

Observability Without Overload: Standardizing Metrics and Logs in Kubernetes

Container Days London - 12th of February 2026
Diana Todea - DevEx Engineer





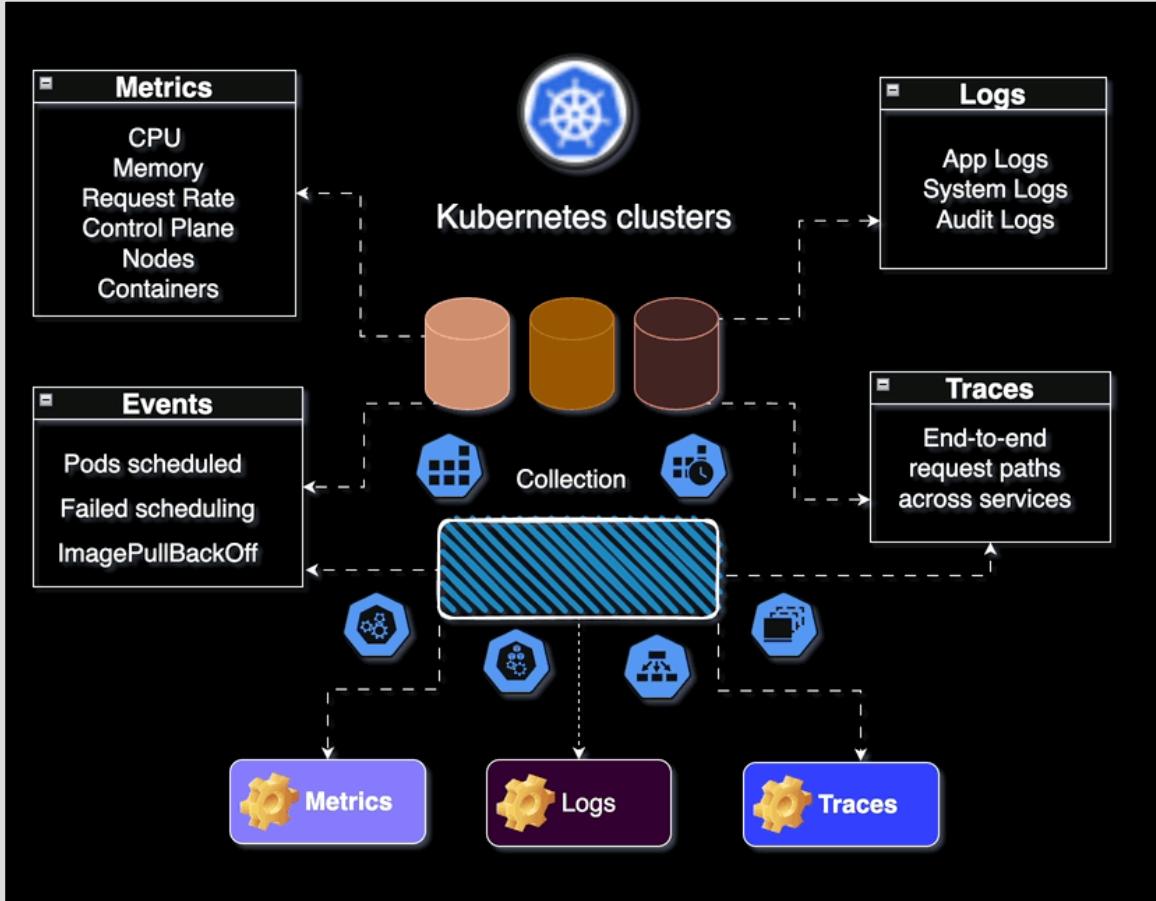
OpenTelemetry member and contributor

Cloud Native Days Romania organizer

Co-lead CNCF Merge-Forward Neurodiversity



A dark-themed profile card for Diana Todea. At the top is a circular photo of a woman with short brown hair, wearing a light-colored shirt. Below the photo is her name, "Diana Todea". Underneath her name are two small icons: a yellow location pin and a blue gear, followed by the text "DX@VictoriaMetrics | Observability SME | OpenTelemetry docs localizatio...". A large QR code is centered below the text area.



Kubernetes Metrics (v1.35)

- Prometheus-native metrics exposed by core Kubernetes components (API server, scheduler, kubelet, controllers)
- Stable metrics (production-safe API)
 - API server: request rate, latency, errors, admission control, deprecated API usage
 - Scheduler: queue depth, scheduling latency, retries, preemption
 - Workloads & nodes: CPU/memory usage (container/pod/node), resource requests & limits
 - Controllers: Job and CronJob execution and sync behavior
- Beta metrics (evolving, backward-compatible)
 - API Priority & Fairness, authn/authz config reloads
 - Validation (CEL), feature gates, probes, image volumes
 - End-to-end pod scheduling latency (SLI)
- Alpha metrics (experimental, no guarantees)
 - Internal and discovery-related behaviors
- Reference: <https://github.com/kubernetes-monitoring/kubernetes-mixin> contains recording rules, dashboards, alerts

Instrumentation pain points

- Related to metrics in Kubernetes, the SIG-Instrumentation currently needs help with:
 - *Guide on how to interpret metrics/dashboards. Better alerts, dashboards. Group metrics by severity.
 - *Find ways how to keep the list updated, metrics visible. They have an automated way to generate the page from kubernetes source code. Need help including other projects (i.e. autoscaler) in this doc or building similar pages for these projects.

Kubernetes Metrics Reference

Details of the metric data that Kubernetes components export.

Metrics (v1.35)

This page details the metrics that different Kubernetes components export. You can query the metrics endpoint for these components using an HTTP scrape, and fetch the current metrics data in Prometheus format.

List of Stable Kubernetes Metrics

Stable metrics observe strict API contracts and no labels can be added or removed from stable metrics during their lifetime.

`apiserver_admission_controller_admission_duration_seconds`

Admission controller latency histogram in seconds, identified by name and broken out for each operation and API resource and type (validate or admit).

- **Stability Level:** STABLE

- **Type:** Histogram

- **Labels:** `name` `operation` `rejected` `type`

`apiserver_admission_step_admission_duration_seconds`

Admission sub-step latency histogram in seconds, broken out for each operation and API resource and step type (validate or admit).

- **Stability Level:** STABLE

- **Type:** Histogram

Quiz time

Kubernetes Logs (v1.35)

- Application/Container logs
 - Formats: Plain text or JSON (application-defined)
 - Examples: Plain
- INFO Server started on port 8080; JSON: {"level":"info","msg":"request handled","status":200}
- Control plane & node logs
(kube-apiserver, scheduler, controller-manager, kubelet, etcd)
 - Formats: Structured JSON (default), plain text (optional)
 - Examples: JSON
- {"ts":"2026-01-10T10:21:33Z","level":"info","component":"kube-apiserver","msg":"Request completed","code":200}

Kubernetes Logs (v1.35) continued

- [Container runtime logs](#) (containerd, CRI-O)

- Formats: Plain text or JSON

```
"2019-04-30T02:12:41.8443515Z stdout F message"
```

- Kubernetes Events (log-like)

- Formats: Structured objects (JSON / YAML)

```
{"reason": "FailedScheduling", "message": "0/3 nodes available", "type": "Warning"}
```

- Key takeaway:

System and audit logs are structured JSON by default, while application logs are free-form, making format standardization an application responsibility.

Instrumentation pain points

- Related to logs in Kubernetes, the SIG-Instrumentation currently needs help with:
*Structured logging: <https://kubernetes.io/docs/concepts/cluster-administration/system-logs/#structured-logging>

```
I1025 00:15:15.525108 1 controller_utils.go:116] "Pod status updated" pod="kube-system/kubedns" status="ready"
```

Instrumentation pain points (continued)

- Related to logs in Kubernetes, the SIG-Instrumentation currently needs help with:
 - *Define the standard for logging - propose libraries, interfaces, metadata schema
 - *Reduce friction for using logging - reduce dependencies and performance overhead
 - *Give more choice over logging - allow pluggable logging implementation
 - *Ensure quality consistent logging - overview migration, create documentation, tooling and educate reviewers
 - *Prevent regressions caused by logging - measure performance overhead and log volume changes

Quiz time

Kubernetes media

- EVENTS

- *often missed Kubernetes events to various outputs so that they can be used for observability or alerting purposes: <https://github.com/resmoio/kubernetes-event-exporter>

- AUDIT LOG (API server)

- Formats: JSON only (strict schema)

```
{"kind": "Event", "level": "Metadata", "verb": "create", "user": {"username": "admin"}, "objectRef": {"resource": "pods"}}
```

- *Kubernetes audit log files: <https://kubernetes.io/docs/tasks/debug/debug-cluster/audit/>

Demo

*Audit Logs Stats app parses Kubernetes audit log files and sends them to [VictoriaLogs](#).

A Grafana dashboard is used to render statistics derived from these logs, helping to identify noisy applications or requests that take too much time.

Source code: <https://github.com/vrutkovs/audit-log-stats>



Home > Dashboards > Audit log

Search... ⌘+k

Edit Export Share

ds VictoriaLogs Username All Verb All Resource All Namespace .* Subresource (^)(status)

Last 7 days Refresh

Matched audit log events

```
:
: 2026-01-12 09:23:15.095 missing _msg field; see https://docs.victoriametrics.com/victorialogs/keyconcepts/#message-field
: 2026-01-12 09:23:15.035 missing _msg field; see https://docs.victoriametrics.com/victorialogs/keyconcepts/#message-field
: 2026-01-12 09:23:15.035 missing _msg field; see https://docs.victoriametrics.com/victorialogs/keyconcepts/#message-field
: 2026-01-12 09:23:15.033 missing _msg field; see https://docs.victoriametrics.com/victorialogs/keyconcepts/#message-field
: 2026-01-12 09:23:15.028 missing _msg field; see https://docs.victoriametrics.com/victorialogs/keyconcepts/#message-field
: 2026-01-12 09:23:14.972 missing _msg field; see https://docs.victoriametrics.com/victorialogs/keyconcepts/#message-field
: 2026-01-12 09:23:14.925 missing _msg field; see https://docs.victoriametrics.com/victorialogs/keyconcepts/#message-field
: 2026-01-12 09:23:14.901 missing _msg field; see https://docs.victoriametrics.com/victorialogs/keyconcepts/#message-field
: 2026-01-12 09:23:14.899 missing _msg field; see https://docs.victoriametrics.com/victorialogs/keyconcepts/#message-field
: 2026-01-12 09:23:14.872 missing _msg field; see https://docs.victoriametrics.com/victorialogs/keyconcepts/#message-field
```

Latency

Slow operations per username

Slow operations per username

33.3 mins
29.2 mins
25 mins
20.8 mins
16.7 mins
12.5 mins
8.33 mins
4.17 mins
0 ns

01/10 00:00 01/11 00:00 01/12 00:00 01/13 00:00 01/14 00:00 01/15 00:00 01/16 00:00

sum_total{user.username="system:admin"} sum_total{user.username="system:apiserver"}
sum_total{user.username="system:serviceaccount:e2e-disruption-monitor-gvkfp:disruption-monitor-sa"}
sum_total{user.username="system:serviceaccount:e2e-disruption-monitor-v9xnp:disruption-monitor-sa"}

Long etcd operations per username

Long etcd operations per username

26.7 mins
23.3 mins
20 mins
16.7 mins
13.3 mins
10 mins
6.67 mins
3.33 mins
0 ns

01/10 00:00 01/11 00:00 01/12 00:00 01/13 00:00 01/14 00:00 01/15 00:00 01/16 00:00

sum_total{user.username="system:admin"} sum_total{user.username="system:apiserver"}
sum_total{user.username="system:serviceaccount:openshift-authentication-operator:authentication-operator"}
sum_total{user.username="system:serviceaccount:openshift-cluster-version:default"}

APF queue wait per username

APF queue wait per username

1.20 s

Large write operations per username

Large write operations per username

1.20 s

Quiz time

Advice for new developers

- Structured logs
- Instrumentation from day zero
- Focus on alerts! Which alerts would you like to see? This will dictate which metrics do you want to have. Dashboards are tied to alerts.



Contributing to K8s SIG-instrumentation

- Further contribution to SIG-instrumentation is needed for end users:
 - *<https://github.com/kubernetes/website/issues/53168>
 - *Adding the contributions to SIG-instrumentation
 - *Once SIG-instrumentation finalizes work
 - *Create tutorials, documentation for end-users

Improve K8s instrumentation DevEx

- One idea would be to classify the metrics for particular use cases ranging from beginner, intermediate to advanced.

Kubernetes Metrics Reference

Details of the metric data that Kubernetes components export.

Metrics (v1.35)

This page details the metrics that different Kubernetes components export. You can query the metrics endpoint for these components using an HTTP scrape, and fetch the current metrics data in Prometheus format.

List of Stable Kubernetes Metrics

Stable metrics observe strict API contracts and no labels can be added or removed from stable metrics during their lifetime.

`apiserver_admission_controller_admission_duration_seconds`

Admission controller latency histogram in seconds, identified by name and broken out for each operation and API resource and type (validate or admit).

- **Stability Level:** STABLE
- **Type:** Histogram
- **Labels:** `name` `operation` `rejected` `type`

`apiserver_admission_step_admission_duration_seconds`

Admission sub-step latency histogram in seconds, broken out for each operation and API resource and step type (validate or admit).

- **Stability Level:** STABLE
- **Type:** Histogram

Next steps

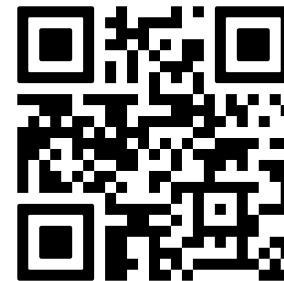
- TRACING
 - *<https://kubernetes.io/docs/concepts/cluster-administration/system-traces/>
- Benefits of tracing: vendor-neutral standard, common tools to work with
- Currently tracing is grouped by stability, but we also want them grouped by criticality
- Definitely something that we want SIG-Instrumentation to focus on!



Quiz time

Community

- Big thanks to [@valyala](#), [@vrutkovs](#), [@bwplotka](#), [@dashpole](#)



VictoriaLogs

Resources

<https://kubernetes.io/docs/reference/instrumentation/metrics/>

<https://kubernetes.io/docs/concepts/cluster-administration/system-logs/#structured-logging>

<https://github.com/kubernetes/community/sig-instrumentation>

<https://github.com/resmoio/kubernetes-event-exporter>

<https://github.com/kubernetes-monitoring/kubernetes-mixin>

<https://github.com/vrutkovs/audit-log-stats>

<https://github.com/VictoriaMetrics/VictoriaLogs>

Kubernetes Slack: #sig-instrumentation



Learn more



community.cncf.io/merge-forward

Creating diverse, supportive communities and ally networks for shared learning, mentorship, friendship, and collaborative idea exchange.

#merge-forward on the
CNCF Slack!

Thank you!

Bsky: @didiviking.bsky.social

X: @dianavtodea

Github: @didiViking/Conferences_Talks

LinkedIn: @diana-todea-b2a79968

