How do you manage
sensitive data in Kubernetes?

srinathchalla@outlook.com

Secrets

Concept

Udemy

Objectives

Concept

- Overview of Secrets

Review Demo

- Creating Secrets – using kubectl and Manually
- Decoding Secrets
- Consuming Secrets as Volumes and Env Variables

srinathchalla@outlook.com

Secrets

Kubernetes object to handle

small amount of sensitive data

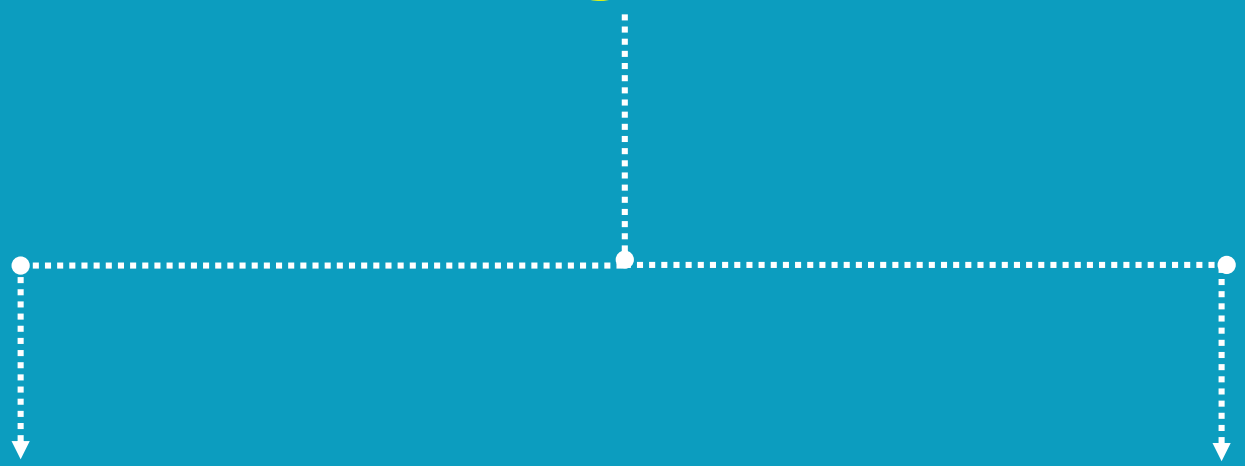
srinathchalla@outlook.com

Overview

- Small amount of sensitive data
  - Passwords, Tokens, or Keys
- Reduces risk of exposing sensitive data
- Created outside of Pods
- Stored inside ETCD database on Kubernetes Master
- Not more than 1MB
- Used in two ways- Volumes or Env variables
- Sent only to the target nodes

srinathchalla@outlook.com

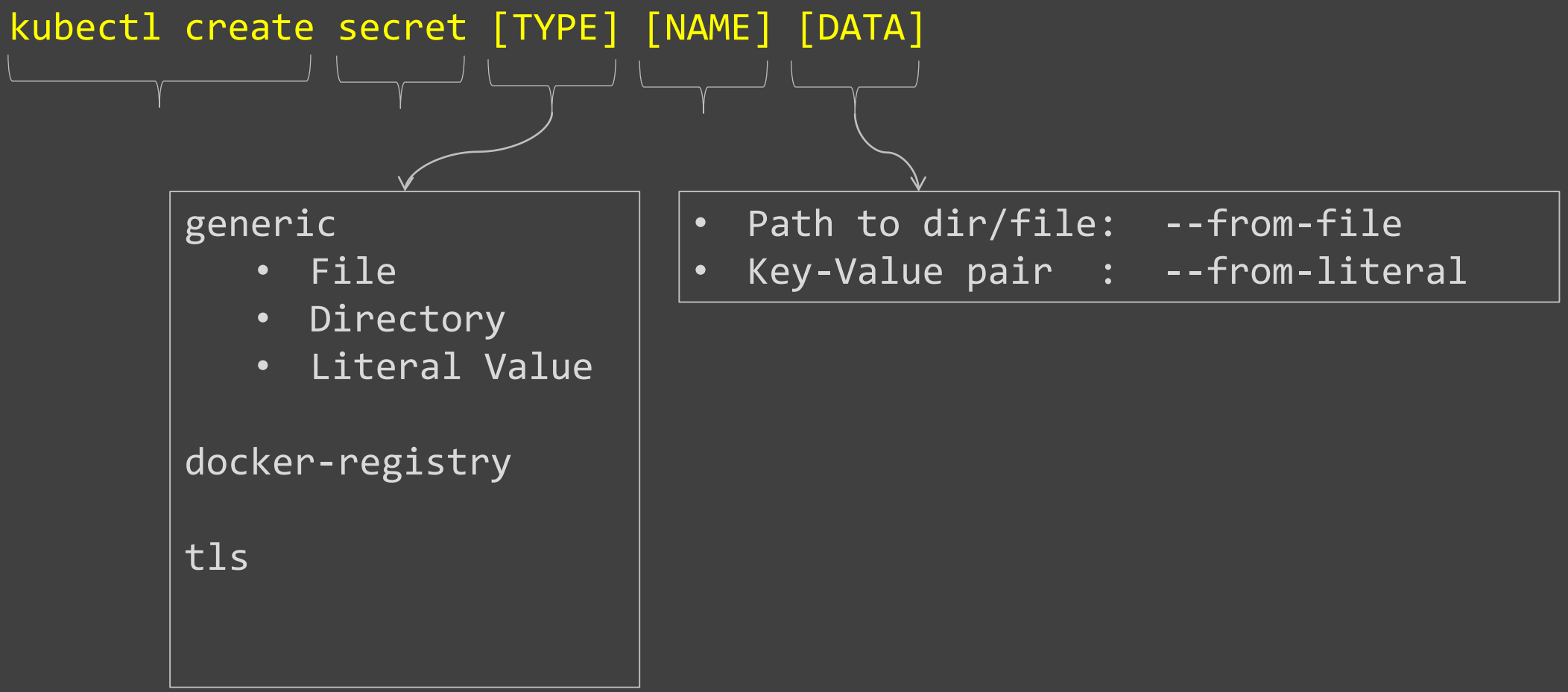
Creating Secrets



kubectl

Manually

Using Kubectl: Syntax



Creating Secret: Kubectl

```
srinath@master:$ echo -n 'admin' > ./username.txt
srinath@master:$ echo -n '1f2d1e2e67df' > ./password.txt
```

```
srinath@master:$ kubectl create secret generic db-user-pass --from-file=./username.txt --
                 from-file=./password.txt
secret "db-user-pass" created
```

```
srinath@master:$ kubectl get secrets
NAME                  TYPE                                  DATA      AGE
db-user-pass          Opaque                                2         51s
```

```
srinath@master:$ kubectl describe secrets db-user-pass
Name:           db-user-pass
Namespace:      default
Labels:         <none>
Annotations:    <none>

Type:           Opaque

Data
====
password.txt:   12 bytes
username.txt:   5 bytes
```

srinathchalla@outlook.com

Creating Secret: Manually

```
srinath@master:$ echo -n 'admin' | base64
YWRtaW4=
srinath@master:$ echo -n '1f2d1e2e67df' | base64
MWYyZDFlMmU2N2Rm
```

```
# mysecret.yaml
apiVersion: v1
kind: Secret
metadata:
  name: mysecret
type: Opaque
data:
  username: YWRtaW4=
  password: MWYyZDFlMmU2N2Rm
```

```
srinath@master:$ kubectl create -f mysecret.yaml
secret/mysecret created
```

srinathchalla@outlook.com

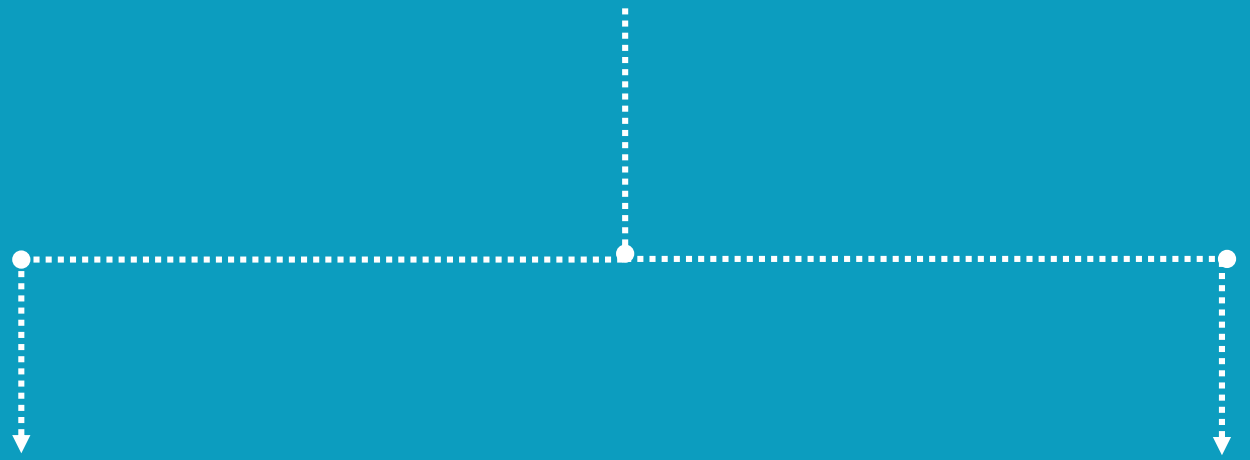
Decoding Secrets

```
srinath@master:$ kubectl get secrets mysecret -o yaml
apiVersion: v1
data:
  password: MWYyZDFlMmU2N2Rm
  username: YWRtaW4=
kind: Secret
metadata:
  creationTimestamp: 2018-09-01T12:46:17Z
  name: mysecret
  namespace: default
  resourceVersion: "616565"
  selfLink:
/api/v1/namespaces/default/secrets/mysecret
  uid: 051e61ae-ade5-11e8-8d64-42010a800003
type: Opaque
```

```
srinath@master:$ echo 'YWRtaW4=' | base64 --decode
admin
srinath@master:$ echo 'MWYyZDFlMmU2N2Rm' | base64 --decode
1f2d1e2e67df
```

srinathchalla@outlook.com

Consuming Secrets in Pods



Volumes

Environmental Variables

srinathchalla@outlook.com

Manually

```
srinath@master:$ echo -n 'admin' | base64
YWRtaW4=
srinath@master:$ echo -n '1f2d1e2e67df' | base64
MWYyZDFlMmU2N2Rm
```

```
# mysecret.yaml
apiVersion: v1
kind: Secret
metadata:
  name: mysecret
type: Opaque
data:
  username: YWRtaW4=
  password: MWYyZDFlMmU2N2Rm
```

```
srinath@master:$ kubectl create -f mysecret.yaml
secret/mysecret created
```

srinathchalla@outlook.com

Consuming “Secrets” from volume

```yaml
# mysecret-pod.yaml
apiVersion: v1
kind: Pod
metadata:
  name: mypod
spec:
  containers:
  - name: mypod
    image: redis
    volumeMounts:
    - name: foo
      mountPath: "/etc/foo"
      readOnly: true
  volumes:
  - name: foo
    secret:
      secretName: mysecret
```

```
srinath@master:$ kubectl create -f mysecret-pod.yaml
secret/mysecret-pod created


srinath@master:$ kubectl get po
NAME      READY     STATUS              RESTARTS   AGE
mypod     1/1       Running             0          22m


srinath@master:$ kubectl exec mypod ls /etc/foo
password
username

srinath@master:$ kubectl exec mypod cat /etc/foo/passwd
1f2d1e2e67df

srinath@master:$ kubectl exec mypod cat /etc/foo/username
admin
```

srinathchalla@outlook.com

Consuming “Secrets” from “Environment Variables”

```yaml
# mysecret-env-pod.yaml
apiVersion: v1
kind: Pod
metadata:
  name: secret-env-pod
spec:
  containers:
  - name: mycontainer
    image: redis
    env:
      - name: SECRET_USERNAME
        valueFrom:
          secretKeyRef:
            name: mysecret
            key: username

      - name: SECRET_PASSWORD
        valueFrom:
          secretKeyRef:
            name: mysecret
            key: password
  restartPolicy: Never
```

```
srinath@master:$ kubectl create -f mysecret-pod-env.yaml
secret/mysecret-pod-env created

srinath@master:$ kubectl get po
NAME             READY     STATUS    RESTARTS   AGE
secret-env-pod   1/1       Running   0          7s

srinath@master:$ kubectl exec secret-env-pod env | grep SECRET
SECRET_PASSWORD=1f2d1e2e67df
SECRET_USERNAME=admin
```

srinathchalla@outlook.com

Summary

Concept

Overview of Secrets

Review Demo

Creating Secrets – using kubectl and Manually

Decoding Secrets

Consuming Secrets as Volumes and Env Variables

srinathchalla@outlook.com

Coming up...

Demo
Secrets

srinathchalla@outlook.com