nginx&&nginx-ingress实现灰度发布

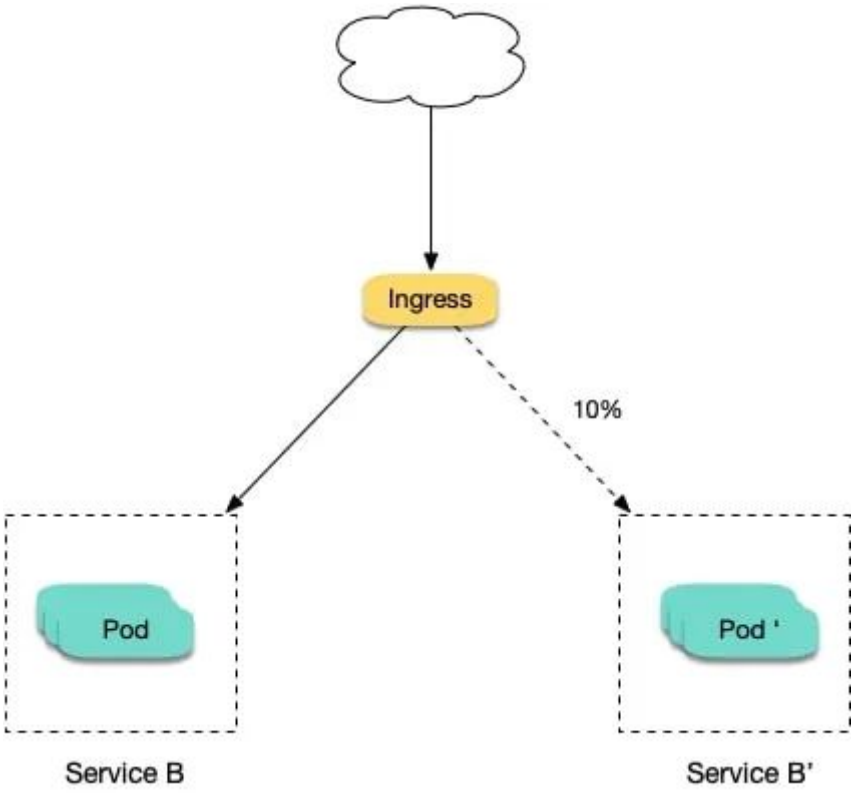
一、名词解释

灰度发布：又名金丝雀发布、黑白发布。18世纪时，由于科技落后，矿井没有好的通风条件和瓦斯检测工具，旷工在矿井作业时经常出现瓦斯中毒现象，因此，旷工们在下井时，会携带一只金丝雀，因为金丝雀在状态好的时

软件新版本发布后，我们先开放给一部分用户先体验和测试，等系统稳定运行一段时间足够稳定了再逐渐全量上线新版本，最后平滑下线旧版本。称之为灰度发布

- 基于用户体系的灰度发布，针对一些特定的人群，为其写入特定的header或者cookie。

- 根据流量规则，随机抽取部分人群参与灰度测试



二、该选那个呢

建议优选cookie,理由如下

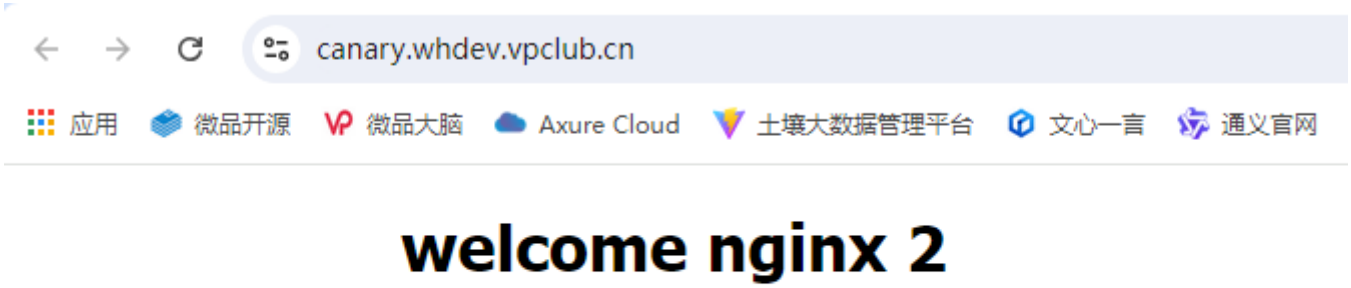
可以控制
基于header可能更便于c/s端，因为b/s无法直接设置header

三、基于nginx-ingress的灰度发布

先创建2个nginx服务
服务1



服务2





- 创建2个ingress

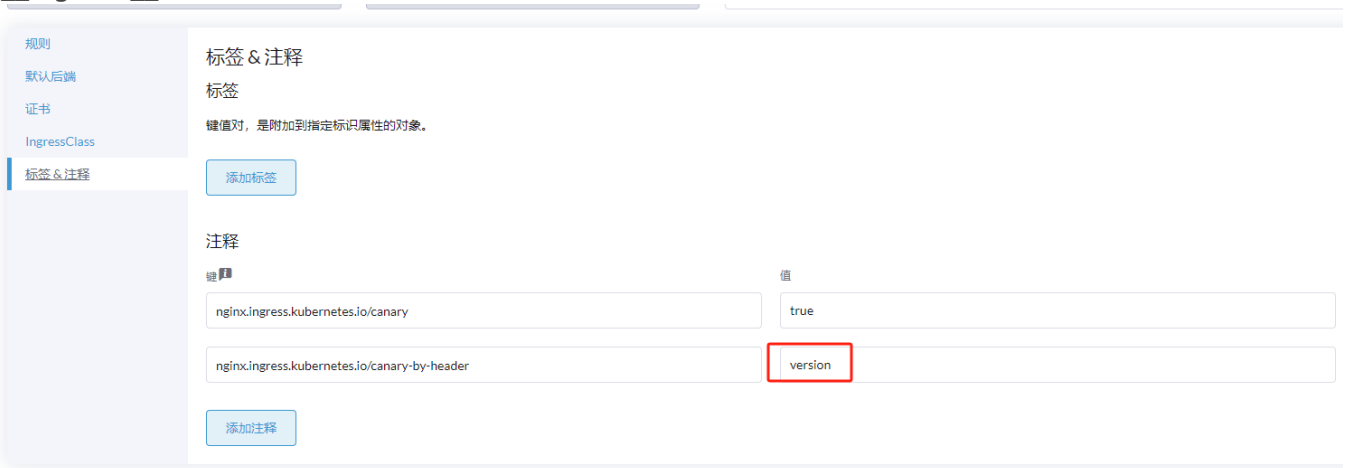
不出意外的话，第二个会创建不成功，需要在第二个ingress加入nginx-ingress注释

```
# 启用灰度发布
nginx.ingress.kubernetes.io/canary: "true"

# 启用基于header的灰度规则，并指定header包含指定键(version)，客户端传递该header键并将值指定为always(总是)或者never(永远不)
nginx.ingress.kubernetes.io/canary-by-header: "version"

# 指定基于header规则时，需要包含的值，不在局限于值为always(总是)或者never(永远不)
nginx.ingress.kubernetes.io/canary-by-header-value: "2.0"


# 启用基于cookie的灰度规则，并指定cookie包含指定键(version)，客户端传递该cookie键并将值指定为always(总是)或者never(永远不)
nginx.ingress.kubernetes.io/canary-by-cookie: "version"

# 网上有部分文章描述说有类似header规则的nginx.ingress.kubernetes.io/canary-by-cookie-value注解指定值，实际上没有这个注解

# 基于流量规则，值为0-100之间，既按百分比，0为不接受任何请求，其最大值(100)可以由canary-weight-total定义
nginx.ingress.kubernetes.io/canary-weight: 10

# 基于流量规则时候的最大值，默认为100，该设置可以忽略
nginx.ingress.kubernetes.io/canary-weight-total

# 规则优先级
# canary-by-header -> canary-by-cookie -> canary-weight
```

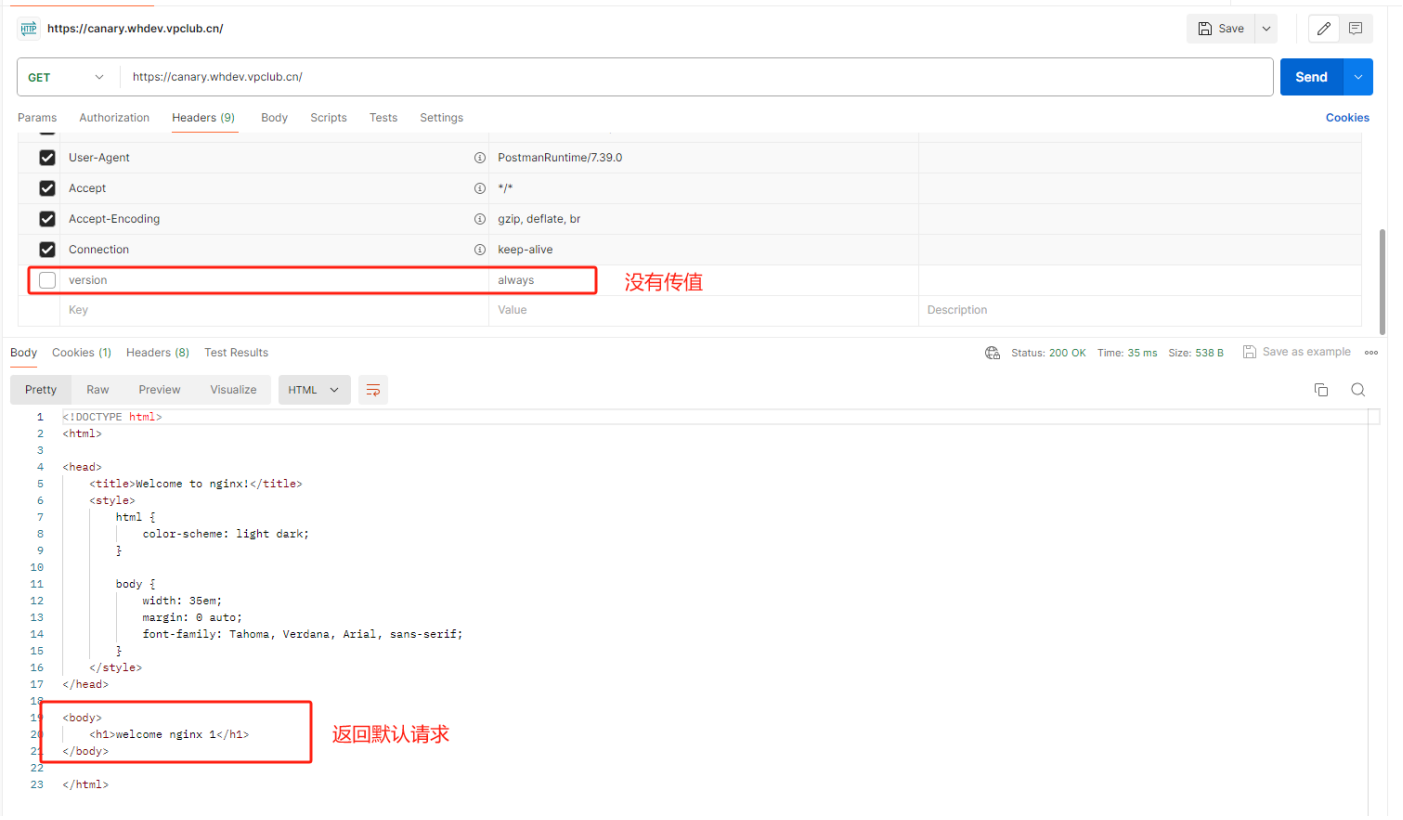
```
# □□□□
# https://github.com/kubernetes/ingress-nginx/blob/main/docs/user-guide/nginx-configuration/annotations.md#canary
```

□□□□□□□header□□□□□□□

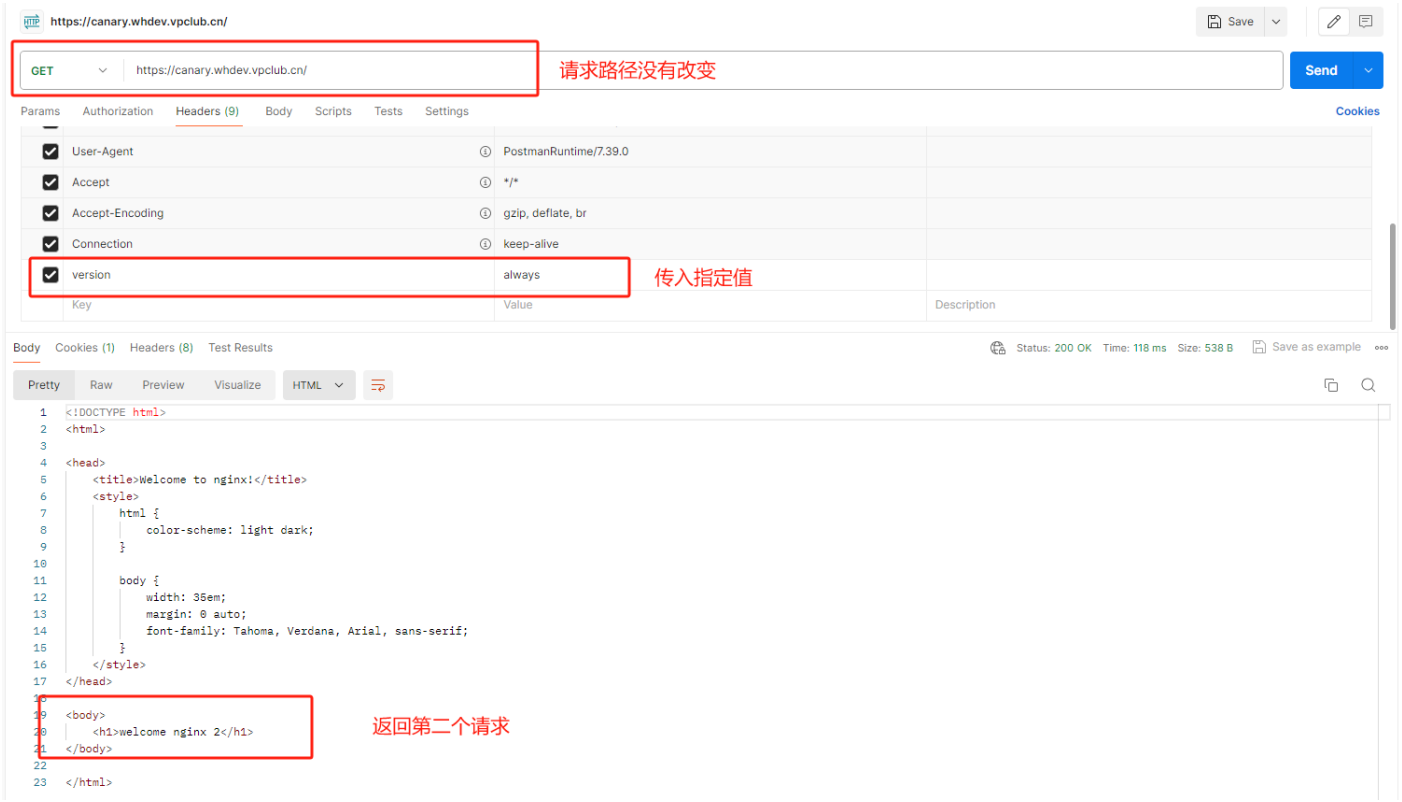
- □□ingress□□



- postman□□□□,□□□□□□

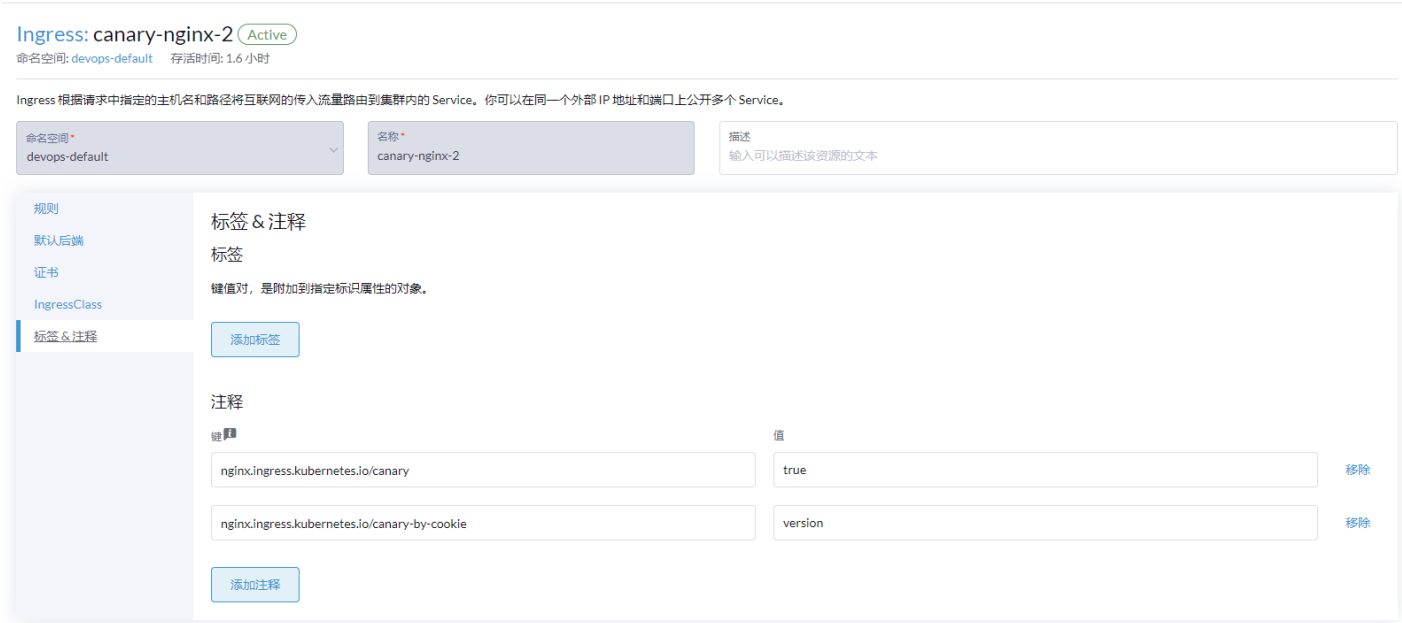


- postmain□□□□□□□□□□□□



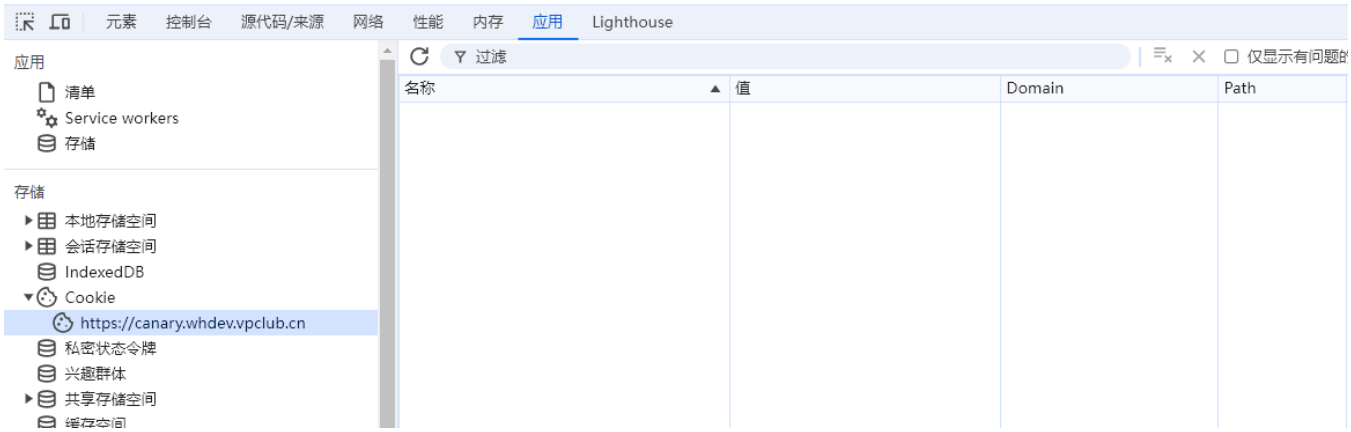
□□□□□□□cookie□□□□□□

- □□ingress□□



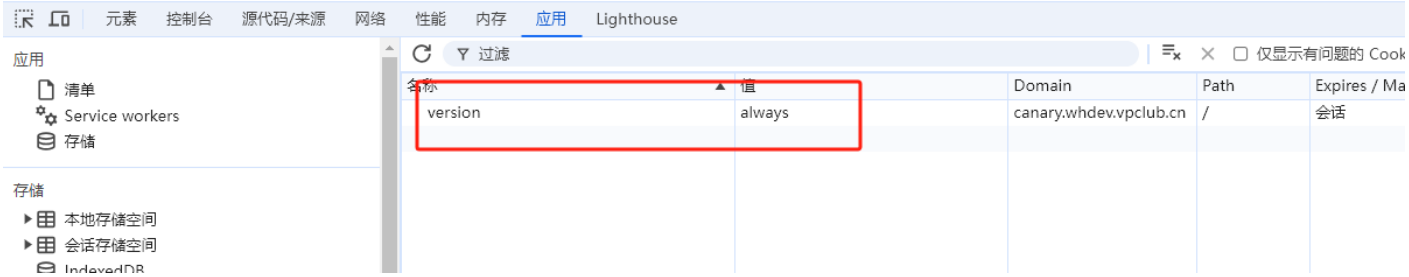
- □□□□□□□□□□cookie,□□□□□





- [redacted]





[redacted]

[redacted]

[redacted]nginx[redacted]

[redacted]nginx-ingress[redacted]nginx[redacted]

- □□□□□□□□□

```
upstream backend_by_weight {
    server  v1.example.com weight=80;
    server  v2.example.com weight=20;
}

server {
    listen 80;
    location / {
        proxy_pass http://backend_by_weight;
    }
}
```

- □□cookie□□□□

```
map $http_cookie $backend_by_cooke {
    default             v1.example.com;
    "~*version=2"       v2.example.com;
}

server {
    listen 80;
    location / {
        proxy_pass http://$backend_by_cooke;
    }
}
```

- □□headerd□□□

```
map $http_version $backend_by_hearder {
    default             v1.example.com;
    "2"                 v2.example.com;
}

server {
```

```
    listen 80;
    location / {
        proxy_pass http://$backend_by_hearder;
    }
}
```

□□□ #10
□ □□□ □□□ 11 □□ 2024 17:21:48
□ □□□ □□□ 30 □□ 2025 04:34:55