

Config maps & secrets

Набор конфигураций для различных окружений. Включают переменные окружения, конфиг файлы, имена хостов, порты, аккаунты.

```
kind: ConfigMap
apiVersion: v1
metadata:
  name: epl
data:
  Competition: epl
  Season: 2022-2023
  Champions: Manchester City
  test.conf: |
    env = plex-test
    endpoint = 0.0.0.0:31001
    char = utf8
    vault = PLEX/test
    log-size = 512M
```

Их можно использовать как переменные окружения, параметры запуска и файлы внутри контейнера.

Использование ConfigMap

В виде переменных окружения:

```
apiVersion: v1
kind: Pod
<Snip>
spec:
  containers:
    - name: ctr1
      env:
        - name: FIRSTNAME
          valueFrom:
            configMapKeyRef: <===== a ConfigMap
              name: multimap <===== called "multimap"
```

key: given

Не изменяются после создания.

В виде аргументов:

Сначала как переменные окружения, затем в команду создания

```
spec:
  containers:
  - name: args1
    image: busybox
    env:
    - name: FIRSTNAME <<===== Environment variable called FIRSTNAME
      valueFrom: <<===== based on
        configMapKeyRef: <<===== a ConfigMap
          name: multimap <<===== called "multimap"
          key: given <<===== and populated by the value in the "given" field
    - name: LASTNAME <<===== Environment variable called LASTNAME
      valueFrom: <<===== based on
        configMapKeyRef: <<===== a ConfigMap
          name: multimap <<===== called "multimap"
          key: family <<===== and populated by the value in the "family" field
    command: [ "/bin/sh", "-c", "echo First name ${FIRSTNAME} last name ${LASTNAME}" ]
```

В виде файлов:

Создается том и привязывается к нему ConfigMap

```
apiVersion: v1
kind: Pod
metadata:
  name: cmvol
spec:
  volumes:
  - name: volmap <<===== Create a volume called "volmap"
    configMap: <<===== based on the ConfigMap
      name: multimap <<===== called "multimap"
  containers:
  - name: ctr
    image: nginx
```

```
volumeMounts: <<==== These lines mount the
- name: volmap <<==== the "volmap" volume into the
  mountPath: /etc/name <<==== container at "/etc/name"
```

Файлы будут созданы в соответствии именам в ConfigMap. При обновлении ConfigMap файлы в контейнере обновятся.

Secrets

Хранятся в Base64 но внутри контейнера хранятся в tmpfs виде обычного текста.

```
apiVersion: v1
kind: Secret
metadata:
  name: tkb-secret
  labels:
    chapter: configmaps
type: Opaque
data: <<==== Change to "stringData" for plain text
  username: bmlnZWxwb3VsdG9u
  password: UGFzc3dvcmQxMjM=
```

Использование в Pod:

```
apiVersion: v1
kind: Pod
metadata:
  name: secret-pod
  labels:
    topic: secrets
spec:
  volumes:
    - name: secret-vol <<==== Volume name
      secret: <<==== Volume type
        secretName: tkb-secret <<==== Populate volume with this Secret
  containers:
    - name: secret-ctr
      image: nginx
      volumeMounts:
        - name: secret-vol <<==== Mount the volume defined above
          mountPath: "/etc/tkb" <<==== into this path
```

В директории /etc/tkb будут храниться файлы, по одному на каждое значение.

Основные команды

Команда	Доп. пар.	Описание
kubectl get cm		Список ConfigMap

Revision #4
Created 28 March 2025 00:30:32 by Admin
Updated 28 March 2025 04:49:34 by Admin