
KubeCon
CloudNativeCon
North America 2023


KubeCon | CloudNativeCon

North America 2023

Safeguarding clusters: exploring the benefits and navigating the dangers of admission controllers

Igor Velichkovich - Senior SDE Amazon EKS
Amine Hilaly - SDE Amazon EKS

Admission Controllers

- Have you ever used Admission controllers?
- Have you ever broken an application or your cluster components using Admission controllers?

Agenda

- Quick intro to mutating/validating webhooks
- Dangers
- Demo of bad webhooks taking down a cluster
- New Common Expression Language (CEL) based features
- Demo new CEL features

Davanum Srinivas @dims · Aug 24, 2022

One default that changed in @kubernetesio v1.25 releases was where we pull images from. Please note that we now use "registry.k8s.io". This works for all releases, not just v1.25. So please change any references in your automation tools to the new endpoint.

> **Moved container registry service from k8s.gcr.io to registry.k8s.io**
>
> Moving container registry from k8s.gcr.io to registry.k8s.io got merged. For more details, see the wiki page, announcement was sent to the kubernetes development mailing list.

3 82 167

An example of Admission webhooks

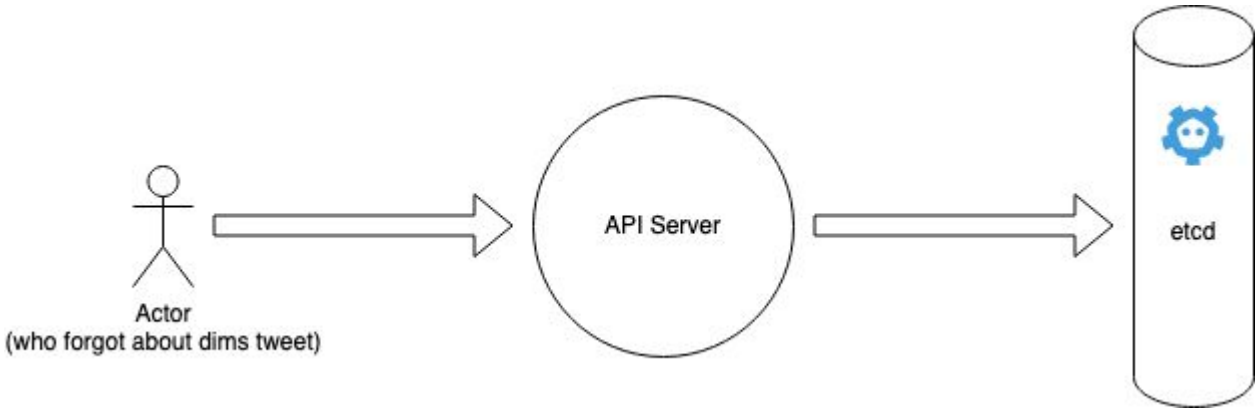
Early 2023:

- *we stopped using k8s.gcr.io*
- *And recommended using registry.k8s.io*

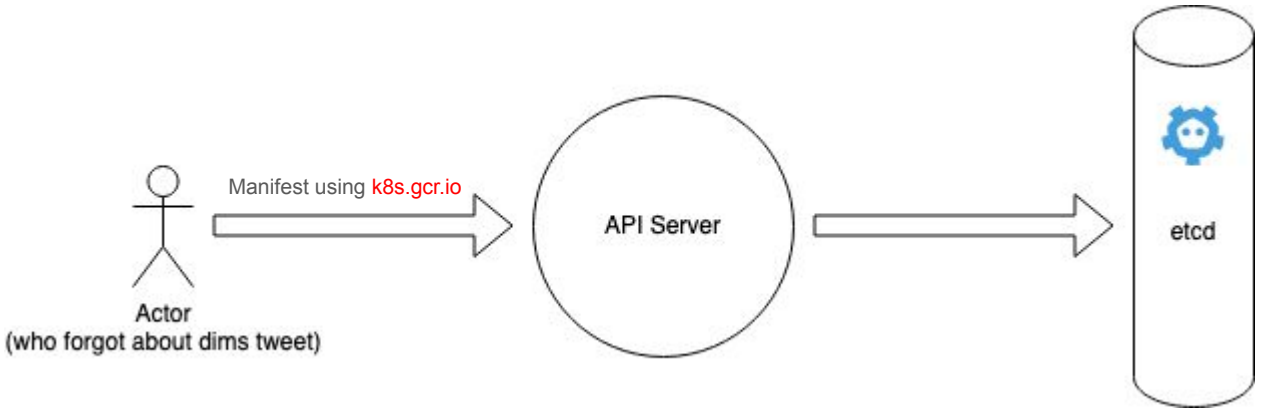
An example of Admission webhooks

As a kubernetes user to want to enforce this “policy”

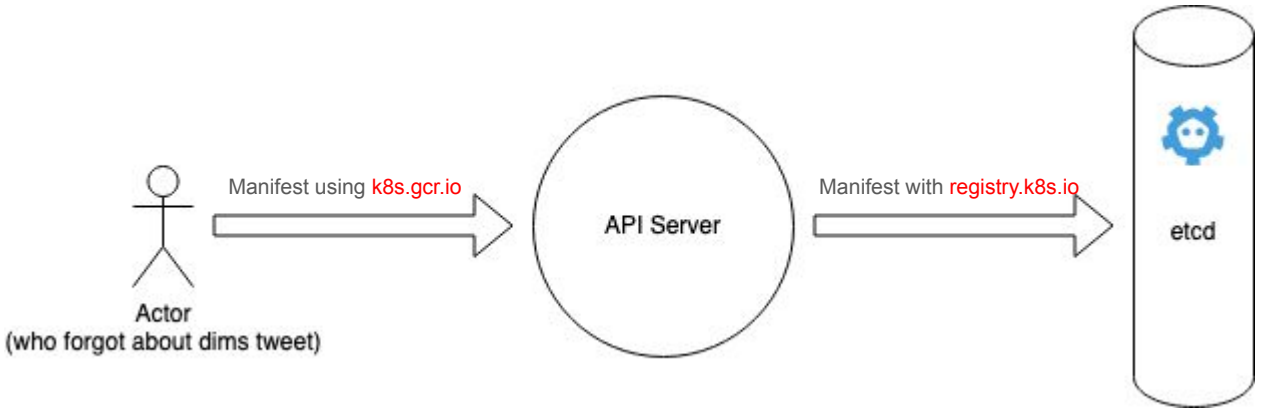
Admission Webhook Intro



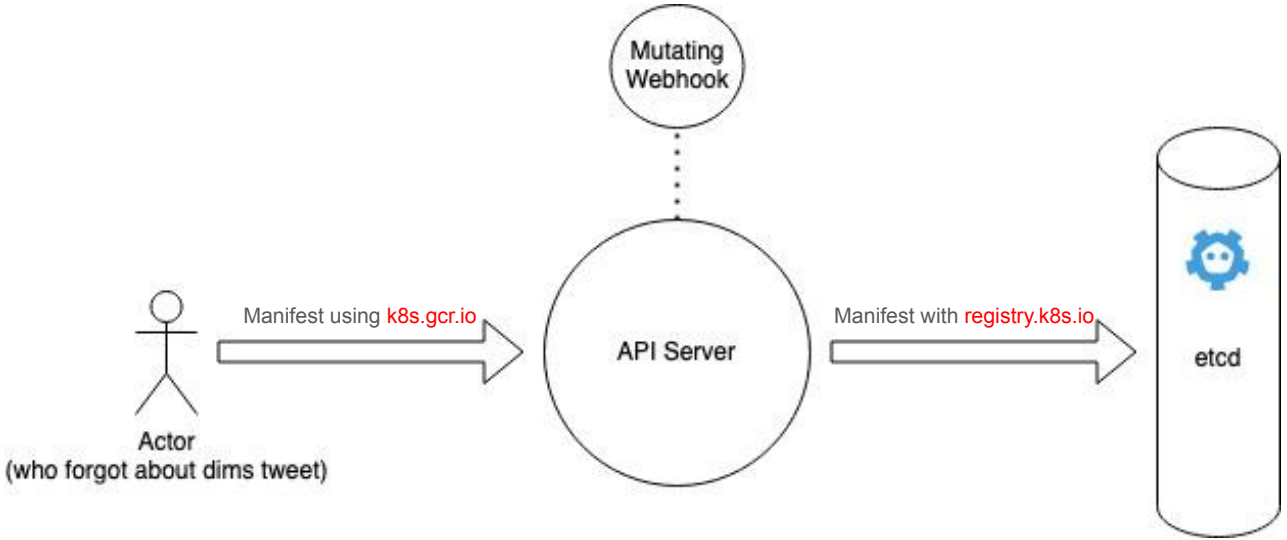
Admission Webhook Intro



Admission Webhook Intro



Admission Webhook Intro



Admission Webhook Intro

Admission Controllers Reference

This page provides an overview of Admission Controllers.

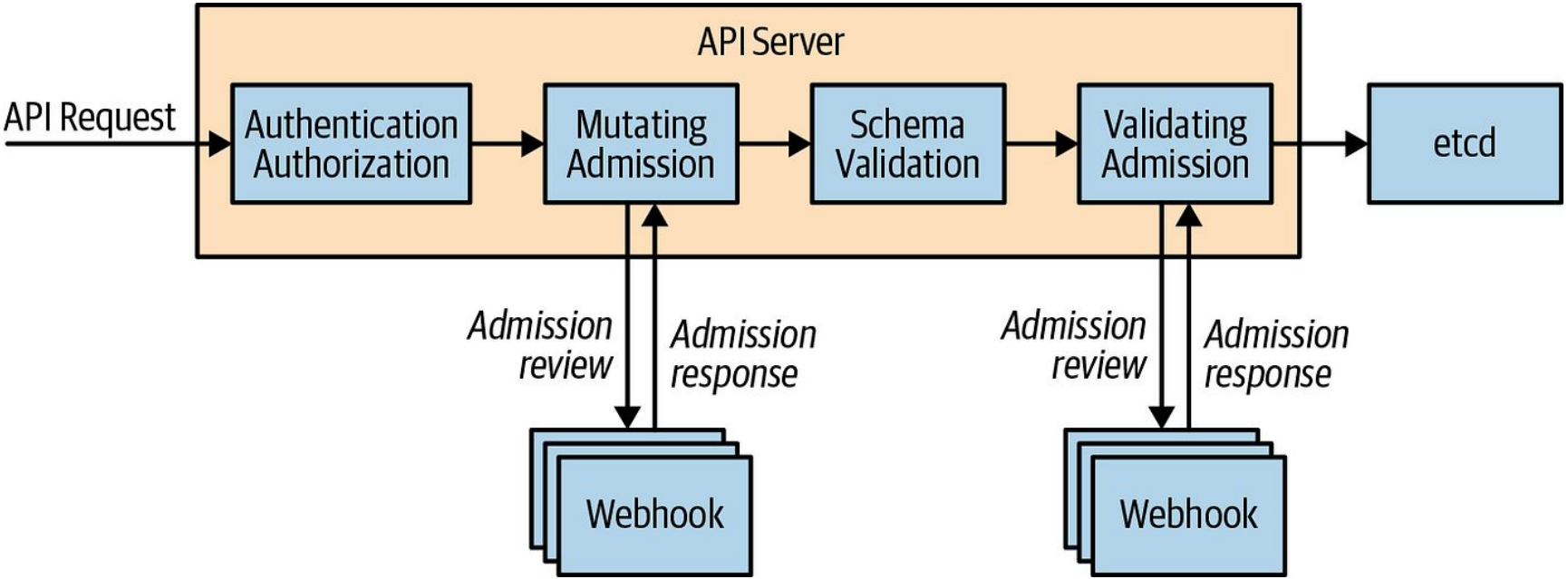
What are they?

An *admission controller* is a piece of code that intercepts requests to the Kubernetes API server prior to persistence of the object, but after the request is authenticated and authorized.

Admission controllers may be *validating*, *mutating*, or both. Mutating controllers may modify objects related to the requests they admit; validating controllers may not.

https://kubernetes.io/docs/reference/access-authn-authz/admission-controllers/

Admission Webhook Intro



Examples

Examples:

```yaml
apiVersion: admissionregistration.k8s.io/v1
kind: ValidatingWebhookConfiguration
metadata:
  name: "pod-policy.example.com"
webhooks:
- name: "pod-policy.example.com"
  rules:
  - apiGroups:   [""]
    apiVersions: ["v1"]
    operations:  ["CREATE"]
    resources:   ["pods"]
    scope:       "Namespaced"
  clientConfig:
    service:
      namespace: "example-namespace"
      name: "example-service"
    caBundle: <CA_BUNDLE>
  admissionReviewVersions: ["v1"]
  sideEffects: None
  timeoutSeconds: 5
```

ValidatingWebhookConfiguration

```yaml
apiVersion: admissionregistration.k8s.io/v1
kind: MutatingWebhookConfiguration
webhooks:
- name: my-webhook.example.com
  objectSelector:
    matchLabels:
      foo: bar
  rules:
  - operations: ["CREATE"]
    apiGroups: ["*"]
    apiVersions: ["*"]
    resources: ["*"]
    scope: "*"
```

MutatingWebhookConfiguration

Dangers of Admission Webhooks

Admission webhooks make your kube-apiserver take a dependency on your admission controller.

Risk is increased by:

- Failing Closed
- Too Broad of Scope
- Too Long Timeout

Failing Closed

- Failure Policy (*failurePolicy*) determines what api-server should do if there are issues communicating with the admission controller
  - Two options are: “**Ignore**” or “**Fail**”
  - **Default is “Fail”**

- **Do:** think through and consciously select a failure policy and understand the risk of failing closed
- **Don’t:** just go with the default value

Too Long Of Timeout

- **Timeout** (*timeoutSeconds*) field defines the **timeout** when calling your admission webhook
  - If a **timeout** occurs, the failure policy will be invoked
  - Default **timeout** is 10 seconds

- **Do:** use a short timeout that's reasonable for your admission webhook
  - Your webhook adds latency to apiserver, the timeout is a way to cap that latency
  - Extra apiserver latency can have severe impacts on clusters, especially at scale
- **Don't:** try to avoid timeouts to your webhook by just setting a long timeout value

Too Broad Of Scope

- Scope of webhook is defined by the “**rules**”
  - Can reduce scope with namespace and label selectors
  - **Rules** are the blast radius of your webhook

- **Do:** set the scope as tight as possible
- **Don’t:** use a large scope and try to reduce scope within your admission controller.

Demo - Let's Break A Cluster

https://drive.google.com/file/d/1qbSkllAtxXoU9UWs6LhjXLLUzIFqA5XF/view?usp=sharing

CEL (Yet another language)

- The Common Expression Language (CEL) implements common semantics for expression evaluation, enabling different applications to more easily interoperate.
  - CEL Playground: https://playcel.undistro.io/

- It’s only an expression language, not a scripting language
- It’s quite fast
- Very easy to write and understand
- Also easy to extend and embed

CEL (Yet another.. language)

Let’s validate this yaml object

```yaml
event:
    name: KubeCon
    location: Chicago
attendees:
    - has_coffe: true
      has_funny_hat: true
    # ....

```

CEL (Yet another.. language)

Let's validate this yaml object

```yaml
event:
    name: KubeCon
    location: Chicago
attendees:
    - has_coffe: true
      has_funny_hat: true
    # ....

```

Using CEL…

```
event.name == "KubeCon" &&
    event.location == "Chicago" &&
    attendees[0].has_coffe &&
    attendees[0].has_funny_hat
```

CEL (Yet another language)

Learn more about it’s benefits and limitations by watching Joe Betz talk titled

Webhook Fatigue? You're Not Alone: Introducing the CEL Expression Language Features

Safer Admission Control

- Validating Admission Policies are a new beta API as of 1.28
  - CEL based policies
  - As a new beta API these are default off in 1.29
  - Mutating Admission Policies are also planned for the future

- Match Conditions are a new beta feature as of 1.28
  - Allow for flexible CEL based selection on whether webhook or policy should be invoked
  - As a beta feature on an existing GA API these are default on

ValidatingAdmissionPolicy

- Validating admission policies offer a declarative, in-process alternative to validating admission webhooks.

- Benefits
  - Less latency
  - No external dependency
  - Less overhead
  - Immediate feedback

```yaml
apiVersion: admissionregistration.k8s.io/v1beta1
kind: ValidatingAdmissionPolicy
metadata:
  name: "demo-policy.example.com"
spec:
  failurePolicy: Fail
  matchConstraints:
    resourceRules:
    - apiGroups:   ["apps"]
      apiVersions: ["v1"]
      operations:  ["CREATE", "UPDATE"]
      resources:   ["deployments"]
  validations:
    - expression: "object.spec.replicas <= 5"
---
apiVersion: admissionregistration.k8s.io/v1beta1
kind: ValidatingAdmissionPolicyBinding
metadata:
  name: "demo-binding-test.example.com"
spec:
  policyName: "demo-policy.example.com"
  validationActions: [Deny]
  matchResources:
    namespaceSelector:
      matchLabels:
        environment: test
```

Match Conditions

- You can define match conditions for webhooks or policies if you need fine-grained request filtering using CEL.

```yaml
matchConditions:
  - name: 'exclude-api-server'
    expression: '!(request.userInfo.username == "system:apiserver")'
  - name: 'exclude-leases'
    expression: '!(request.resource.group == "coordination.k8s.io" && resource.resource == "leases")'
```

Demo - New CEL Based Features

https://drive.google.com/file/d/1zrKdoMvEhoTaAJm3zHf8T2DvWrx1LQBh/view?usp=sharing

Available Fields for CEL

Match conditions have access to the following CEL variables:

- `object` - The object from the incoming request. The value is null for DELETE requests. The object version may be converted based on the matchPolicy.
- `oldObject` - The existing object. The value is null for CREATE requests.
- `request` - The request portion of the AdmissionReview, excluding `object` and `oldObject`.
- `authorizer` - A CEL Authorizer. May be used to perform authorization checks for the principal (authenticated user) of the request. See Authz in the Kubernetes CEL library documentation for more details.
- `authorizer.requestResource` - A shortcut for an authorization check configured with the request resource (group, resource, (subresource), namespace, name).

https://kubernetes.io/docs/reference/access-authn-authz/extensible-admission-controllers/

Admission Request Fields


```go
// AdmissionRequest describes the admission.Attributes for the admission request.
type AdmissionRequest struct {
    UID                types.UID                `json:"uid" protobuf:"bytes,1,opt,name=uid"`
    Kind               metav1.GroupVersionKind      `json:"kind" protobuf:"bytes,2,opt,name=kind"`
    Resource           metav1.GroupVersionResource  `json:"resource" protobuf:"bytes,3,opt,name=resource"`
    SubResource        string                   `json:"subResource,omitempty" protobuf:"bytes,4,opt,name=subResource"`
    RequestKind        *metav1.GroupVersionKind     `json:"requestKind,omitempty" protobuf:"bytes,13,opt,name=requestKind"`
    RequestResource    *metav1.GroupVersionResource `json:"requestResource,omitempty" protobuf:"bytes,14,opt,name=requestResourc
    RequestSubResource string                   `json:"requestSubResource,omitempty" protobuf:"bytes,15,opt,name=requestSubR
    Name               string                   `json:"name,omitempty" protobuf:"bytes,5,opt,name=name"`
    Namespace          string                   `json:"namespace,omitempty" protobuf:"bytes,6,opt,name=namespace"`
    Operation          Operation                `json:"operation" protobuf:"bytes,7,opt,name=operation"`
    UserInfo           authenticationv1.UserInfo    `json:"userInfo" protobuf:"bytes,8,opt,name=userInfo"`
    Object             runtime.RawExtension         `json:"object,omitempty" protobuf:"bytes,9,opt,name=object"`
    OldObject          runtime.RawExtension         `json:"oldObject,omitempty" protobuf:"bytes,10,opt,name=oldObject"`
    DryRun             *bool                    `json:"dryRun,omitempty" protobuf:"varint,11,opt,name=dryRun"`
    Options            runtime.RawExtension         `json:"options,omitempty" protobuf:"bytes,12,opt,name=options"`
}
```

https://github.com/kubernetes/kubernetes/blob/master/staging/src/k8s.io/api/admission/v1/types.go

Conclusion

- Be mindful of your admission webhook settings
- Try out the new CEL features and provide feedback to the community!

Thank you! - Resource Links