1.1 Label 与 Label Selector

1.1.1 标签

步骤 1　创建目录用于保存本次实验文件

```
[root@k8s-master labfile]# mkdir /labfile/labelfile
[root@k8s-master labfile]# cd /labfile/labelfile
```

步骤 2　创建 labelpod.yaml 并使用多个标签

```
[root@k8s-master labelfile]# vim labelpod.yaml
```

```
kind: Pod
apiVersion: v1
metadata:
  name: labelpod
  labels:
    app: busybox
    version: new
spec:
  containers:
    - name: labelpod
      image: busybox
      args:
      - /bin/sh
      - -c
      - sleep 30000
```

步骤 3　创建 Pod

```
[root@k8s-master labelfile]# kubectl apply -f labelpod.yaml
```

步骤 4　查看 pod 的 label

```
[root@k8s-master labelfile]# kubectl get pod --show-labels
```

```
NAME       READY   STATUS    RESTARTS   AGE     LABELS
labelpod   1/1     Running   0          2m19s   app=busybox,version=new
```

步骤 5　为 labelpod 添加一个新的 label

```
[root@k8s-master labelfile]# kubectl label pod labelpod time=2019
```

再次查看 pod 的 label

```
[root@k8s-master labelfile]# kubectl get pod --show-labels
```

```
NAME       READY   STATUS    RESTARTS   AGE   LABELS
labelpod   1/1     Running   0          24m   app=busybox,time=2019,version=new
```

1.1.2 标签选择器

步骤 1 创建 labelpod2.yaml

```
[root@k8s-master labelfile]# vim labelpod2.yaml
kind: Pod
apiVersion: v1
metadata:
  name: labelpod2
  labels:
    app: httpd
    version: new
spec:
  containers:
    - name: httpd
      image: httpd
```

步骤 2 创建并查看新创建的 labelpod2

```
[root@k8s-master labelfile]# kubectl apply -f labelpod2.yaml
[root@k8s-master labelfile]# kubectl get pod --show-labels
NAME        READY   STATUS    RESTARTS   AGE   LABELS
labelpod    1/1     Running   0          31m   app=busybox,time=2019,version=new
labelpod2   1/1     Running   0          19s   app=httpd,version=new
```

步骤 3 使用基于等值的标签选择器

```
[root@k8s-master labelfile]# kubectl get pod -l app=httpd 或者
[root@k8s-master labelfile]# kubectl get pod -l app==httpd
NAME        READY   STATUS    RESTARTS   AGE
labelpod2   1/1     Running   0          5m5s
```

步骤 4 使用基于不等值的标签选择器

```
[root@k8s-master labelfile]# kubectl get pod -l app!=httpd
NAME       READY   STATUS    RESTARTS   AGE
labelpod   1/1     Running   0          74m
```

步骤 5 查看各 pod 针对某标签键的值

```
[root@k8s-master labelfile]# kubectl get pod -L app
```

```
NAME        READY   STATUS    RESTARTS   AGE   APP
labelpod    1/1     Running   0          95m   busybox
labelpod2   1/1     Running   0          64m   httpd
```

1.1.3 使用标签选择器实现调度

步骤 1 使用 nodeselector 实现调度，创建 yaml 文件

```
[root@k8s-master labelfile]# vim nsdeploy.yaml
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.7.9
        ports:
        - containerPort: 80
      nodeSelector:
        env: test
```

步骤 2 创建 deployment

```
[root@k8s-master labelfile]# kubectl apply -f nsdeploy.yaml
```

```
deployment.apps/nginx-deployment created
```

步骤 3 查看 deployment 状态

```
[root@k8s-master labelfiles]# kubectl get deployment
```

```
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   0/3     3            0           16s
```

步骤 4　查看 pod 状态

```
[root@k8s-master labelfiles]# kubectl get pod
NAME                                READY   STATUS    RESTARTS   AGE
labelpod                            1/1     Running   0          70m
labelpod2                           1/1     Running   0          53m
nginx-deployment-6f56f4648c-2r742   0/1     Pending   0          27s
nginx-deployment-6f56f4648c-4zbmf   0/1     Pending   0          27s
nginx-deployment-6f56f4648c-cfk5w   0/1     Pending   0          27s
```

步骤 5　查看 pod 详细信息

```
[root@k8s-master labelfiles]# kubectl describe pod nginx-deployment-6f56f4648c-2r742
Events:
Type     Reason            Age   From               Message
----     ------            ----  ----               -------
Warning  FailedScheduling  39s   default-scheduler  0/1 nodes are available: 1 node(s) didn't match node selector.
```

步骤 6　将节点打上标签

```
[root@k8s-master labelfile]# kubectl label node k8s-master env=test
node/k8s-master labeled
```

步骤 7　查看节点上的标签

```
[root@k8s-master labelfiles]# kubectl get node --show-labels
NAME         STATUS   ROLES    AGE    VERSION   LABELS
k8s-master   Ready    master   2d2h   v1.14.1   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,env=test,kubernetes.io/arch=amd64,kubernetes.io/s-master,kubernetes.io/os=linux,node-role.kubernetes.io/master=
```

步骤 8　查看 deployment 的状态，可以看到状态变为正常。

```
[root@k8s-master labelfiles]# kubectl get deployments.
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   3/3     3            3           3m15s
```

步骤 9　使用 node affinity 调度，创建一个新的 yaml 文件 nadeploy2.yaml

```
[root@k8s-master labelfile]# vim nadeploy2.yaml
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: httpd-deployment
  labels:
    app: httpd
spec:
  replicas: 3
  selector:
    matchLabels:
      app: httpd
  template:
    metadata:
      labels:
        app: httpd
    spec:
      containers:
      - name: httpd
        image: httpd
      affinity:
        nodeAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            nodeSelectorTerms:
            - matchExpressions:
              - key: env
                operator: In
                values:
                - na
                - nb
```

步骤 10　创建 deployment

```
[root@k8s-master labelfile]# kubectl apply -f nadeploy2.yaml
```

步骤 11　查看 deployment 状态，pod 没有成功部署

```
[root@k8s-master labelfiles]# kubectl get deployment
NAME              READY   UP-TO-DATE   AVAILABLE   AGE
httpd-deployment  0/3     3            0           16s
```

步骤 12　修改 yaml 文件，在集合中加入 env=test

```
[root@k8s-master labelfile]# vim nadeploy2.yaml
```

```
……
```

```
    affinity:
      nodeAffinity:
        requiredDuringSchedulingIgnoredDuringExecution:
          nodeSelectorTerms:
          - matchExpressions:
            - key: env
              operator: In
              values:
              - na
              - nb
              - test
```

步骤 13 重新部署 deployment，并且查看 deployment 的状态

```
[root@k8s-master labelfiles]# kubectl apply -f nadeploy2.yaml
[root@k8s-master labelfiles]# kubectl get deployments.
NAME              READY   UP-TO-DATE   AVAILABLE   AGE
httpd-deployment   3/3     3            3           105s
```