KCD
GROWING CLOUD NATIVE TOGETHER
HYDERABAD

# CEL-ebrating Simplicity:

## CRD Validation in Kubernetes Beyond Webhooks

@chiragkyal

# Chirag Kyal

- Software Engineer Red Hat
- OPENSHIFT Red Hat Edge kubernetes
- Member of Bengaluru Technical Community, Red Hat
- Ex- IBM Cloud AiKno

@chiragkyal

# CEL-ebrating Simplicity:

## CRD Validation in Kubernetes Beyond Webhooks

@chiragkyal

# CEL-ebrating Simplicity

# CEL-ebrating

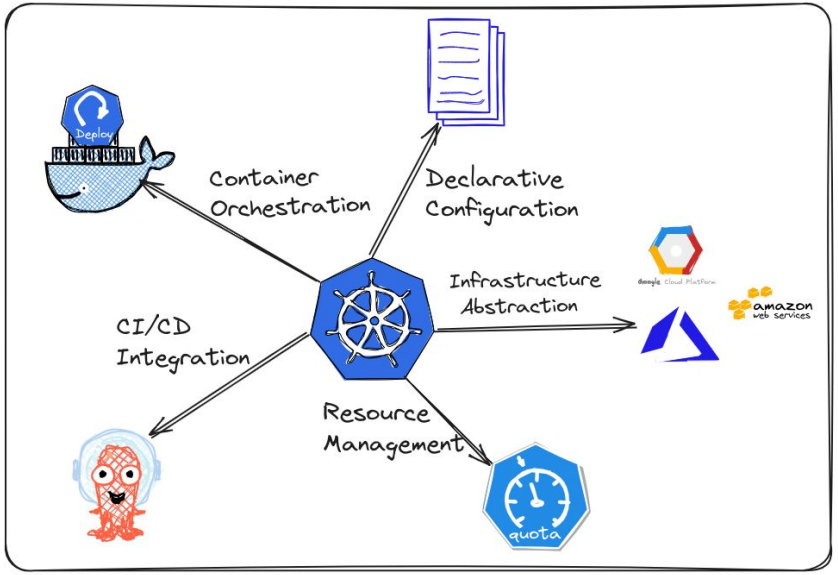

#KuberTENes.

# CEL-ebrating Simplicity

## CEL-ebrating


#KuberTENes.

## Simplicity



# CEL-ebrating Simplicity

```yaml
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
 name: books.mylibrary.com
spec:
 group: mylibrary.com
  ......
```

Extending the Kubernetes API with

## CustomResourceDefinitions (CRDs)

- Allows you to create your own custom resources.
- Operator with custom business logic.

# CRD Validation

## WHY ?

- Field Verification
  - ***port*** is integer, within a range
- Field is Valid compared to other field
  - ensure that the ***startDate*** is before the ***endDate***
- Field is Immutable
  - ***username*** cannot be changed, once defined
- Enhancing Compatibility, Stability and Security

# CRD Validation

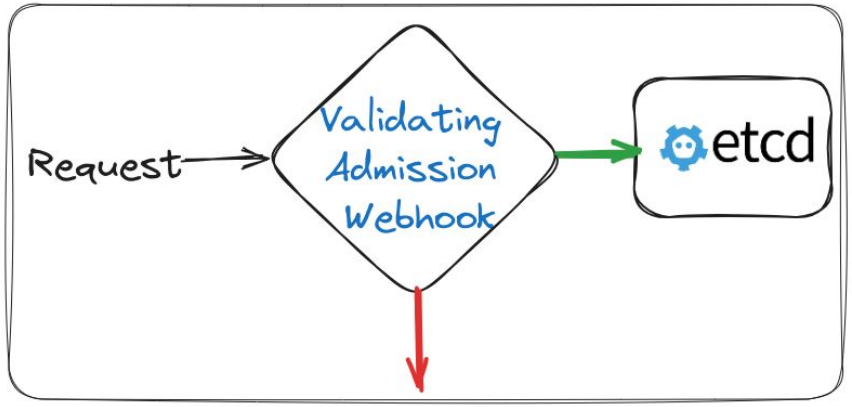
## WHY ?

- ➔ Field Verification
  - ◆ ***port*** is integer, within a range
- ➔ Field is Valid compared to other field
  - ◆ ensure that the ***startDate*** is before the ***endDate***
- ➔ Field is Immutable
  - ◆ ***username*** cannot be changed, once defined
- ➔ Enhancing Compatibility, Stability and Security

**If not, all the best! :D**

# CRD Validation

## HOW ?



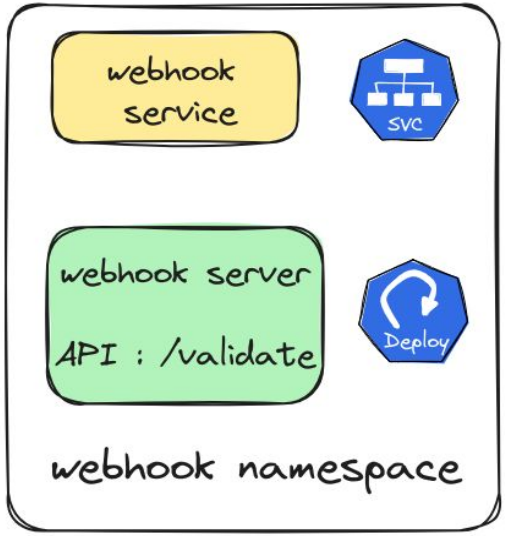

[Beyond OpenAPI schema validation]

Works by **intercepting** requests to API Server

# Validating Admission Webhooks are simple ?

# Validating Admission Webhooks are simple ?

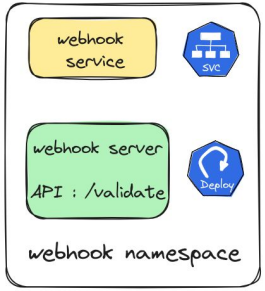
1. Create a HTTPs backend server
   - Implement the Admission Review Handler
   - Define Validation Logic
   - Return Validation Response
   - Define CMD flags
2. Create the Deployment Configuration
   - Build and Deploy your Admission Controller code
   - Expose the Admission Controller's endpoint service
   - Necessary RBAC



# Validating Admission Webhooks are simple ?

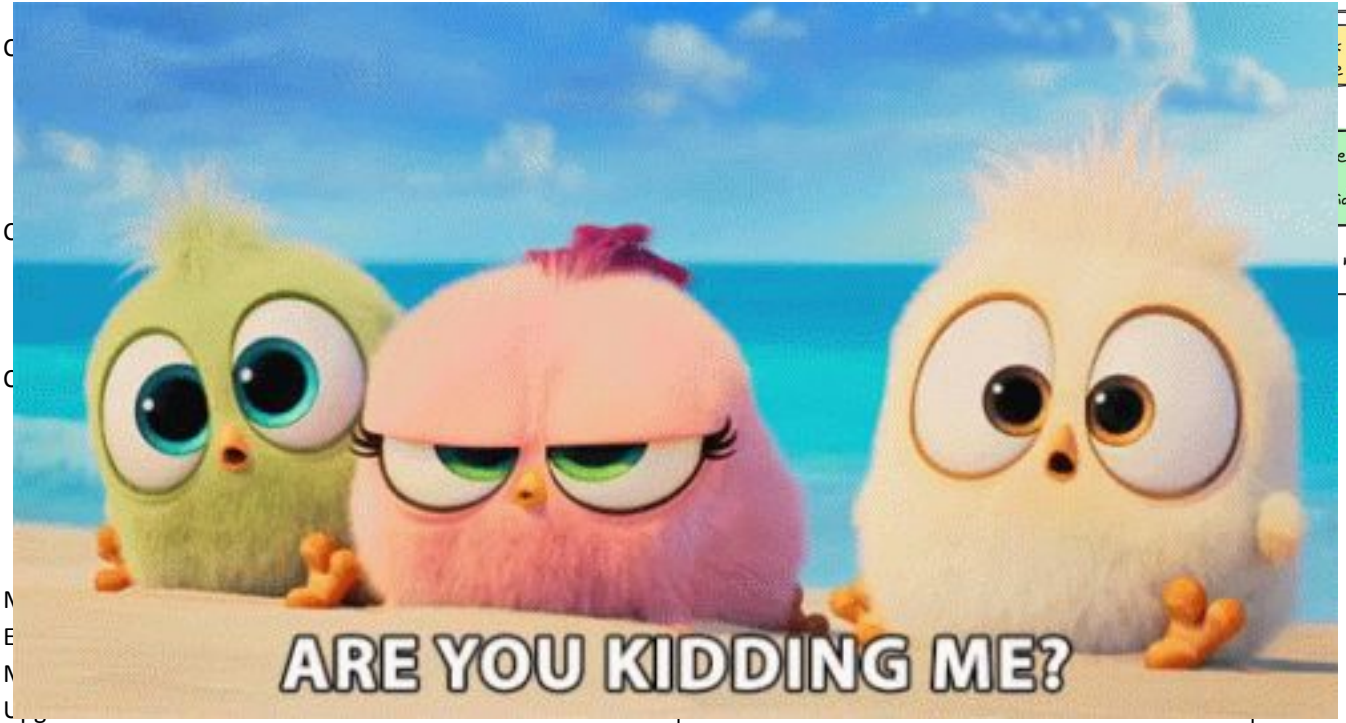
1. Create a HTTPs backend server
   - Implement the Admission Review Handler
   - Define Validation Logic
   - Return Validation Response
   - Define CMD flags
2. Create the Deployment Configuration
   - Build and Deploy your Admission Controller code
   - Expose the Admission Controller's endpoint service
   - Necessary RBAC
3. Create ***ValidatingWebhookConfiguration***
   - Rules
   - Service reference
   - caBundle
   - Failure policy
   - Timeouts
4. Manage and maintain the certificates (server <> client trust)
5. Ensure service network works all the time with api server
6. Monitoring and Alerting
7. Upgrade and Rollback mechanism
8. Runbooks for all possible problems

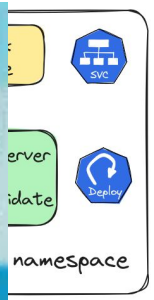
```yaml
apiVersion: admissionregistration.k8s.io/v1
kind: ValidatingWebhookConfiguration
metadata:
 name: "my-webhook.example.com"
webhooks:
- name: "my-webhook.example.com"
 rules:
 - apiGroups:   [""]
   apiVersions: ["v1"]
   operations:  ["CREATE"]
   resources:   ["pods"]
   scope:       "Namespaced"
 clientConfig:
   service:
     namespace: "webhook-namespace"
     name: "webhook-service"
     path: "/validate"
     port: 8443
   caBundle: <CA_BUNDLE>
 failurePolicy: Fail
 timeoutSeconds: 5
```



# Validating Admission Webhooks are simple ?

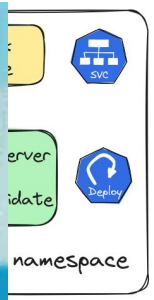
1. C
2. C
3. C
4. M
5. E
6. M
7. U
8. Runbooks for all possible problems



# Validating Admission Webhooks are simple ?

1. C
2. C
3. C
4. M
5. E
6. M
7. U
8. Runbooks for all possible problems

# How to make it Simple?

# CEL-ebrating Simplicity

# CEL-ebrating Simplicity

**Common Expression Language (CEL)**

# Common Expression Language (CEL)

Example: Validate that the three fields defining replicas are within a self-defined range

CEL

```
self.minReplicas <= self.replicas
&&
self.replicas <= self.maxReplicas
```

```
minReplicas: 2
replicas: 6
maxReplicas: 5
```

```
minReplicas: 2
replicas: 3
maxReplicas: 5
```

# Common Expression Language (CEL)

Example: Validate that the three fields defining replicas are within a self-defined range

CEL

```
self.minReplicas <= self.replicas
&&
self.replicas <= self.maxReplicas
```

```
minReplicas: 2
replicas: 6
maxReplicas: 5
```

```
minReplicas: 2
replicas: 3
maxReplicas: 5
```

Example: Key must exist in a map

CEL

```
'Available' in self.stateCounts
```

```
stateCounts
   Available: 2
   Degraded: 1
```

```
stateCounts
   Degraded: 1
```

# Common Expression Language (CEL)

Example: Validate that the three fields defining replicas are within a self-defined range

CEL

```
self.minReplicas <= self.replicas
&&
self.replicas <= self.maxReplicas
```

```
minReplicas: 2
replicas: 6
maxReplicas: 5
```

```
minReplicas: 2
replicas: 3
maxReplicas: 5
```

Example: Key must exist in a map

CEL

```
'Available' in self.stateCounts
```

```
stateCounts
   Available: 2
   Degraded: 1
```

```
stateCounts
   Degraded: 1
```

- Straightforward syntax
- Fast and Developer-friendly
- Similar to the expressions in C, C++, Java, Go

- Expression language (single), not scripting
- if/else vs **ternary operator** : *<condition> ? <ifTrue> : <ifFalse>*
- for/while vs **macros**: *has, all, exists, filter …*

# Common Expression Language (CEL)

| | |
|---|---|
| `self.health.startsWith('ok')` | Validate a 'health' string field has the prefix 'ok' |
| `self.names.isSorted()` | Verify that a list of names is kept in alphabetical order |
| `self.names.size() == self.details.size()`<br>`&& self.names.all(n, n in self.details)` | Validate the 'details' map is keyed by the items in the 'names' listSet |
| `self.set1.all(e, !(e in self.set2))` | Validate that two listSets are disjoint |

More at https://github.com/google/cel-spec

# CEL + CRD Validation

# CEL + CRD Validation

By using extension

```
x-kubernetes-validations
```

GA in Kubernetes 1.29

# CEL + CRD Validation

Directly in the OpenAPI schema

```yaml
openAPIV3Schema:
 type: object
 properties:
   spec:
     type: object
     x-kubernetes-validations:
       - rule: "self.minReplicas <= self.replicas"
         message: "replicas should be greater than or equal to minReplicas."
       - rule: "self.replicas <= self.maxReplicas"
         message: "replicas should be smaller than or equal to maxReplicas."
     properties:
       ...
       minReplicas:
         type: integer
       replicas:
         type: integer
       maxReplicas:
         type: integer
     required:
       - minReplicas
       - replicas
       - maxReplicas
```

- Multiple CEL rules are allowed
- *Rule* is scoped to the location of the *x-kubernetes-validations* extension in the schema.
- The self variable accesses the scoped value. (here self refers to spec field)
- *Message :* Human readable error (optional)

# CEL + CRD Validation

```yaml
openAPIV3Schema:
 type: object
 properties:
   spec:
     type: object
     x-kubernetes-validations:
       - rule: "self.minReplicas <= self.replicas"
         message: "replicas should be greater than or equal to minReplicas."
       - rule: "self.replicas <= self.maxReplicas"
         message: "replicas should be smaller than or equal to maxReplicas."
     properties:
       ...
       minReplicas:
         type: integer
       replicas:
         type: integer
         x-kubernetes-validation:
           - rule: "self % 2 == 0"
             message: "replicas must be an even number."
       maxReplicas:
         type: integer
     required:
       - minReplicas
       - replicas
       - maxReplicas
```

Multiple *x-kubernetes-validations* in a CRD, at different levels

Scoped to ***spec*** field

Scoped to ***replicas*** field

Encouraged to put ***as close to the field as possible***, for convenience.

https://kubernetes.io/docs/tasks/extend-kubernetes/custom-resources/custom-resource-definitions/#validation-rules

# Demo Time


```
// +kubebuilder:validation:XValidation:rule=
```

# Takeaway

- `“+kubebuilder:validation:XValidation”` marker translates to `“x-kubernetes-validations”`
- Make CRD validation rules **self-contained** and **declarative** using CEL.
- Can handle complex, cross-field and immutability checks.
- **Lightweight** and **safe** to be run *directly in the* ***kube-apiserver***.

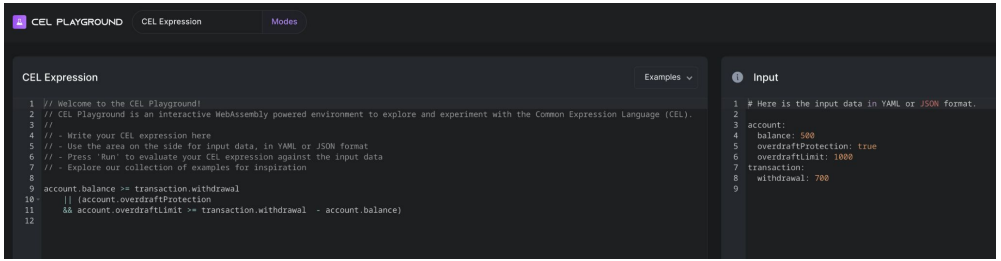
***Think beyond Webhooks with CEL***

# Takeaway

- `“+kubebuilder:validation:XValidation”` marker translates to `“x-kubernetes-validations”`
- Make CRD validation rules **self-contained** and **declarative** using CEL.
- Can handle complex, cross-field and immutability checks.
- **Lightweight** and **safe** to be run *directly in the* ***kube-apiserver***.

***Think beyond Webhooks with CEL***

# Bonus




**CEL Playground** (https://playcel.undistro.io/) for quick testing CEL expressions.

KCD HYDERABAD

# Thank You! for CEL-ebrating Simplicity


Chirag Kyal

@chiragkyal