KubeCon | CloudNativeCon

Europe 2023

SIG-Auth Deep Dive

David Eads (Red Hat), Jordan Liggitt (Google), Rita Zhang (Microsoft)

Code of Conduct

Remember the **Golden Rule**: Treat others as you would want to be treated - with kindness and respect

Scan the QR code to access and review the **CNCF Code of Conduct**:

SCAN ME

Virtual Audience Closed Captioning

KubeCon | CloudNativeCon Europe 2023

Closed captioning for the virtual audience is available during each session through Wordly. The Wordly functionality can be found under the “Translation” tab on the session page.

Wordly will default to English. If another language is needed, simply click the dropdown at the bottom of the “Translation” tab and choose from one of 26+ languages available so you don’t miss a beat from our presenters.

*Note: Closed captioning is ONLY available during the scheduled live sessions and will not be available for the recordings on-demand within the virtual conference platform.

Session Q+A

- Virtual attendees may submit questions to speakers through the CNCF Slack channel: **#2-Kubecon-sessions**

- Please create a thread and tag the speaker(s) with questions about their talk.

- Questions will be answered by the speaker and/or other community members after the session concludes.

Sponsor Shout-Out!

Thank you to our Session Recording Sponsor:

uptycs

Enjoy the Session!

KubeCon | CloudNativeCon

Europe 2023

SIG-Auth Deep Dive

David Eads, Red Hat
Jordan Liggitt, Google
Rita Zhang, Microsoft

SIG-Auth

KubeCon | CloudNativeCon Europe 2023

What do we do?

SIG-Auth is responsible for features in Kubernetes that control and **protect access** to the API and other core components. This includes **authentication** and **authorization**, but also encompasses features like **auditing** and some **security policy**.

https://github.com/kubernetes/community/blob/master/sig-auth/charter.md

SIG-Auth

Sub-Projects

- Audit Logging
- Authenticators
- Authorizers
- Certificates
- Encryption at rest
- Multi Tenancy
- Node Identity and Isolation
- Policy Management
- Service Accounts
- Secrets Store CSI Driver

Enhancements

Stable

- 2799 Reduce legacy service account token attack surface area
  - Stop auto-generating legacy tokens (beta v1.24, stable v1.26)

Implementable

- 2799 Reduce legacy service account token attack surface area
  - Track use of legacy tokens (beta v1.27, targeting stable v1.28)
  - Clean up unused legacy tokens (targeting alpha v1.28)
- 3325 API to get current user attributes, `kubectl whoami` (beta v1.27, targeting stable v1.28)
- 3299 KMS v2 encryption at rest (beta v1.27, targeting stable v1.29)
- 3257 Cluster Trust Bundles (API alpha in v1.27, targeting volume mount alpha in v1.28)

Provisional

- 3221 Structured Authorization Configuration (in design, targeting alpha v1.28)
- 3331 Structured OIDC Configuration (in design, targeting alpha v1.28)
- 3766 ReferenceGrant (in design)
- 2718 Client Exec Proxy (in design)

KubeCon | CloudNativeCon

Europe 2023

Graduating Enhancements

Legacy Token Reduction

KEP-2799

Move away from long lived secret based SA tokens
Use ephemeral tokens via the token request API instead

- In v1.24+, secret based tokens default to not being auto-generated (**LegacyServiceAccountTokenNoAutoGeneration**
- In v1.26, feature is stable

Implementable Enhancements

Legacy Token Reduction

KEP-2799

Move away from long lived secret based SA tokens
Use ephemeral tokens via the token request API instead

- v1.26: **LegacyServiceAccountTokenTracking** alpha feature gate
  - Use of auto-generated secret based tokens emits warnings
  - Use of secret based tokens labels the secret with the last used date
- v1.27: **LegacyServiceAccountTokenTracking** feature is beta
- v1.28: Targeting alpha of cleanup of unused auto-generated token secrets

SelfSubjectReview

KEP-3325 - Beta in v1.27, targeting stable in v1.28

```
kubectl auth whoami -o json

{
    "kind": "SelfSubjectReview",
    "apiVersion": "authentication.k8s.io/v1beta1",
    "metadata": {
        "creationTimestamp": "2023-04-21T17:05:55Z"
    },
    "status": {
        "userInfo": {
            "username": "alice",
            "groups": [
                "system:authenticated"
            ]
        }
    }
}

--runtime-config=authentication.k8s.io/v1beta1=true
--feature-gates=APISelfSubjectReview=true
```

Encryption at Rest Improvements - v1 & v2

- Change algorithm for KMS data encryption to AES-GCM. No user action required.
  - v1.25: KMS v1 write using AES-GCM, read AES-GCM with fallback to AES-CBC
  - KMS v2 only allows AES-GCM
- Dynamic reload of *EncryptionConfiguration* file - v1.26
  - Does not require a kube-apiserver restart
- Custom resource encryption - v1.26
  - Add custom resources to *EncryptionConfiguration*
- Encrypt all resources - v1.27
  - `*.*` to encrypt all resources in all groups
    `*.<group>` to encrypt all resources in one group

```yaml
kind: EncryptionConfiguration
apiVersion: apiserver.config.k8s.io/v1
resources:
- resources:
  - secrets
  providers: …

- resources:
  - pandas.awesome.bears.com   🐼
  providers: …

- resources:
  - "*.*"
  providers: …
```

KMS v1 Gaps

- **Performance**
  - a new Data Encryption Key (DEK) is generated for each encryption
  - slow cluster startup times due to the large number of requests made to the remote vault
  - rate limits on external KMS
  - 160ms per request
- **Key Rotation**
  - manual and error-prone
  - hard to determine what keys are in-use
- **Health check & status**
  - kube-apiserver has to make encrypt and decrypt calls to determine KMS plugin health
- **Observability**
  - unable to correlate events across kube-apiserver, KMS plugin, and external KMS

What's new in KMS v2?

KEP-3299 - Alpha v1.25, v1.26, Beta v1.27, Stable targeted v1.29

- Benefits
  - Performance - DEK Re-use in API Server (80µs per request)
  - Health check & status - new status API health of the KMS plugin
  - Observability - new UID generated for each envelope operation.
  - New proto format for stored data in etcd
  - Key Rotation - key_id from status API is used to track current KMS key and KEK rotation allowing rotation without API server restart
- kubernetes.io/docs/tasks/administer-cluster/kms-provider
- Get involved #sig-auth-kms-dev on slack

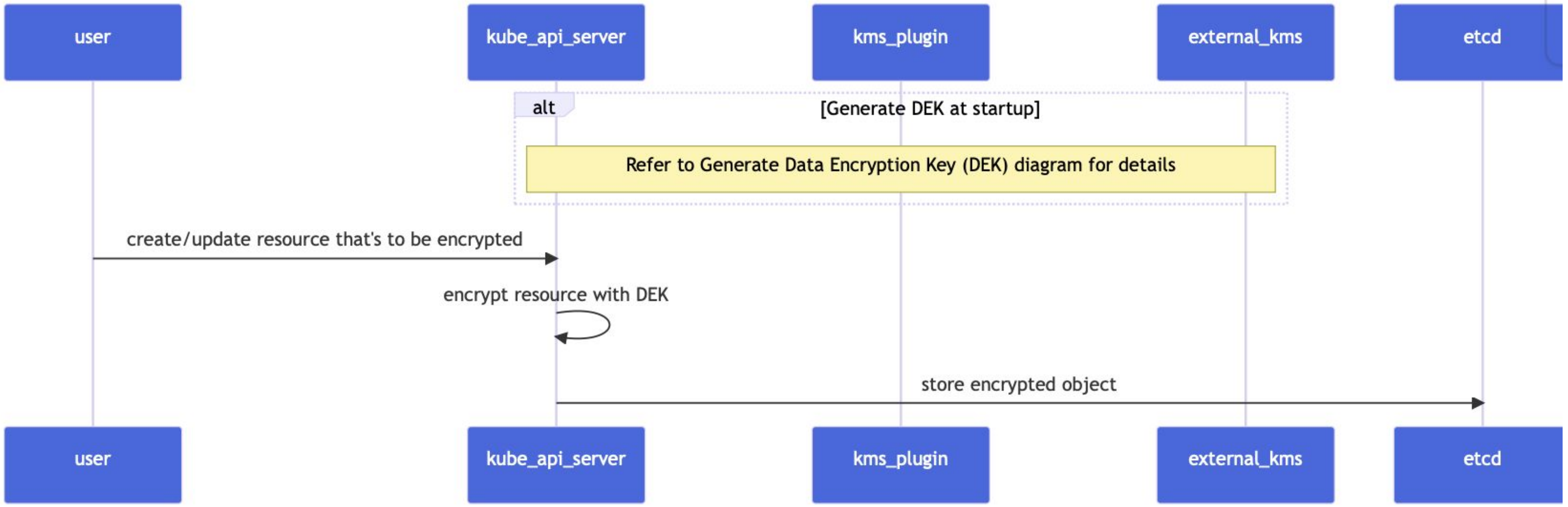
New proto format for stored data in etcd

For storage, a new structured protobuf format.

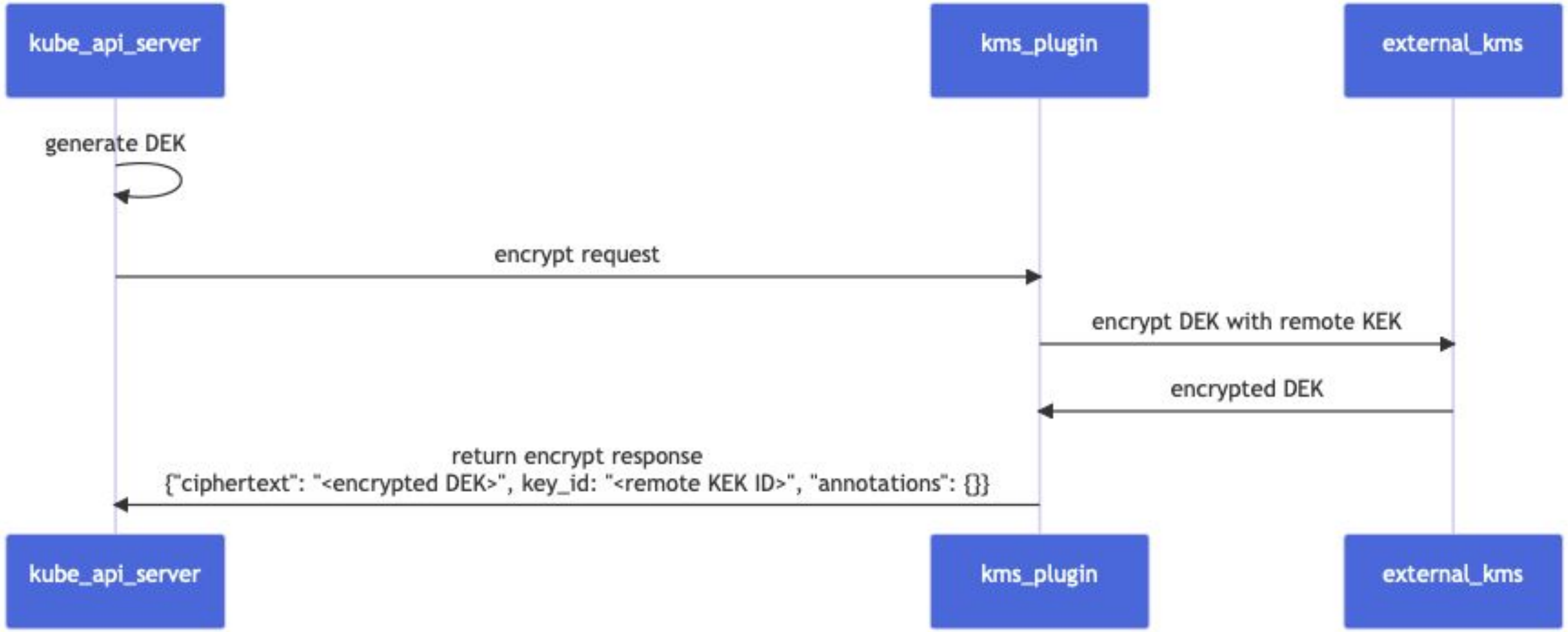
The prefix for the new format is `k8s:enc:kms:v2:<config name>:`.

```go
// EncryptedObject is the representation of data stored in etcd after envelope encryption.
type EncryptedObject struct {
   // EncryptedData is the encrypted data.
   EncryptedData []byte `protobuf:"bytes,1,opt,name=encryptedData,proto3"`

   // KeyID is the KMS key ID used for encryption operations.
   KeyID string `protobuf:"bytes,2,opt,name=keyID,proto3"`

   // EncryptedDEK is the encrypted DEK.
   EncryptedDEK  []byte `protobuf:"bytes,3,opt,name=encryptedDEK,proto3"`

   // Annotations is additional metadata that was provided by the KMS plugin.
   Annotations   map[string][]byte `protobuf:"name=annotations"`
}
```

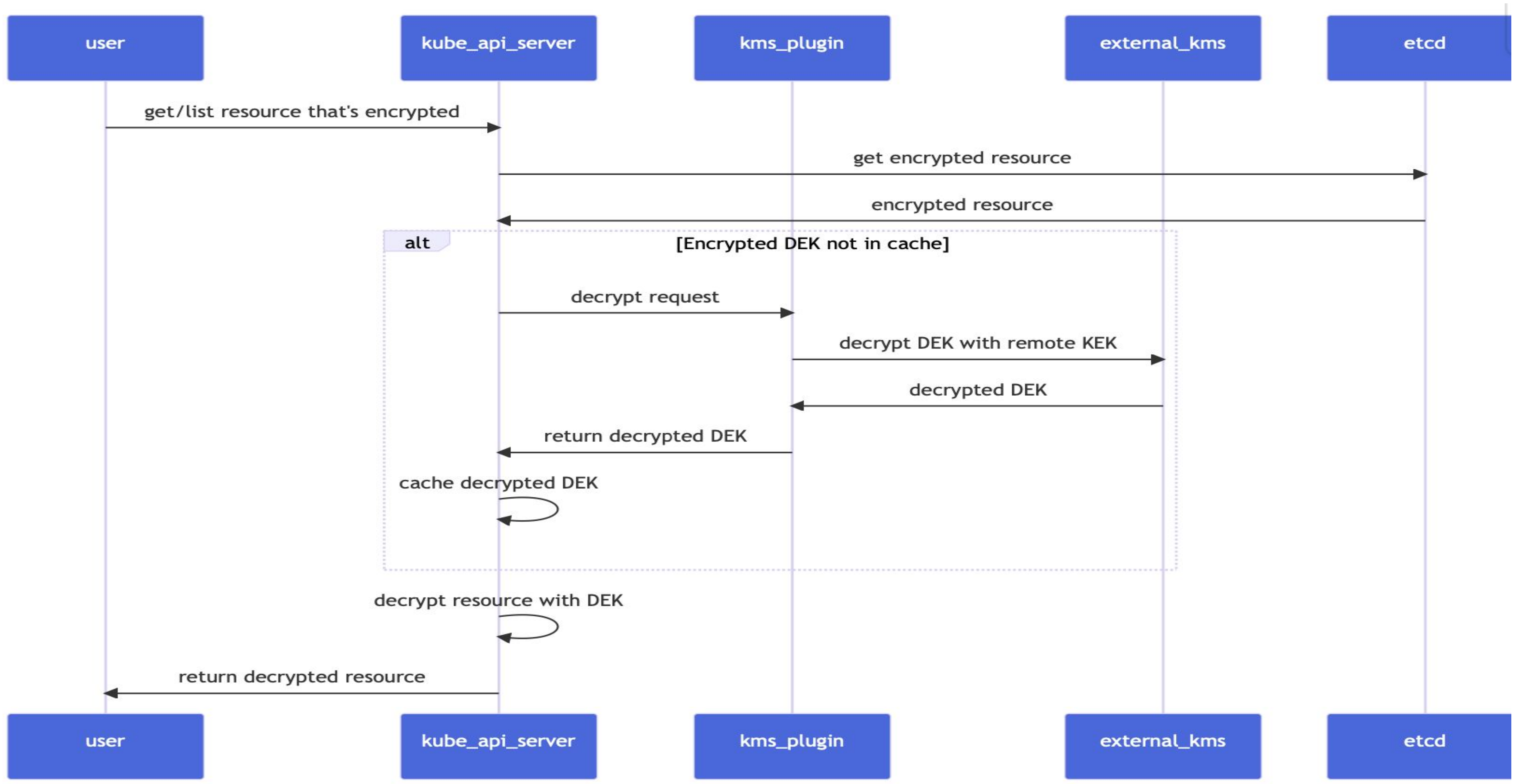
KMS v2 - Encrypt Request


user
kube_api_server
kms_plugin
external_kms
etcd
alt
[Generate DEK at startup]
Refer to Generate Data Encryption Key (DEK) diagram for details
create/update resource that's to be encrypted
encrypt resource with DEK
store encrypted object
user
kube_api_server
kms_plugin
external_kms
etcd

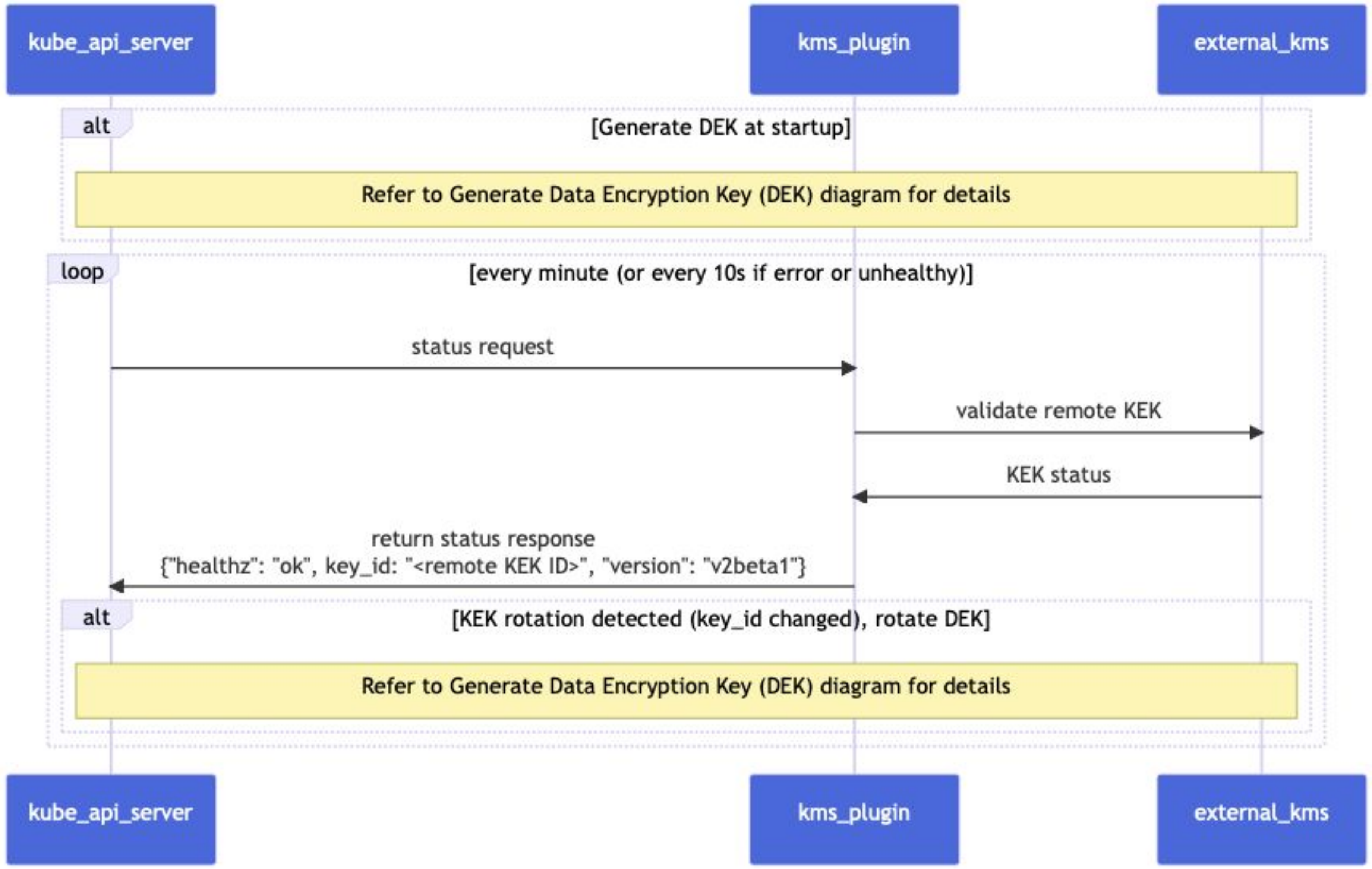
KMS v2 - Generate DEK



KMS v2 - Decrypt Request



KMS v2 - Status Request


kube_api_server
kms_plugin
external_kms
alt
[Generate DEK at startup]
Refer to Generate Data Encryption Key (DEK) diagram for details
loop
[every minute (or every 10s if error or unhealthy)]
status request
validate remote KEK
KEK status
return status response
{"healthz": "ok", key_id: "<remote KEK ID>", "version": "v2beta1"}
alt
[KEK rotation detected (key_id changed), rotate DEK]
Refer to Generate Data Encryption Key (DEK) diagram for details
kube_api_server
kms_plugin
external_kms

Cluster Trust Bundles

KubeCon | CloudNativeCon Europe 2023

KEP-3257

```go
type ClusterTrustBundleSpec struct {
 // The name of the associated signer.
 // +optional
 SignerName string `json:"signerName,omitempty"`

 // The individual trust anchors for this bundle.
 // A PEM bundle of PEM-wrapped,
 // DER-formatted X.509 certificate.
 // The order of certificates has no meaning.
 PEMTrustAnchors string `json:"pemTrustAnchors"`
}
```

DRAFT

```yaml
apiVersion: v1
kind: Pod
metadata:
 namespace: client
 name: client
spec:
 containers:
   name: main
   image: my-image
   volumeMounts:
    - mountPath: /var/run/example-com-server-tls-trust-anchors
      name: example-com-server-tls-trust-anchors
      readOnly: true
 volumes:
  - name: example-com-server-tls-trust-anchors
    projected:
      sources:
+       - pemTrustAnchors:
+           signerName: example.com/server
+           path: ca_certificates.pem
```

Cluster Trust Bundles

Trust the kube-apiserver, find your CA bundle

```
apiVersion: certificates.k8s.io/v1
kind: CertificateSigningRequest
spec:
 signerName: example.com/foo

apiVersion: certificates.k8s.io/v1alpha1
kind: ClusterTrustBundle
metadata:
 name: example.com:foo:v12
spec:
 signerName: example.com/foo
 trustBundle: [content here]
```

Current

```
kubectl get clustertrustbundle
--field-selector=spec.signerName=example.com/foo
```

Coming Soon

```
volumes:
+  - name: example-com-server-tls-trust-anchors
+    projected:
+      sources:
+      - pemTrustAnchors:
+          signerName: example.com/foo
+          path: ca_certificates.pem
```

Future

- Projected volume for workload client certificates? Pre-KEP proposal

KubeCon | CloudNativeCon
Europe 2023

In-Design Enhancements

Structured Authorization Configuration

KubeCon CloudNativeCon Europe 2023

KEP-3221

This proposal would allow:
- Dynamic config reload
- Multiple webhooks
- Configurable failure policy
- CEL-based filtering

```yaml
apiVersion: apiserver.config.k8s.io/v1alpha1
kind: AuthorizationConfiguration
authorizers:
  - type: Webhook
    webhook:
      unauthorizedTTL: 30s
      timeout: 3s
      subjectAccessReviewVersion: v1
      onError: Deny
      kubeConfigFile: /kube-system-authz-webhook.yaml
      matchConditions:
        - expression: |
            request.resourceAttributes.namespace == 'kube-system'
        - expression: |
            !('system:serviceaccounts:kube-system' in request.user.groups)
  - type: Node
  - type: RBAC
  - type: Webhook
    webhook:
      authorizedTTL: 5m
      unauthorizedTTL: 30s
      timeout: 3s
      subjectAccessReviewVersion: v1
      onError: NoOpinion
      kubeConfigFile: /authz-webhook.yaml
```

Structured OIDC Configuration

KubeCon | CloudNativeCon Europe 2023

KEP-3331

This proposal would allow:

- Dynamic config reload
- Multiple OIDC providers
- CEL-based authentication validation rules
- CEL-based claim extraction for user/group attributes
- Support for non-OIDC ID JWT credentials

```
claimValidationRules:
- rule: 'claims.aud == "charmander" || claims.aud == "bulbasaur"'
  message: clients other than charmander or bulbasaur are not allowed
- rule: 'claims.roles.split(",").exists(r, r == "kubernetes-user")'
  message: only kubernetes-user group members can access the cluster
- rule: '!claims.username.startsWith("system:")'
  message: system identities are not allowed

userValidationRules:
- rule: '!user.username.startsWith("myidp.")'
  message: system identities are not allowed
```

```
claimMappings:
  username: 'claims.username + ":external-user"'
  groups: 'claims.roles.split(",")'
  uid: 'claims.sub'
  extra:
  - key: '"client_name"'
    expression: 'claims.aud'
```

ReferenceGrant

KEP-3766

- Expansion of narrow API used by sig-network, sig-storage
- Goals
  - Users express intent to allow use of a resource
  - Controllers verify that intent was granted
  - Admins avoid broad permission grants to controllers
- Example uses
  - Gateway cross-namespace TLS secrets
  - PersistentVolumeClaim cross-namespace data sources

KubeCon | CloudNativeCon
Europe 2023

Cross-SIG Work

Cross SIG Work

- CEL for Admission Control (api-machinery)
  - kubernetes/enhancements#3488
- Storage Version API (api-machinery)
  - kubernetes/enhancements#2339
- Kube API Server Identity (api-machinery)
  - kubernetes/enhancements#1965

Shout outs!

🙏 THANK YOU! 🙏

Anish @aramase KMSv2	Krzysztof @ibihim KMSv2 / kube-rbac-proxy	Nilekh @nilekhc KMSv2	Maksim @nabokihms whoami
Taahir @ahmedtd Cluster Trust Bundle	Standa @stlaz kube-rbac-proxy	Roma @r-erema OIDC tests	

SIG-Auth

Where can you find us?

Slack channel: #sig-auth

Home page: https://github.com/kubernetes/community/tree/master/sig-auth

Mailing list: https://groups.google.com/forum/#!forum/kubernetes-sig-auth

Bi-weekly meetings Wednesday at 11PT (agenda/recordings links on home page)