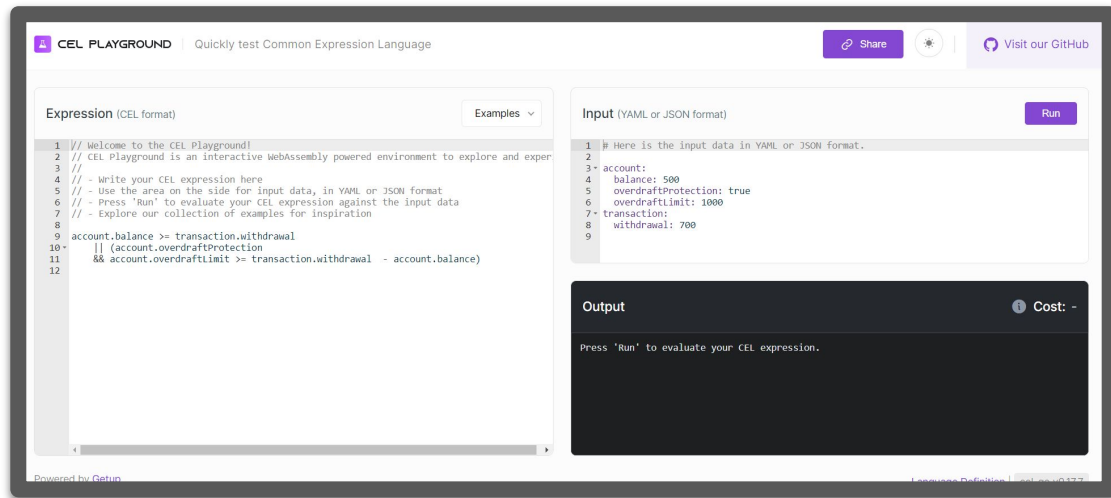






CEL PLAYGROUND



Kevin Conner, Director of Engineering
kev.conner@getupcloud.com

ABOUT GETUP



Agenda



CEL Playground

Common Expression Language (CEL)

- Quick Tour
- Kubernetes Extensions

Kubernetes Use Cases

- WebHooks
- ValidatingAdmissionPolicy
- Authentication Configuration Claim Mapping
- Authorization Configuration
- Custom Resource Validation

Questions

GETUP



CEL PLAYGROUND

GETUP

What is CEL Playground?

CEL Playground is an interactive WebAssembly (Wasm) powered environment to explore and experiment with the [Common Expression Language \(CEL\)](#) providing a simple and user-friendly interface to write and quickly evaluate CEL expressions for use in Kubernetes, Istio and other Cloud Native technologies.



Easy Sharing



Learning



Quick Testing



Avoid Mistakes

GETUP





CEL PLAYGROUND


GETUP

The original idea was to provide a simple browser-based testing environment for Common Expression Language (CEL), the Playground enable and enhance the user experience and exploration of CEL.

sig-api-machinery-cel-dev

 **Matheus Moraes** 11:03 AM
Hello everyone,
Does anyone working with `ValidatingAdmissionPolicy` use a playground for CEL or something similar?
I'm working on `Marvin`, a cluster scanner that also uses CEL, and I think a playground would help a lot in testing CEL expressions.
Something similar to `rego playground` and `CUE playground`

 **jpbetz** 4 months ago
This has been suggested a few times. I think it is very much needed, but I'm not aware of anyone building one yet.

 **jpbetz** 4 months ago
It might be possible to build it as a static site using WASM if anyone is interested in building it but concerned about hosting costs.

2

SIG API machinery Agenda & Notes... ☆ 📁 ☁ ⌚

File Edit View Insert Format Tools Extensions ...

🔍 ↶ ↷ 🖨 ⌨ 100% ▾ | Heading 3 ▾ ⋮

1 2 3 4 5 6

- Downside
 - Maintain effort
 - Immature itself(missing tooling)
- Tooling to experience with CEL in K8s (kubectl get, dry run, etc.)
 - Command line tool ([repo](#))
 - A cel playground would be a great idea (not client validation) (could be kubectl get format)

GETUP

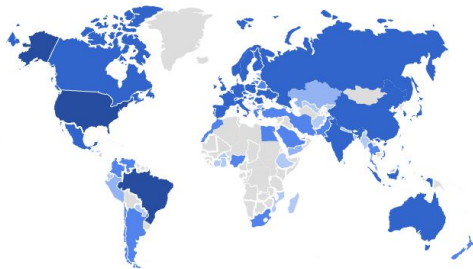


CEL PLAYGROUND

GETUP

Where is CEL Playground?

Users▼ by Country ID▼



| COUNTRY | USERS | |
|----------------|-------|---|
| United States | 725 | - |
| Brazil | 317 | - |
| India | 141 | - |
| Germany | 128 | - |
| United Kingdom | 126 | - |
| France | 102 | - |
| China | 82 | - |

Users

2.5K

New users

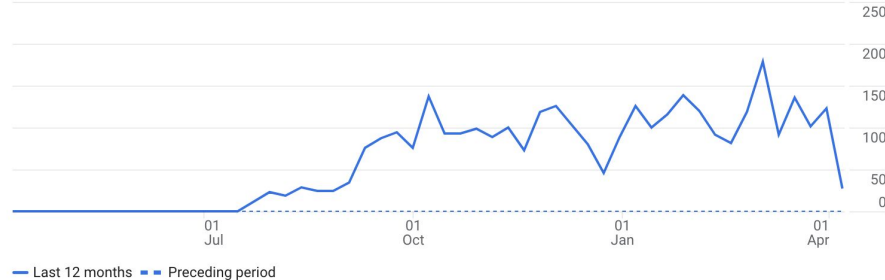
2.5K

Conversions

0

Event count

34K



GETUP



CEL PLAYGROUND

GETUP

Demo

playcel.undistro.io

GETUP



CEL PLAYGROUND

GETUP

Agenda

CEL Playground

Common Expression Language (CEL)

- Quick Tour
- Kubernetes Extensions

Kubernetes Use Cases

- WebHooks
- ValidatingAdmissionPolicy
- Authentication Configuration Claim Mapping
- Authorization Configuration
- Custom Resource Validation

Questions

GETUP



CEL PLAYGROUND

GETUP

Common Expression Language

- Developed by Google
 - Small & Fast
 - Embeddable
 - Extensible
- Supports
 - Logical Expressions
 - Relationships
 - Arithmetical Expressions
 - Conditional Expressions

&&, ||, !

==, !=, <, >, <=, >=, in

-, *, /, %, +, -

?:

.?

GETUP



CEL PLAYGROUND

GETUP

Common Expression Language

- Types

- Unsigned and Signed 64 bit integers -1, 0, 1, 255u
- 64-bit Double 3.1415926
- Booleans true, false
- Strings "Test string"
- Bytes b"abc", b"\303\277"
- Lists ["abc", "def"]
- Maps {"abc": <val>, "def": <val2>}
- Null null
- Duration duration("10m")
- Timestamp timestamp("2024-04-10T09:00:00-07:00")

Common Expression Language

- Conversions

- bytes
- integer
- unsigned integers
- double
- boolean
- strings
- bytes

`bytes("abc")` - string

`int(5u)` - uint, double, string, timestamp

`uint("5")` - int, double, string

`double("3.1415926")` - int, uint, string

`bool("true")`, `bool("false")`

`string(10.5)` - int, uint, double, bytes, timestamp, duration

`bytes("abc")` - string

Common Expression Language

- Macros

- `has(e.f)` `has(map_data.key4), has(obj_data.field1)`
- `e.all(x, p)` `list_data.all(x, x>10), map_data.all(k, k.startsWith("key"))`
- `e.exists(x, p)` `list_data.exists(x, x<10), map_data.exists(k, k > "key2")`
- `e.map(x, t)` `list_data.map(x, x+10)`
- `e.map(x, p, t)` `list_data.map(x, x>=12, x+10)`
- `e.filter(x, p)` `list_data.filter(x, x>=12)`

Common Expression Language

- Functions

- contains
- endsWith
- startsWith
- size
- matches

`"abcdef".contains("cd")`

`"abcdef".endsWith("ef")`

`"abcdef".startsWith("ab")`

`size("abcdef"), size(list_data), size(map_data)`

`"abcdef".matches("^.*c.*f$"), matches("abcdef", "^.*c.*f$")`

- Plus Timestamp/Duration functions

GETUP



CEL PLAYGROUND

GETUP

Kubernetes Extensions

- Optional Expressions `.?`
- Cross Type Numeric Comparisons `1.0 < 2`
- String Library Extensions
 - `charAt` `"abc".charAt(1)`
 - `indexOf` `"abcdefedcba".indexOf("c")`
 - `lastIndexOf` `"abcdefedcba".lastIndexOf("c")`
 - `lowerAscii` `"ABCDEF".lowerAscii()`
 - `replace` `"bababaaaba".replace("ab", "yz", 2)`
 - `split` `"abc, def, ghi".split(", ")`
 - `substring` `"bababaaaba".substring(3, 5)`
 - `trim` `" abcdef ".trim()`
 - `format` `"testing formatting: %s-%d".format(["abc", 5])`
 - `join` `["1", "2", "3"].join(",")`

Kubernetes Extensions

- URL
 - getScheme
`url("https://example.com/path?query=true").getScheme()`
 - getHost
`url("https://example.com/path?query=true").getHost()`
 - getHostname
`url("https://example.com/path?query=true").getHostname()`
 - getPort
`url("https://example.com:433/path?query=true").getPort()`
 - getEscapedPath
`url("https://example.com/p a t h ?query=true").getEscapedPath()`
 - getQuery
`url("https://example.com/path?query=true").getQuery()`
 - isURL
`isURL("https://example.com/path?query=true")`

Kubernetes Extensions

Decimal values - m, "", k, M, G, T, P, E

International System of Units(SI) - Ki, Mi, Gi, Ti, Pi, Ei

- Quantities

- isQuantity
- isGreaterThan
- isLessThan
- compareTo
- asApproximateFloat
- asInteger
- isInteger
- add
- sub

```
isQuantity("100Mi")
```

```
quantity("50k").isGreaterThan(quantity("90k"))
```

```
quantity("50k").isLessThan(quantity("90k"))
```

```
quantity("50k").compareTo(quantity("90k"))
```

```
quantity("3m").asApproximateFloat()
```

```
quantity("1Ki").asInteger()
```

```
quantity("50k").isInteger()
```

```
quantity("50k").add(quantity("90k")).asInteger() - can also be integer
```

```
quantity("50k").sub(quantity("90k")).asInteger() - can also be integer
```

GETUP



CEL PLAYGROUND

GETUP

Kubernetes Extensions

- Regex
 - find `"abcdef abc def abcddeff".find("c[a-z]*e")`
 - findAll `"abcdef abc def abcddeff".findAll("c[a-z]*e")`
- Lists
 - indexOf `[1, 5, 3, 1, 6, 3, 9].indexOf(3)`
 - lastIndexOf `[1, 5, 3, 1, 6, 3, 9].lastIndexOf(3)`
- Lists (Comparable Types)
 - isSorted `[1, 5, 3, 1, 6, 3, 9].isSorted()`
 - max `[1, 5, 3, 1, 6, 3, 9].max()`
 - Min `[1, 5, 3, 1, 6, 3, 9].min()`
- Lists (Summable Types)
 - sum `[1, 5, 3, 1, 6, 3, 9].sum()`

Kubernetes Extensions

Authorizer resource based authorization

- Obtain ResourceCheck for resource
- Refine with combination of
 - subresource
 - namespace
 - name
- Check verb for decision
- Check if allowed
- Check if errored

```
Authorizer.group("").resource("pods")
```

```
resourceCheck.subresource("status")
```

```
resourceCheck.namespace("default")
```

```
resourceCheck.name("podname")
```

```
resourceCheck.check("watch")
```

```
decision.allowed(), decision.reason()
```

```
decision.errored(), decision.error()
```

Kubernetes Extensions

Authorizer non-resource based authorization

- Obtain PathCheck for resource
- Check HTTP verb for decision
- Check if allowed
- Check if errored

```
Authorizer.path("/health")  
resourceCheck.check("get")  
decision.allowed(), decision.reason()  
decision.errored(), decision.error()
```

Service Account authorization

- Obtain Authorizer for service account
- Perform resource and non-resource based authorization

```
authorizer.serviceAccount("default", "myserviceaccount")
```

GETUP



CEL PLAYGROUND

GETUP

Demo

playcel.undistro.io

GETUP



CEL PLAYGROUND

GETUP

Agenda

CEL Playground

Common Expression Language (CEL)

- Quick Tour
- Kubernetes Extensions

 **Kubernetes Use Cases**

- **WebHooks**
- **ValidatingAdmissionPolicy**
- **Authentication Configuration Claim Mapping**
- **Authorization Configuration**
- **Custom Resource Validation**

Questions

GETUP



CEL PLAYGROUND

GETUP

Webhooks

Webhooks are

- Remote services for mutating and validating resources
- Invoked by the kubernetes API server
- Expensive
 - Time consuming network calls
 - Can Potentially fail
 - Coarse Grained access
 - Can filter using labels
 - namespaceSelector
 - objectSelector

```
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
```

Webhooks

CEL webhook integration

- Applies to both Mutating & Validating Webhooks
- Provides fine grained request filtering
- Runs within the kubernetes API server
- All conditions must evaluate to true

You can have up to 64matchConditions per webhook

matchConditions:

- **name:** 'exclude-leases' *# Each match condition must have a unique name*
expression: '!(request.resource.group == "coordination.k8s.io" && request.resource.resource == "leases")'
- **name:** 'exclude-kubelet-requests'
expression: '!("system:nodes" in request.userInfo.groups)' *# Match requests made by non-node users.*
- **name:** 'rbac' *# Skip RBAC requests, which are handled by the second webhook.*
expression: 'request.resource.group != "rbac.authorization.k8s.io"'

GETUP



CEL PLAYGROUND

GETUP

Webhooks

CEL matchConditions have access to

- 'object'
- 'oldObject'
- 'request'
- 'authorizer'
- 'authorizer.requestResource'

The object from the incoming request (null for DELETE)

The existing object (null for CREATE)

The request portion of **AdmissionReview**, not including **object** and **oldObject**

An Authorizer for performing authorization checks

Authorizer request configured within the requested resource group, resource, subresource, namespace, name

GETUP



CEL PLAYGROUND

GETUP

Demo

playcel.undistro.io

GETUP



CEL PLAYGROUND

GETUP

Validating Admission Policy

- An alternative to Validating Webhooks for many use cases
- Runs within the kubernetes API server
- Consists of
 - ValidatingAdmissionPolicy
 - ValidatingAdmissionPolicyBinding
 - Optional Parameter Resources
- Can Deny, Warn or Audit
- CEL Expressions used in
 - variables
 - matchConditions
 - validations
 - auditAnnotations

```
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
```

GETUP



CEL PLAYGROUND

GETUP

ValidatingAdmissionPolicy

CEL expressions have access to

- 'object' The object from the incoming request (null for DELETE)
- 'oldObject' The existing object (null for CREATE)
- 'request' The request portion of **AdmissionReview**, not including **object** and **oldObject**
- 'params' Optional parameters bound through the **ValidatingAdmissionPolicyBinding**
- 'namespaceObject' The namespace object for the incoming resource
- 'variables' The variables defined in the policy
- 'authorizer' An Authorizer for performing authorization checks
- 'authorizer.requestResource' Authorizer request configured within the requested resource group, resource, subresource, namespace, name

GETUP



CEL PLAYGROUND

GETUP

Validating Admission Policy

Variables

- Define variables for use in other expressions
- Evaluated on-demand
- Located under **variables**
- Can reference earlier variables

```
variables:  
- name: environment  
  expression: "'environment' in namespaceObject.metadata.labels ? namespaceObject.metadata.labels['environment'] : 'prod'"  
- name: exempt  
  expression: "'exempt' in object.metadata.labels && object.metadata.labels['exempt'] == 'true'"  
- name: containers  
  expression: "object.spec.template.spec.containers"  
- name: containersToCheck  
  expression: "variables.containers.filter(c, c.image.contains('example.com/'))"
```

GETUP



CEL PLAYGROUND

GETUP

Validating Admission Policy

Match Conditions

- Same as with Webhooks
- Provides fine grained request filtering
- Runs within the kubernetes API server
- All conditions must evaluate to true

```
matchConditions:  
  - name: 'exclude-leases' # Each match condition must have a unique name  
    expression: '!(request.resource.group == "coordination.k8s.io" && request.resource.resource == "leases")'  
  - name: 'exclude-kubelet-requests'  
    expression: '!("system:nodes" in request.userInfo.groups)' # Match requests made by non-node users.  
  - name: 'rbac' # Skip RBAC requests.  
    expression: 'request.resource.group != "rbac.authorization.k8s.io"'
```

GETUP



CEL PLAYGROUND

GETUP

Validating Admission Policy

Validations

- Only run if resource matched by **matchConditions**
- Runs within the kubernetes API server
- All validations must evaluate to true
- Can return friendly messages using expressions

```
variables:  
- name: environment  
  expression: "'environment' in namespaceObject.metadata.labels ? namespaceObject.metadata.labels['environment'] : 'prod'"   
- name: exempt  
  expression: "'exempt' in object.metadata.labels && object.metadata.labels['exempt'] == 'true'"   
- name: containers  
  expression: "object.spec.template.spec.containers"  
- name: containersToCheck  
  expression: "variables.containers.filter(c, c.image.contains('example.com/'))"  
validations:  
- expression: "variables.exempt || variables.containersToCheck.all(c, c.image.startsWith(variables.environment + '.'))"  
  messageExpression: "'only ' + variables.environment + ' images are allowed in namespace ' + namespaceObject.metadata.name"
```

Validating Admission Policy

Audit Annotations

- Only run if request is rejected
- Include friendly audit annotations in the audit event of the API request

```
validations:  
  - expression: "object.spec.replicas > 50"  
    messageExpression: "'Deployment spec.replicas set to ' + string(object.spec.replicas)"  
auditAnnotations:  
  - key: "high-replica-count"  
    valueExpression: "'Deployment spec.replicas set to ' + string(object.spec.replicas)"
```


Demo

playcel.undistro.io

GETUP



CEL PLAYGROUND

GETUP

Claim Mapping

CEL authentication integration

- Command Line option to API Server
- Applies to JWT Authenticators
- Supports
 - Claim Validation Rules
 - Claim Mapping
 - User Validation Rules

```
apiVersion: apiserver.config.k8s.io/v1beta1
kind: AuthenticationConfiguration
jwt:
- issuer:
  url: https://example.com
  audiences:
  - my-app
  - other-app
  audienceMatchPolicy: MatchAny
  claimValidationRules:
  - claim: hd
    requiredValue: example.com
  - expression: 'claims.hd == "example.com"'
    message: the hd claim must be set to example.com
  - expression: 'claims.exp - claims.nbf <= 86400'
    message: total token lifetime must not exceed 24 hours
  claimMappings:
    username:
      expression: 'claims.username + ":external-user"'
    groups:
      expression: 'claims.roles.split(",")'
    uid:
      claim: 'sub'
      extra:
      - key: 'client_name'
        valueExpression: 'claims.some_claim'
    userValidationRules:
  - expression: "!user.username.startsWith('system:')"
    message: username cannot use reserved system: prefix
  - expression: "user.groups.all(group, !group.startsWith('system:'))"
    message: groups cannot use reserved system: prefix
```

GETUP



CEL PLAYGROUND

GETUP

Claim Mapping

Claim Validation Rules

- Has access to token claims via 'claims' map

Claim Mapping

- Maps username, groups, uid and extra claims
- Has access to token claims via 'claims' map

User Validation Rules

- Has access to single 'user' map containing
 - username
 - groups
 - uid
 - extra

```
apiVersion: apiserver.config.k8s.io/v1beta1
kind: AuthenticationConfiguration
jwt:
- issuer:
  url: https://example.com
  audiences:
  - my-app
  - other-app
  audienceMatchPolicy: MatchAny
  claimValidationRules:
  - claim: hd
    requiredValue: example.com
  - expression: 'claims.hd == "example.com"'
    message: the hd claim must be set to example.com
  - expression: 'claims.exp - claims.nbf <= 86400'
    message: total token lifetime must not exceed 24 hours
  claimMappings:
  username:
    expression: 'claims.username + ":external-user"'
  groups:
    expression: 'claims.roles.split(",")'
  uid:
    claim: 'sub'
    extra:
    - key: 'client_name'
      valueExpression: 'claims.some_claim'
  userValidationRules:
  - expression: "!user.username.startsWith('system:')"
    message: username cannot used reserved system: prefix
  - expression: "user.groups.all(group, !group.startsWith('system:'))"
    message: groups cannot used reserved system: prefix
```

Authorization Configuration

CEL authorization integration

- Command Line option to API Server
- Specifies authorizer chain for server
- Enables use of multiple webhooks
- Filtered using CEL **matchConditions**
- Can access **SubjectAccessReview** contents
 - resourceAttributes
 - nonResourceAttributes
 - user
 - groups
 - uid
 - extra

GETUP

```
apiVersion: apiserver.config.k8s.io/v1beta1
kind: AuthorizationConfiguration
authorizers:
- name: system-crd-protector
  type: Webhook
  webhook:
    unauthorizedTTL: 30s
    timeout: 3s
    subjectAccessReviewVersion: v1
    matchConditionSubjectAccessReviewVersion: v1
    failurePolicy: Deny
    connectionInfo:
      type: KubeConfig
      kubeConfigFile: /kube-system-authz-webhook.yaml
    matchConditions:
      # only send resource requests to the webhook
      - expression: has(request.resourceAttributes)
      # only intercept requests to kube-system
      - expression: request.resourceAttributes.namespace == 'kube-system'
      # don't intercept requests from kube-system service accounts
      - expression: !('system:serviceaccounts:kube-system' in request.user.groups)
- type: Node
- type: RBAC
- name: opa
  type: Webhook
  webhook:
    unauthorizedTTL: 30s
    timeout: 3s
    subjectAccessReviewVersion: v1
    matchConditionSubjectAccessReviewVersion: v1
    failurePolicy: Deny
    connectionInfo:
      type: KubeConfig
      kubeConfigFile: /opa-kube-system-authz-webhook.yaml
    matchConditions:
      # only send resource requests to the webhook
      - expression: has(request.resourceAttributes)
      # only intercept requests to kube-system
      - expression: request.resourceAttributes.namespace == 'kube-system'
      # don't intercept requests from kube-system service accounts
      - expression: !('system:serviceaccounts:kube-system' in request.user.groups)
```



Custom Resource Validation

- Validation rules bound within schema
 - x-kubernetes-validations
- Rules bound within the schema
- Rules have access to
 - self
 - oldSelf

```
...
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
```

Custom Resource Validation

- Rules can be bound to
 - Root object
 - 'apiVersion', 'kind', 'metadata.name' & 'metadata.namespace'
 - Embedded object
 - Properties accessible via 'self.field'
 - Object with additionalProperties
 - Properties accessible via 'self["name"].field'
 - An Array
 - Accessed via 'self[index].field'
 - A Scalar
 - Self is bound to scalar value

GETUP



CEL PLAYGROUND

GETUP

Agenda

CEL Playground

Common Expression Language (CEL)

- Quick Tour
- Kubernetes Extensions

Kubernetes Use Cases

- WebHooks
- ValidatingAdmissionPolicy
- Authentication Configuration Claim Mapping
- Authorization Configuration
- Custom Resource Validation



Questions

GETUP



CEL PLAYGROUND

GETUP

CEL Playground RoadMap

On the Road to [CNCF sandbox!](#)

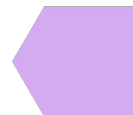
Better support for Kubernetes use cases



ValidatingAdmissionPolicy **validations**



ValidatingWebhookConfiguration
matchConditions



CRD **validations**



Authorization **matchConditions**

**Supporting other
CNCF & OSS projects**

Istio, Envoy & Google Cloud Certificate Authority Service...
Others?

Testing on live clusters



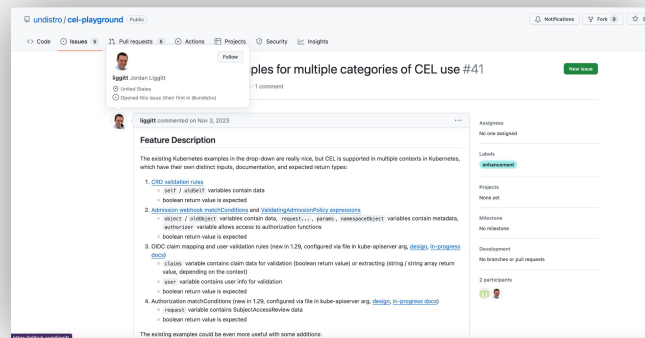
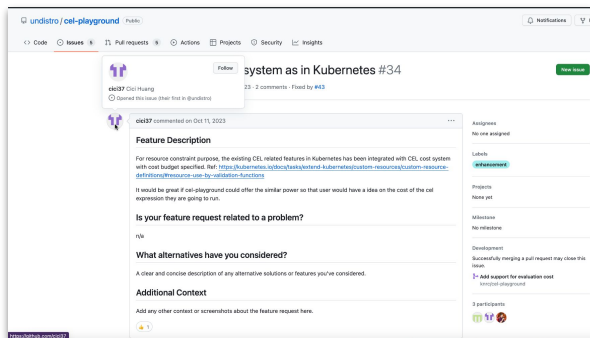
CEL PLAYGROUND

GETUP

Give Us your Feedback, Please!

<https://github.com/undistro/cel-playground/issues>

Be like [Jordan](#), [CiCi](#) and others!
Give feedback, submit an issue!



CEL PLAYGROUND

GETUP

Are you already using CEL Playground?

Support our community efforts!

Add your organization to our Adopters.md file here:

<https://github.com/undistro/cel-playground/blob/main/ADOPTERS.md>



Contributions are welcome!

👉 **CEL Playground:** playcel.undistro.io

👉 **GitHub Repository:** github.com/undistro/cel-playground

CONTACT US

Matheus Faria: matheus.faria@getupcloud.com

Kevin Conner: kev.conner@getupcloud.com

Today's slides can be downloaded from:

<https://github.com/undistro/cel-playground/blob/main/presentations/OpenShiftCommons/2024-04-10.pdf>

GETUP



CEL PLAYGROUND

GETUP

A modern office interior with a glass wall. On the left, a glowing blue 'GETUP' logo is visible on the wall. In the center, a large wooden conference table is surrounded by chairs. Above the table, three pendant lights hang. To the right, a large potted plant is visible. The scene is framed by a dark metal grid, and green plants are visible in the foreground.

GETUP

THANK YOU!

GETUP.IO