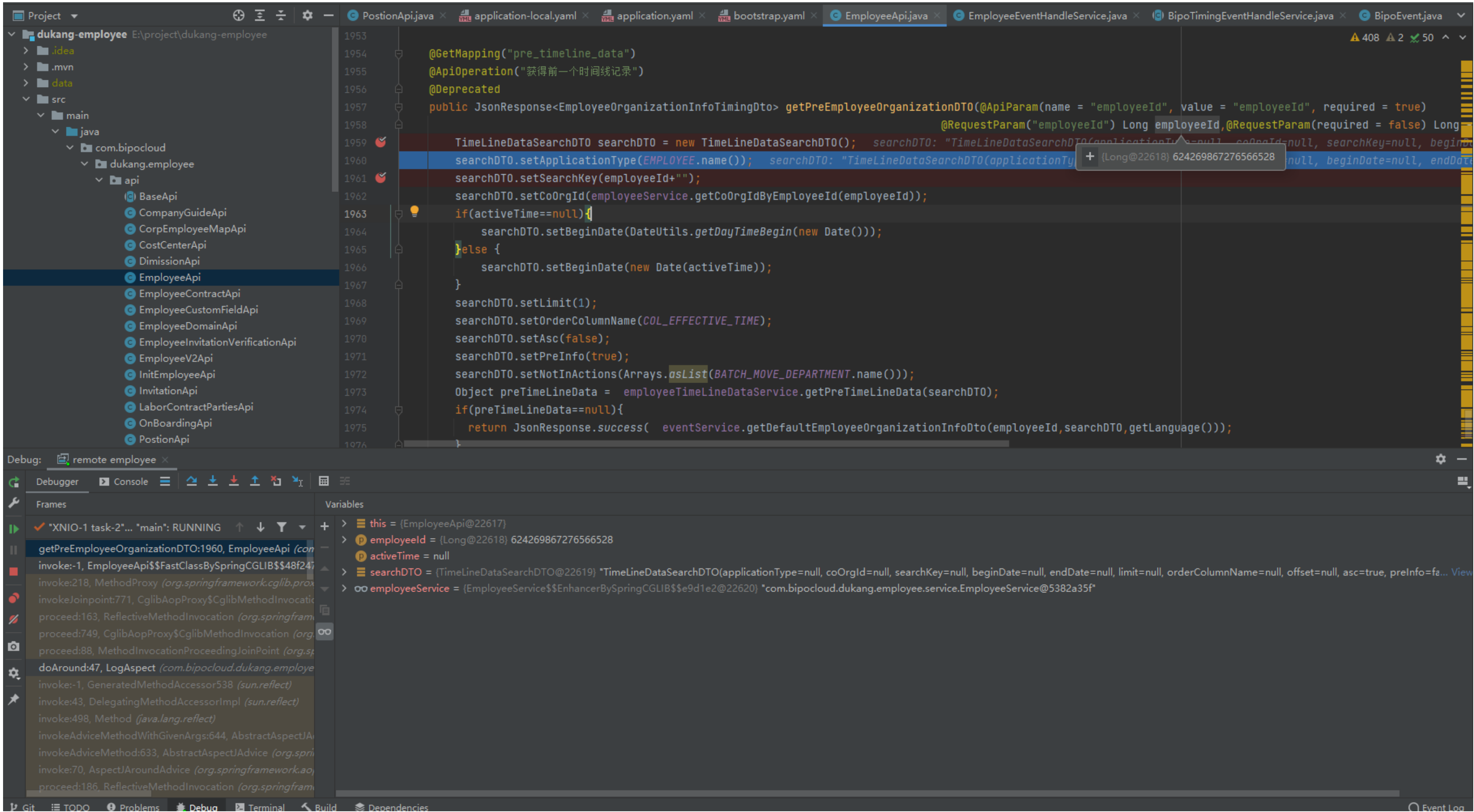


# 通过IDE远程调试

## (以IDEA为例，Eclipse同理)



# 通过IDE远程调试



# Demo

# 问题 / Tips

- 📁 本地调试为单线程执行，若开发人员在断点调试某个接口，影响整个app的请求，导致系统无法使用
- 同一环境的同一个app，同一时间只能允许一名开发人员进行远程调试
- 需保证本地代码和远程运行代码一致，否则断点出现错行问题
- 远程调试过程中运行日志不会在本地打印输出