

## ConfigMap

## 直接手写配置创建

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: configmap
  namespace: dev
data:
  application.yaml: |
    username: admin
    address: 123456
```

[illegible]

接下来创建 pod-configmap.yaml，将上面的 configmap 挂载进去

```
apiVersion: v1
kind: Pod
metadata:
  name: pod-configmap
  namespace: dev
spec:
  containers:
  - name: nginx
    image: nginx:1.17.1
    # 将configmap挂载到目录
    volumeMounts:
    # name 与下面的 volumes 一致
    - name: config
      # 挂载路径
      mountPath: /configmap/config
  volumes: # 引用configmap
  - name: config
    configMap:
      name: configmap
```

• • • • •

```
# 创建 pod[root@k8s-master2 test]# kubectl apply -f pod-configmap.yaml pod/pod-configmap created# 查看 pod[root@k8s-master2 test]# kubectl get pod pod-configmap -n devNAME
```

## 通过文件创建

上面手工配置适用于少量、单一方式的配置文件，若文件众多、文件冗长，可以从文件/目录下的文件创建 configmap,同时挂载到容器中保持原来的目录结构。

如下的目录结构：

```
[root@k8s-master2 cmtest]# tree subdir/
subdir/
├── test1.yaml
└── test2.yaml

0 directories, 2 files
```

创建configmap

```
# 从文件创建[root@k8s-master2 cmtest]# kubectl create configmap file-configmap --from-file=subdir -n devconfigmap/file-configmap created# 查看 configmap 明细[root@k8s-master2 cmtest]
Data====test1.yaml:----aldfjkjlsajdfilsalsjf
test2.yaml:---kldfjlasjflksjk
Events:  <none>
```

接下来创建 pod-file-configmap.yaml，将上面的 configmap 挂载进去

```
apiVersion: v1
kind: Pod
metadata:
  name: pod-file-configmap
  namespace: dev
spec:
  containers:
    - name: nginx
      image: nginx:1.17.1
      # 将configmap挂载到目录
      volumeMounts:
        # name 与下面的 volumes 一致
        - name: config
          # 挂载路径
          mountPath: /configmap/config
  volumes: # 引用configmap
    - name: config
      configMap:
        name: file-configmap
```

```
# 创建 pod[root@k8s-master2 test]# kubectl apply -f pod-file-configmap.yaml pod/pod-file-configmap created# 查看pod[root@k8s-master2 test]# kubectl get pod pod-file-configmap -n de
```



Secret 是和 Configmap 比较相似的一类对象，它主要用于存储敏感信息，例如密码、身份证号、密钥、证书等等

- 首先使用 base64 对数据进行编码

○ ○ ○ ○ ○

```
# 准备 username [root@k8s-master2 test]# echo -n 'admin' | base64 YWRtaW4= # 准备 password [root@k8s-master2 test]# echo -n '123456' | base64 MTIzNDU2
```

- 然后, 编写 `secret.yaml`, 创建 `secret`

```
apiVersion: v1
kind: Secret
metadata:
  name: secret
  namespace: dev
type: Opaque
data:
  username: YWRtaW4=
  password: MTIzNDU2
```

[illegible]

```
# 创建 secret [root@k8s-master2 test]# kubectl apply -f secret.yaml secret/secret created # 查看 secret [root@k8s-master2 test]# kubectl describe secret -n dev Name:
Type: kubernetes.io/service-account-token
Data ==== ca.crt: 1070 bytes namespace: 3 bytes token: eyJhbGciOiJSUzI1NiIsImtpZCI6Im1sdmxcQy1lNnM3NHFucllWbWZ3YVBA1Rqd2kTO9zN3hLekVNWjR5bW5ifQ.eyJpc3MiOiJrdD
Name: secret Namespace: dev Labels: <none> Annotations: Type: Opaque
Data ==== username: 5 bytes password: 6 bytes
```

- 创建 pod-secret.yaml,将上面创建的 secret 挂载进去:

```
apiVersion: v1
kind: Pod
metadata:
  name: pod-secret
namespace: dev
spec:
  containers:
  - name: nginx
    image: nginx:1.17.1
    # 将secret挂载到目录
    volumeMounts:
    - name: 与下方定义的一致
      - name: config
    mountPath: /secret/config
  volumes:
  - name: config
    secret:
      secretName: secret
```

欢迎关注我的公众号“云原生拓展”，原创技术文章第一时间推送。