Audit in Kubernetes, the Future is Here

Maciej Szulik, Stefan Schimanski – Red Hat
@soltysh @the_sttts

kubernetes

Quick history lesson anyone?

https://unsplash.com

Basic Audit in v1.4

```
AUDIT: id="5c3b8227-4af9-4322-8a71-542231c3887b"
    ip="127.0.0.1" method="GET" user="admin" as="<self>"
    asgroups="<lookup>" namespace="default"
    uri="/api/v1/namespaces/default/pods"


AUDIT: id="5c3b8227-4af9-4322-8a71-542231c3887b" response="200"
```

Advanced Audit

https://unsplash.com

Advanced Audit (as alpha in v1.7)

Meta data output & **full objects** for request/response

JSON or **text**-based **file output** & **webhook** support

Filtering with a policy

Configurable consistency with batching and flush

Advanced Audit GA’d in v1.12

https://unsplash.com

An Audit Event

audit.k8s.io/v1

- one event per request
- to be filled by apiservers
- sent to audit backend

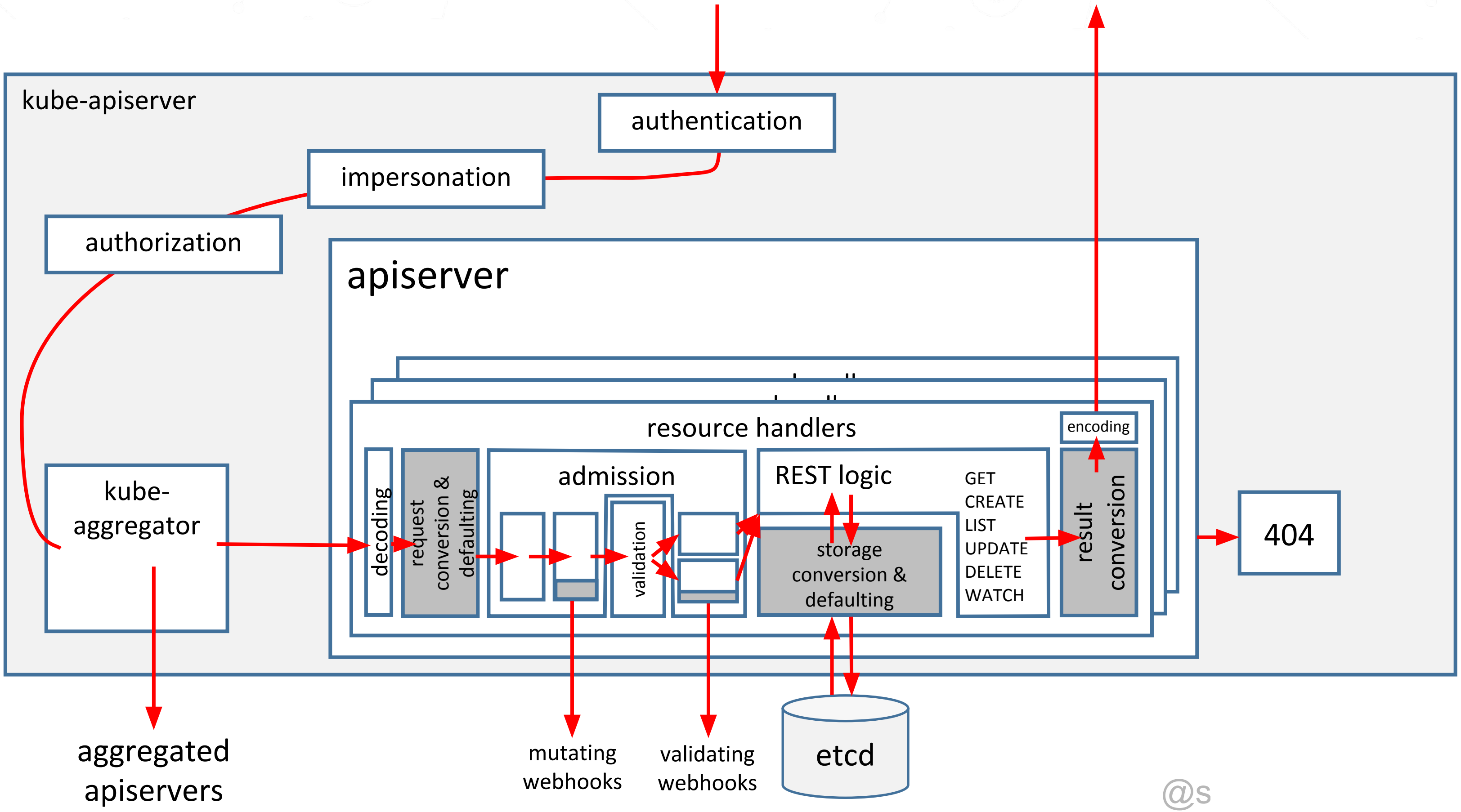
```
type Event struct {
    Level Level
    AuditID types.UID
    Stage Stage
    RequestURI string
    Verb string
    Annotations map[string]string
    RequestReceivedTimestamp metav1.MicroTime
    StageTimestamp metav1.MicroTime
    User authnv1.UserInfo
    ImpersonatedUser *authnv1.UserInfo
    SourceIPs []string
    UserAgent string
    ObjectRef *ObjectReference
    ResponseStatus *metav1.Status
    RequestObject *runtime.Unknown
    ResponseObject *runtime.Unknown

}
```

metadata

when?

who?

what?



kube-apiserver

authentication

impersonation

authorization

apiserver

resource handlers

encoding

decoding

request conversion & defaulting

admission

validation

REST logic

GET
CREATE
LIST
UPDATE
DELETE
WATCH

storage conversion & defaulting

result conversion

kube-aggregator

404

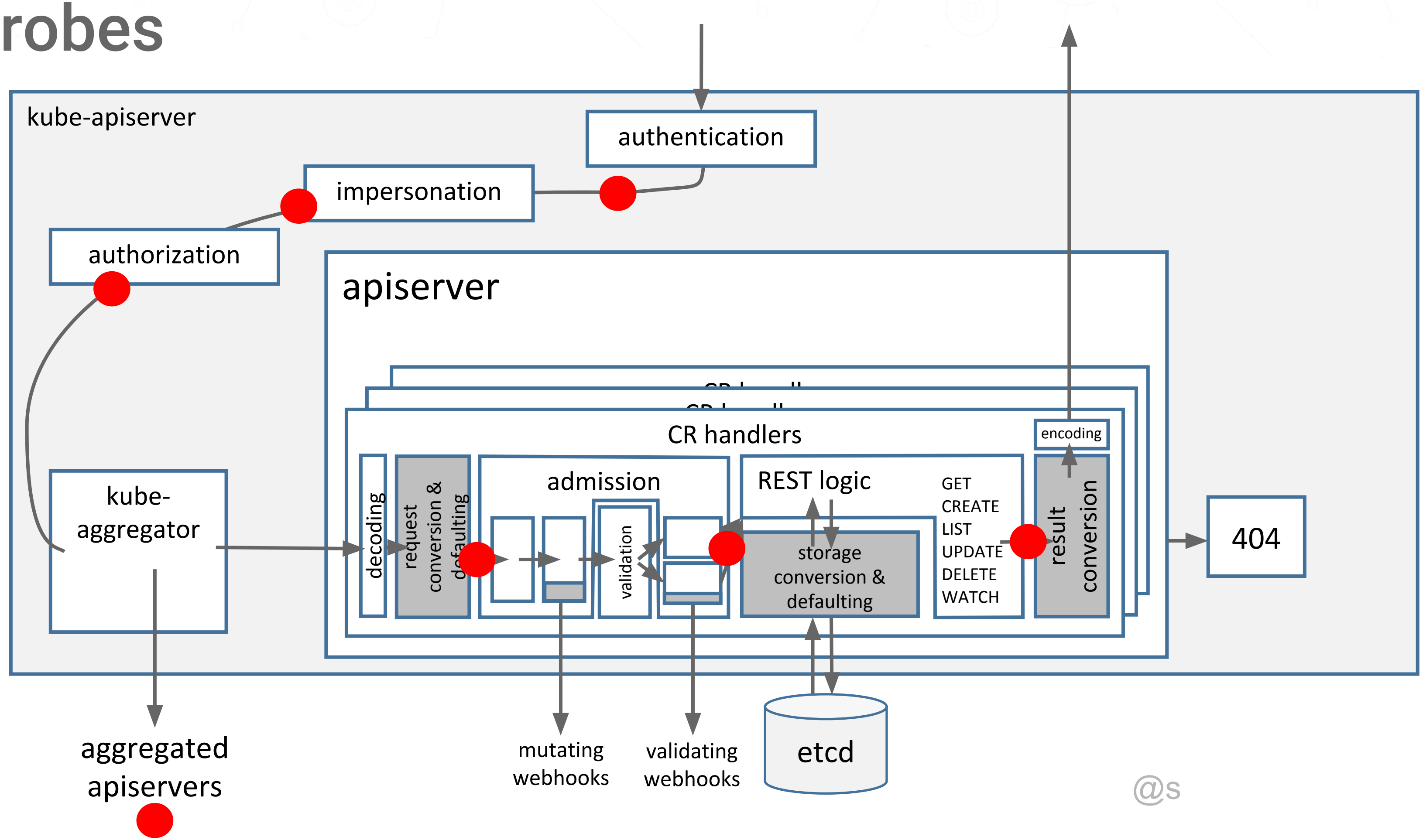
aggregated apiservers

mutating webhooks

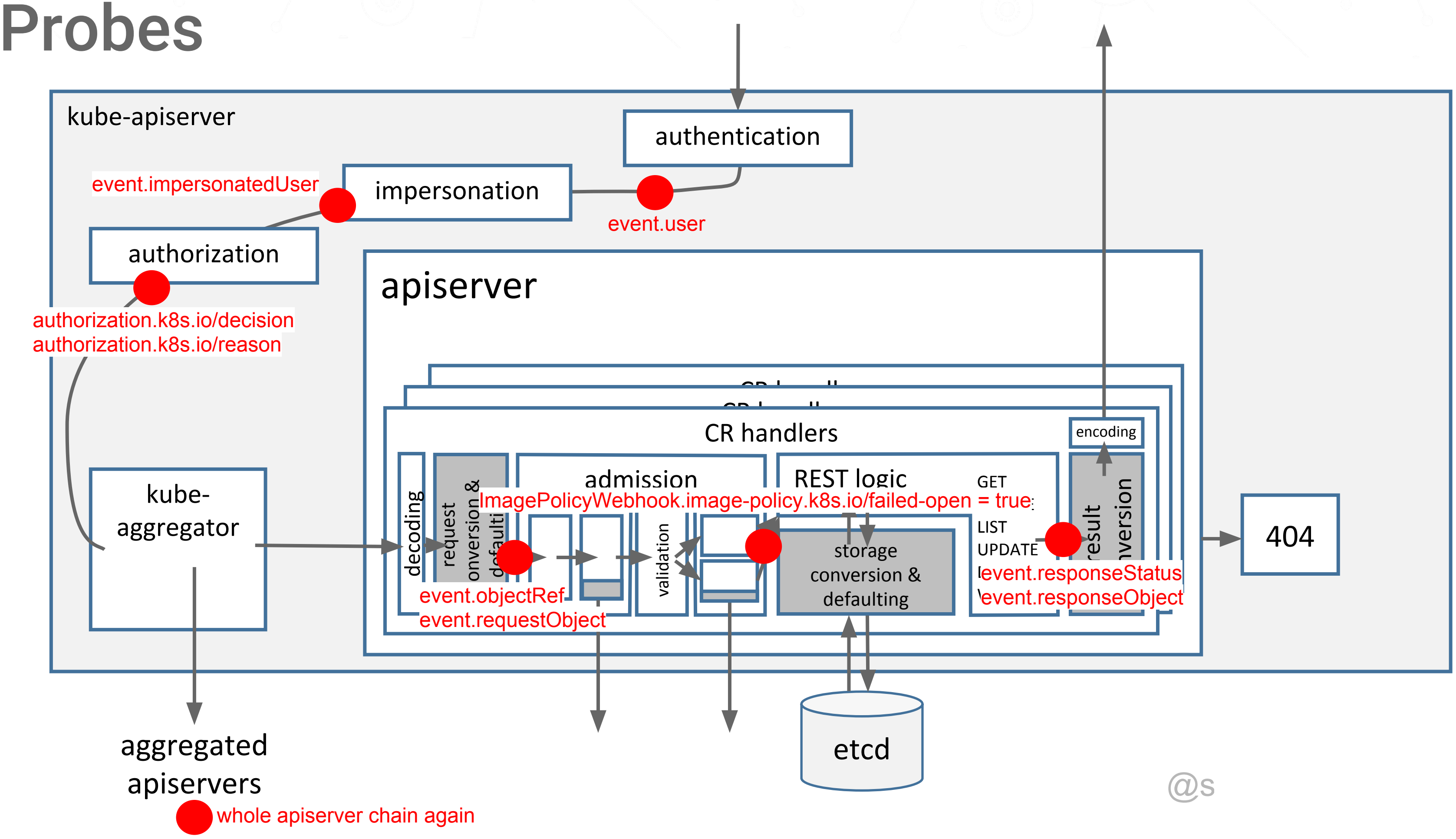
validating webhooks

etcd

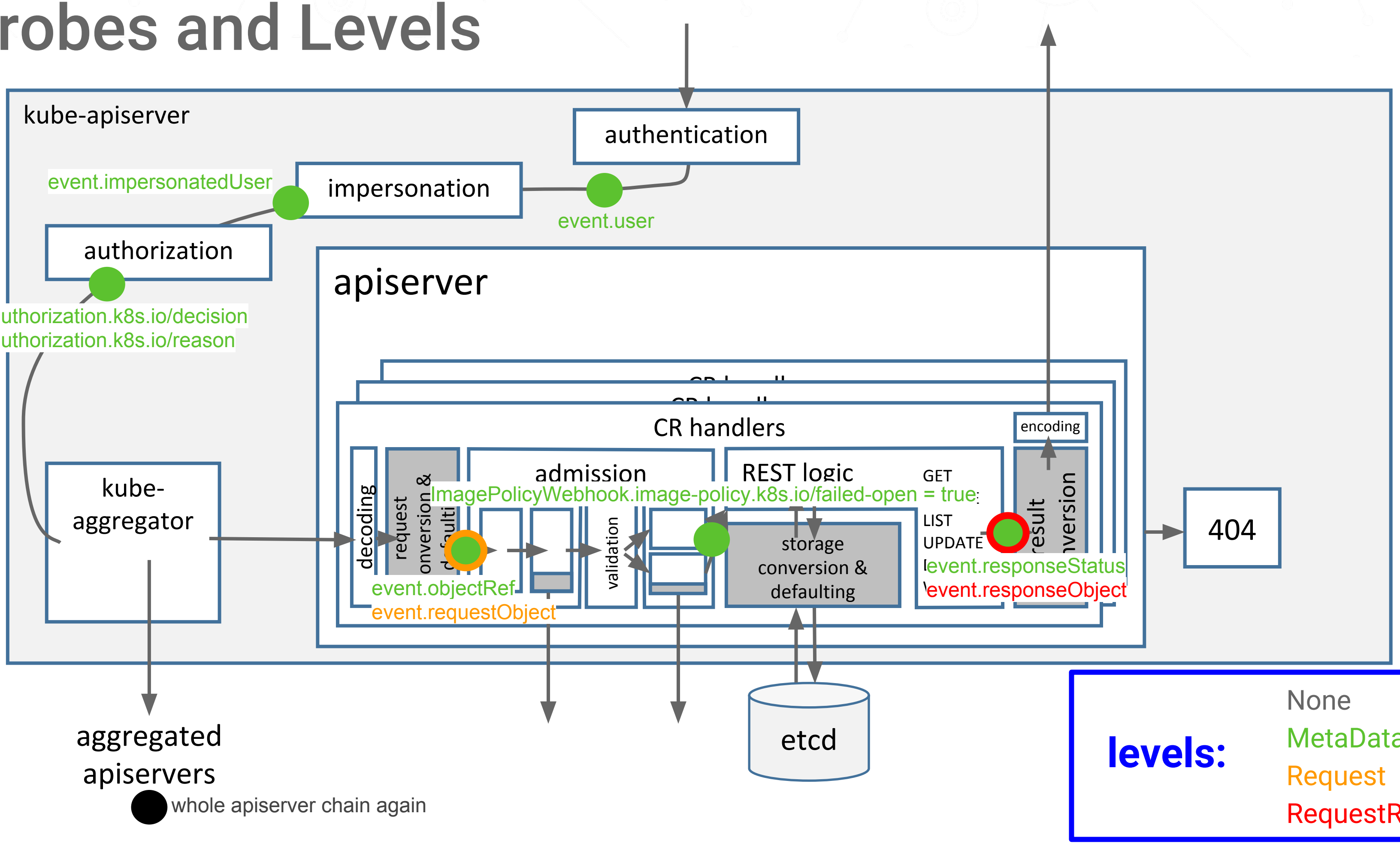
Probes


kube-apiserver
authentication
impersonation
authorization
apiserver
CR handlers
CR handlers
CR handlers
encoding
admission
REST logic
GET
CREATE
LIST
UPDATE
DELETE
WATCH
decoding
request conversion & defaulting
validation
storage conversion & defaulting
result conversion
kube-aggregator
404
aggregated apiservers
mutating webhooks
validating webhooks
etcd
@s

Probes


kube-apiserver
authentication
impersonation
event.impersonatedUser
event.user
authorization
authorization.k8s.io/decision
authorization.k8s.io/reason
apiserver
CR handlers
encoding
decoding
request conversion & defaulti
admission
ImagePolicyWebhook.image-policy.k8s.io/failed-open = true
validation
REST logic
GET
LIST
UPDATE
storage conversion & defaulting
result conversion
event.responseStatus
event.responseObject
event.objectRef
event.requestObject
kube-aggregator
404
etcd
aggregated apiservers
whole apiserver chain again
@s

Probes and Levels



levels:

None
MetaData
Request
RequestResponse

Performance

vs.

consistency

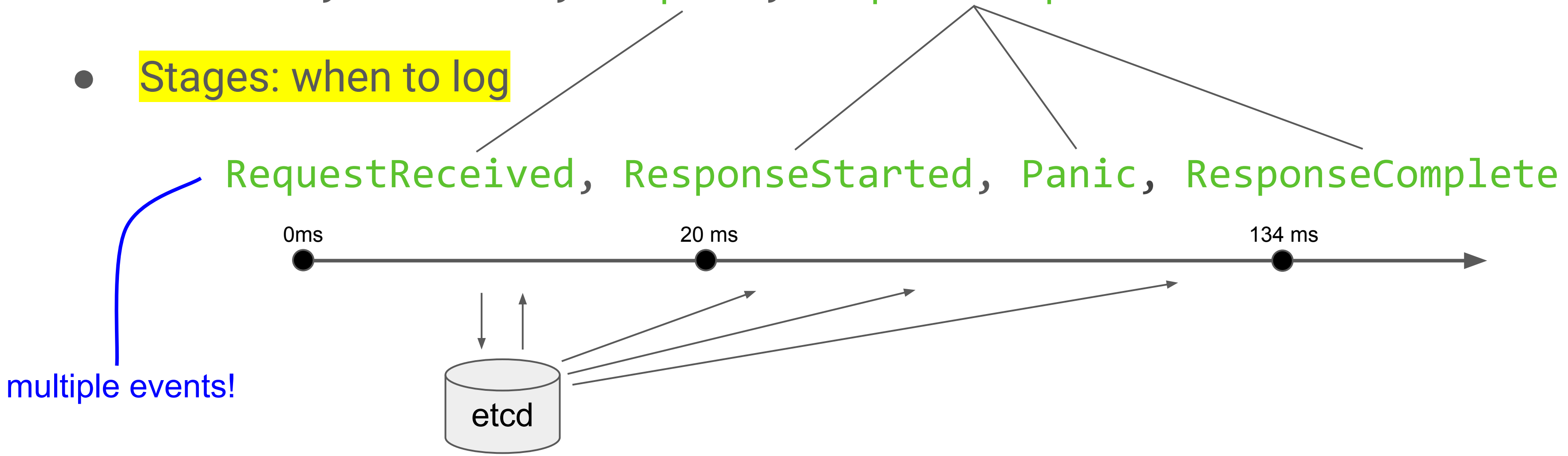
https://unsplash.com

Performance impact vs. consistency

- Levels: how deep to log

`None, MetaData, Request, RequestRespone`

- Stages: when to log



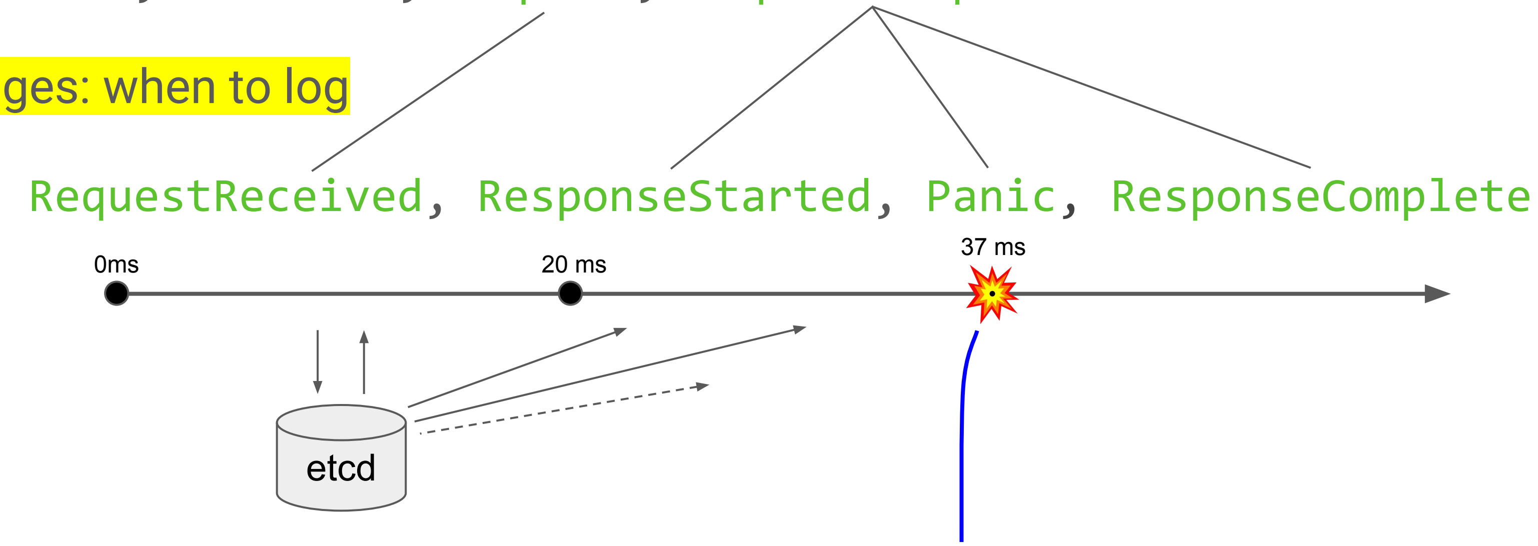
`RequestReceived, ResponseStarted, Panic, ResponseComplete`

Performance impact vs. consistency

- Levels: how deep to log

  `None, MetaData, Request, RequestRespone`

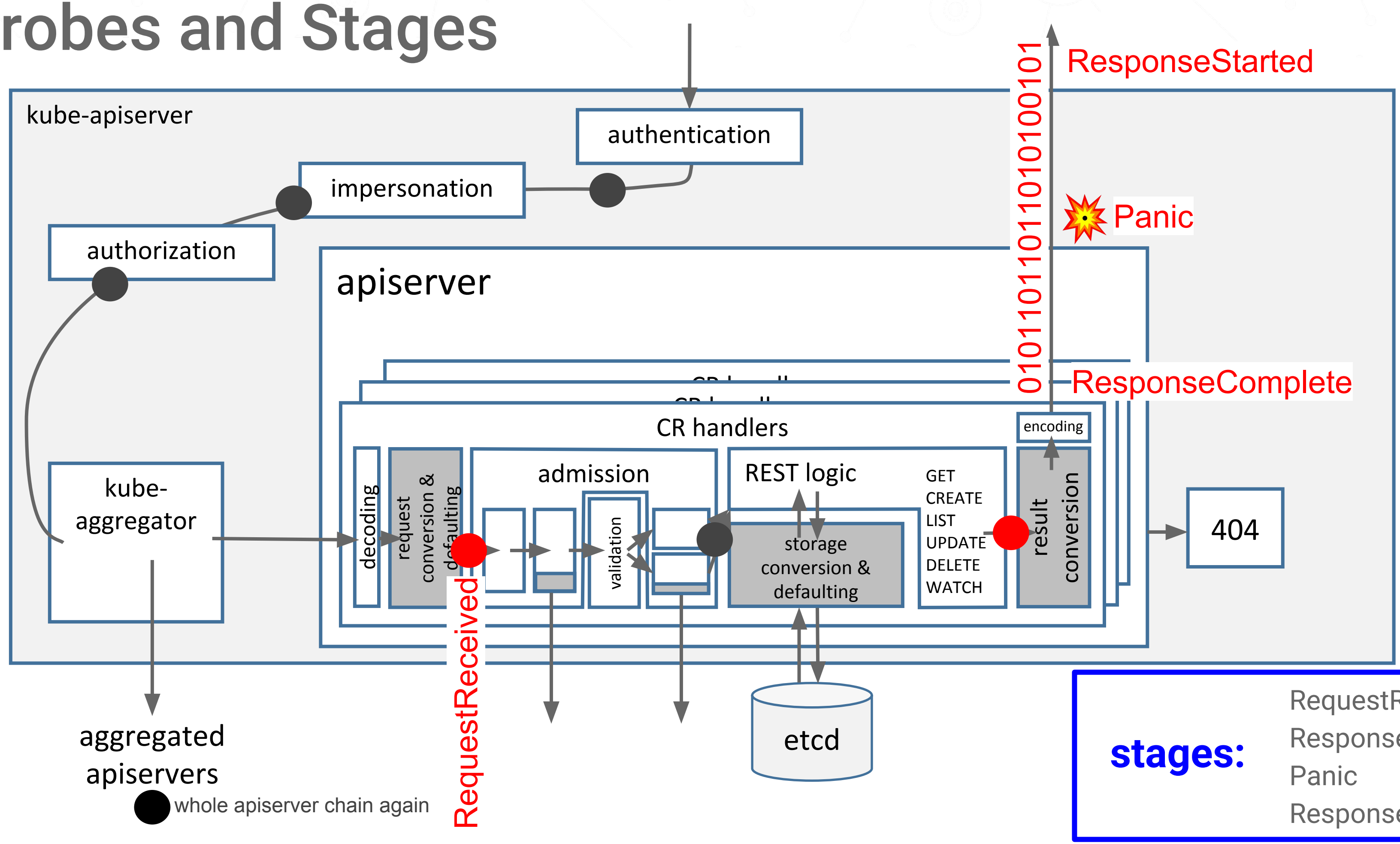
- Stages: when to log



`RequestReceived, ResponseStarted, Panic, ResponseComplete`

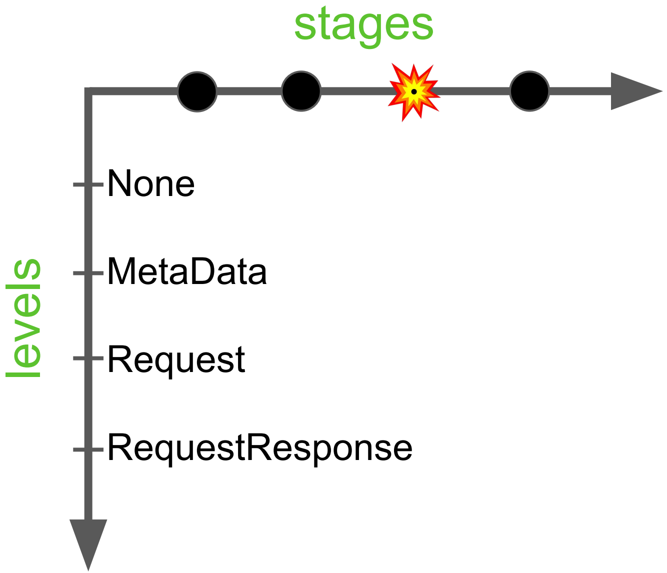
due to bugs, timeouts, ...

Probes and Stages



stages:

RequestReceived
ResponseStarted
Panic
ResponseComplete


stages
levels
None
MetaData
Request
RequestResponse

Defining a policy

kube-apiserver

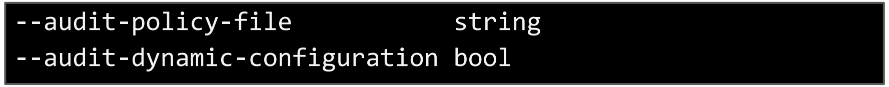
--audit-policy-file string

Path to the audit policy configuration.

--audit-dynamic-configuration bool

v1alpha1 in 1.13

Enables dynamic audit configuration.

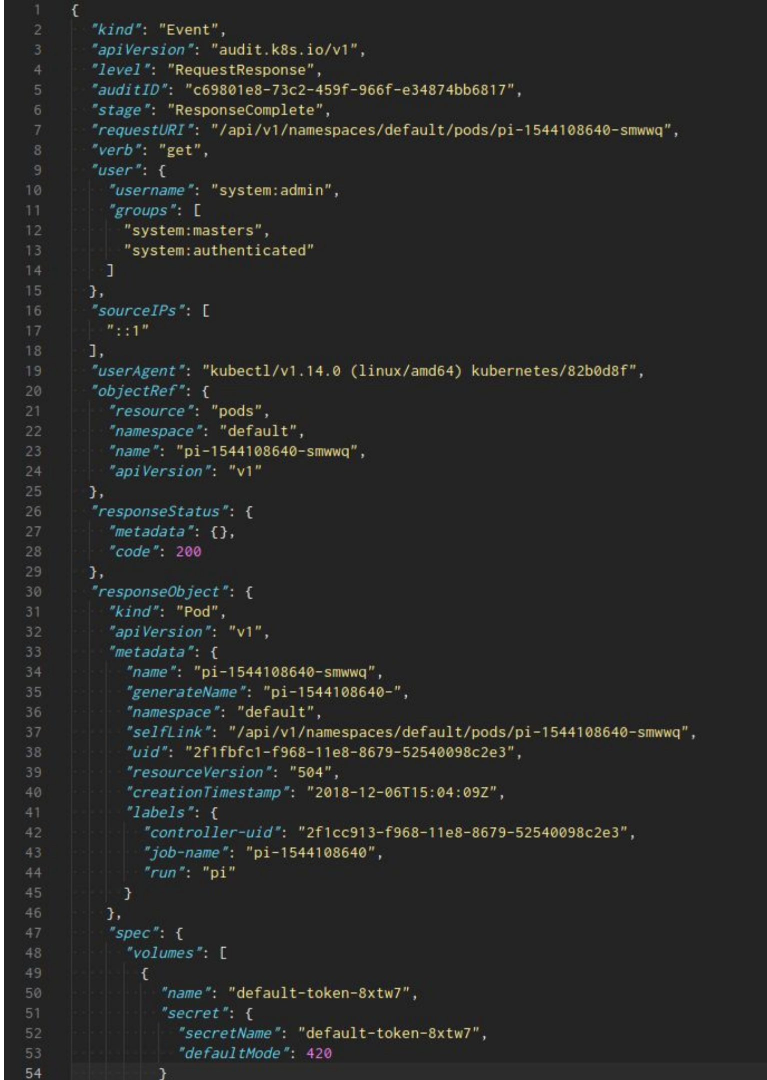


```
--audit-policy-file             string
--audit-dynamic-configuration  bool
```

```yaml
apiVersion: audit.k8s.io/v1
kind: Policy
omitStages:
  - "RequestReceived"
rules:
  - level: “None”
    ...
```

Deep object logging

```yaml
- level: RequestResponse
  resources:
  - group: "" # core
  - group: "apps"
  omitStages:
  - RequestReceived
```



```json
{
  "kind": "Event",
  "apiVersion": "audit.k8s.io/v1",
  "level": "RequestResponse",
  "auditID": "c69801e8-73c2-459f-966f-e34874bb6817",
  "stage": "ResponseComplete",
  "requestURI": "/api/v1/namespaces/default/pods/pi-1544108640-smwwq",
  "verb": "get",
  "user": {
    "username": "system:admin",
    "groups": [
      "system:masters",
      "system:authenticated"
    ]
  },
  "sourceIPs": [
    "::1"
  ],
  "userAgent": "kubectl/v1.14.0 (linux/amd64) kubernetes/82b0d8f",
  "objectRef": {
    "resource": "pods",
    "namespace": "default",
    "name": "pi-1544108640-smwwq",
    "apiVersion": "v1"
  },
  "responseStatus": {
    "metadata": {},
    "code": 200
  },
  "responseObject": {
    "kind": "Pod",
    "apiVersion": "v1",
    "metadata": {
      "name": "pi-1544108640-smwwq",
      "generateName": "pi-1544108640-",
      "namespace": "default",
      "selfLink": "/api/v1/namespaces/default/pods/pi-1544108640-smwwq",
      "uid": "2f1fbfc1-f968-11e8-8679-52540098c2e3",
      "resourceVersion": "504",
      "creationTimestamp": "2018-12-06T15:04:09Z",
      "labels": {
        "controller-uid": "2f1cc913-f968-11e8-8679-52540098c2e3",
        "job-name": "pi-1544108640",
        "run": "pi"
      }
    },
    "spec": {
      "volumes": [
        {
          "name": "default-token-8xtw7",
          "secret": {
            "secretName": "default-token-8xtw7",
            "defaultMode": 420
          }
```

Excluding secrets

```
- level: Metadata
  resources:
  - group: "" # core
    resources: ["secrets", "configmaps"]
  - group: authentication.k8s.io
    resources: ["tokenreviews"]
  omitStages:
  - RequestReceived
```

Logging objects at different levels

```
- level: Request
  verbs: ["get", "list", "watch"]
  resources:
  - group: "batch"
  omitStages:
  - RequestReceived
- level: RequestResponse
  resources:
  - group: "batch"
  omitStages:
  - RequestReceived
```

```
- level: None
  nonResourceURLs:
  - /healthz*
  - /version
  - /swagger*
```

Logging events performed by a particular user

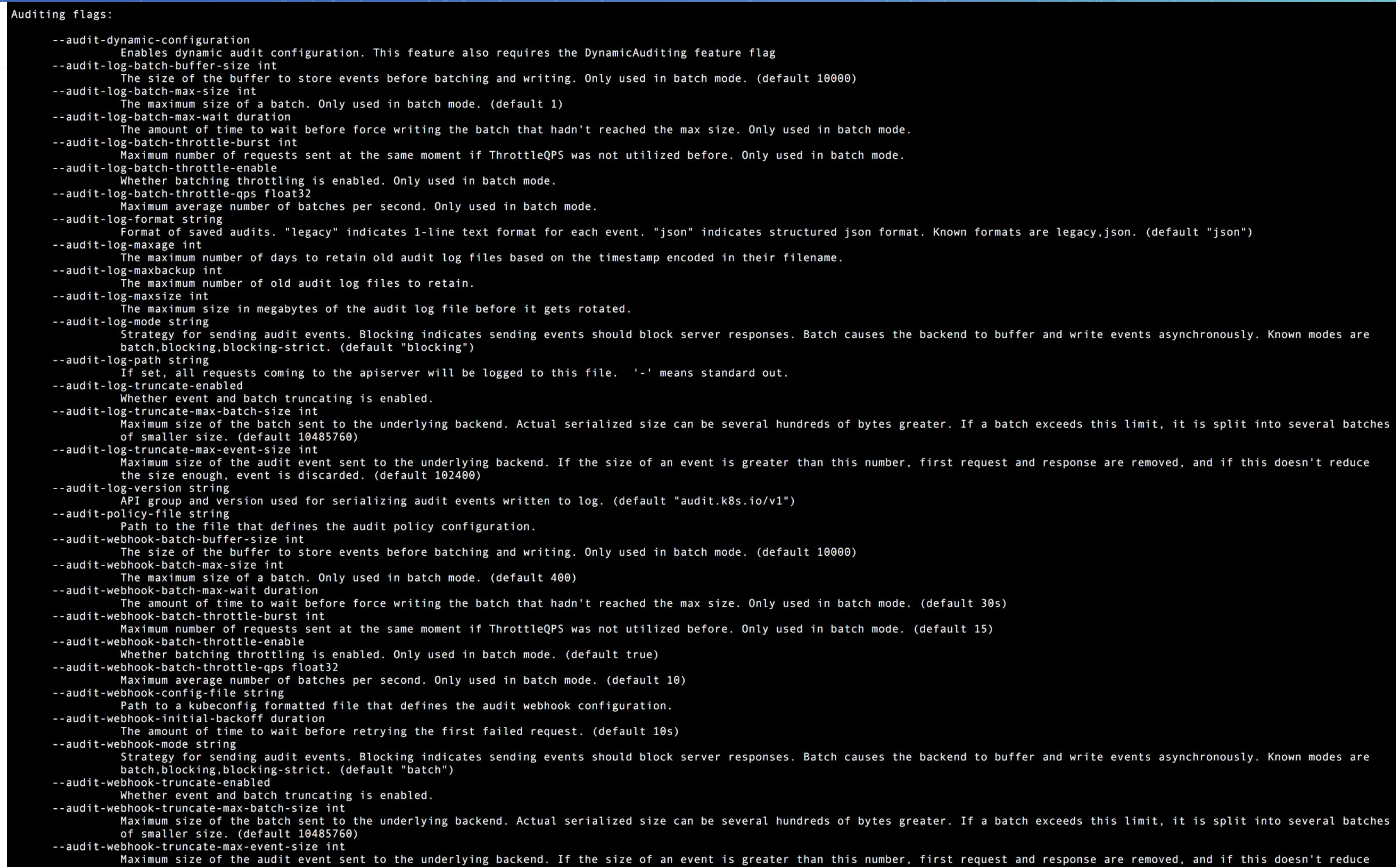
```
- level: RequestResponse
  users: ["naughtyuser"]
  omitStages:
    - RequestReceived
```

Integrating with your infrastructure

https://unsplash.com

Config

kube-apiserver



```
Auditing flags:

      --audit-dynamic-configuration
                Enables dynamic audit configuration. This feature also requires the DynamicAuditing feature flag
      --audit-log-batch-buffer-size int
                The size of the buffer to store events before batching and writing. Only used in batch mode. (default 10000)
      --audit-log-batch-max-size int
                The maximum size of a batch. Only used in batch mode. (default 1)
      --audit-log-batch-max-wait duration
                The amount of time to wait before force writing the batch that hadn't reached the max size. Only used in batch mode.
      --audit-log-batch-throttle-burst int
                Maximum number of requests sent at the same moment if ThrottleQPS was not utilized before. Only used in batch mode.
      --audit-log-batch-throttle-enable
                Whether batching throttling is enabled. Only used in batch mode.
      --audit-log-batch-throttle-qps float32
                Maximum average number of batches per second. Only used in batch mode.
      --audit-log-format string
                Format of saved audits. "legacy" indicates 1-line text format for each event. "json" indicates structured json format. Known formats are legacy,json. (default "json")
      --audit-log-maxage int
                The maximum number of days to retain old audit log files based on the timestamp encoded in their filename.
      --audit-log-maxbackup int
                The maximum number of old audit log files to retain.
      --audit-log-maxsize int
                The maximum size in megabytes of the audit log file before it gets rotated.
      --audit-log-mode string
                Strategy for sending audit events. Blocking indicates sending events should block server responses. Batch causes the backend to buffer and write events asynchronously. Known modes are
                batch,blocking,blocking-strict. (default "blocking")
      --audit-log-path string
                If set, all requests coming to the apiserver will be logged to this file.  '-' means standard out.
      --audit-log-truncate-enabled
                Whether event and batch truncating is enabled.
      --audit-log-truncate-max-batch-size int
                Maximum size of the batch sent to the underlying backend. Actual serialized size can be several hundreds of bytes greater. If a batch exceeds this limit, it is split into several batches
                of smaller size. (default 10485760)
      --audit-log-truncate-max-event-size int
                Maximum size of the audit event sent to the underlying backend. If the size of an event is greater than this number, first request and response are removed, and if this doesn't reduce
                the size enough, event is discarded. (default 102400)
      --audit-log-version string
                API group and version used for serializing audit events written to log. (default "audit.k8s.io/v1")
      --audit-policy-file string
                Path to the file that defines the audit policy configuration.
      --audit-webhook-batch-buffer-size int
                The size of the buffer to store events before batching and writing. Only used in batch mode. (default 10000)
      --audit-webhook-batch-max-size int
                The maximum size of a batch. Only used in batch mode. (default 400)
      --audit-webhook-batch-max-wait duration
                The amount of time to wait before force writing the batch that hadn't reached the max size. Only used in batch mode. (default 30s)
      --audit-webhook-batch-throttle-burst int
                Maximum number of requests sent at the same moment if ThrottleQPS was not utilized before. Only used in batch mode. (default 15)
      --audit-webhook-batch-throttle-enable
                Whether batching throttling is enabled. Only used in batch mode. (default true)
      --audit-webhook-batch-throttle-qps float32
                Maximum average number of batches per second. Only used in batch mode. (default 10)
      --audit-webhook-config-file string
                Path to a kubeconfig formatted file that defines the audit webhook configuration.
      --audit-webhook-initial-backoff duration
                The amount of time to wait before retrying the first failed request. (default 10s)
      --audit-webhook-mode string
                Strategy for sending audit events. Blocking indicates sending events should block server responses. Batch causes the backend to buffer and write events asynchronously. Known modes are
                batch,blocking,blocking-strict. (default "batch")
      --audit-webhook-truncate-enabled
                Whether event and batch truncating is enabled.
      --audit-webhook-truncate-max-batch-size int
                Maximum size of the batch sent to the underlying backend. Actual serialized size can be several hundreds of bytes greater. If a batch exceeds this limit, it is split into several batches
                of smaller size. (default 10485760)
      --audit-webhook-truncate-max-event-size int
                Maximum size of the audit event sent to the underlying backend. If the size of an event is greater than this number, first request and response are removed, and if this doesn't reduce
```

How to send audit events

stdout

--audit-log-path {-,some-file-name}
--audit-webhook-config-file <kubeconfig>

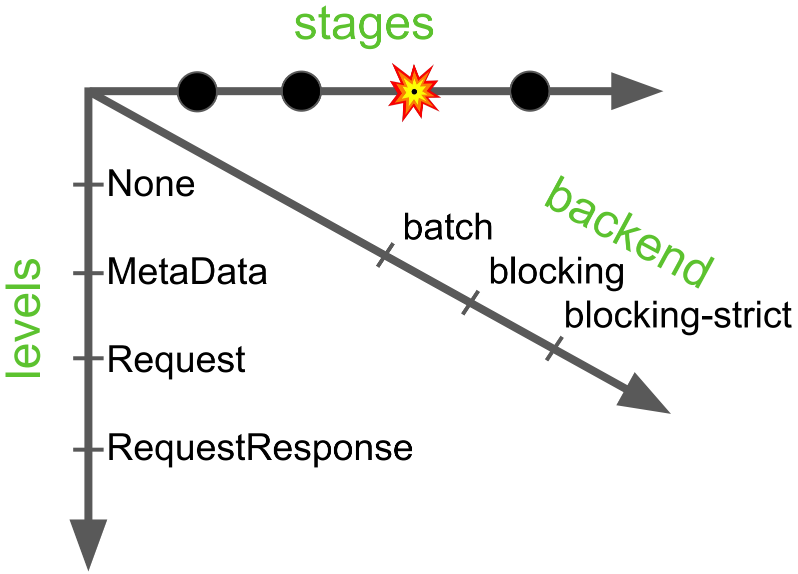
--audit-{log,webhook}**-mode** string

Strategy for sending audit events. Blocking indicates sending events should block server responses. Batch causes the backend to buffer and write events asynchronously. Known modes are: **batch**, **blocking**, **blocking-strict**. (default: "blocking" for log, "batch" for webhook)

```
--audit-{log,webhook}-batch-buffer-size int   (default: 10000 events)
--audit-{log,webhook}-batch-max-size int      (default: 400 events)
--audit-{log,webhook}-batch-max-wait int      (default: 30s)
```

Note: on shutdown, we gracefully flush audit events

Performance vs. consistency



Dynamic Audit Configuration

v1alpha1 in 1.13

```
kube-apiserver
  --audit-dynamic-configuration
  --feature-gates DynamicAuditing=true
  --runtime-config auditregistration.k8s.io/v1alpha=true
```

while alpha

```
apiVersion: auditregistration.k8s.io/v1alpha1
kind: AuditSink
metadata:
  name: <name>
policy:
  level: None/Metadata/Request/RequestResponse
  stages:
  - RequestReceived/ResponseStarted/ResponseComplete
webhook:
  clientConfig:
    url: <backend url>
    service: <optional service name>
    caBundle: <ca bundle>
  throttle: ...
```

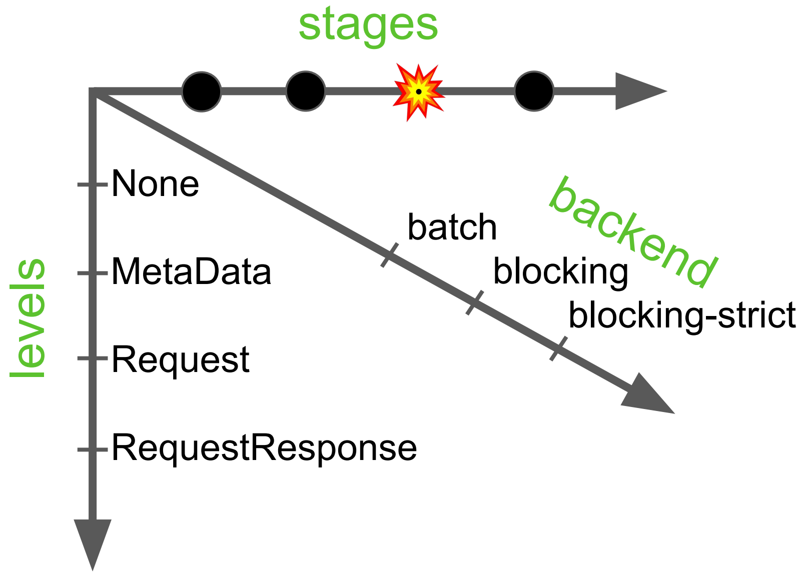
References

kubernetes.io/docs/tasks/debug-application-cluster/audit

kubernetes/community/contributors/design-proposals/api-machinery/auditing.md

kubernetes/enhancements/keps/sig-auth/0014-dynamic-audit-configuration.md

github.com/liggitt/audit2rbac



Backend options:

- **log**
- **webhook** via kubeconfig
- soon: webhook via **AuditSink** resource

```
apiVersion: audit.k8s.io/v1
kind: Policy
omitStages:
- "RequestReceived"
rules:
- level: “None”
  ...
```