

From Monitoring to Observability: Left Shift your SLOs with Chaos



Michael Friedrich
Senior Developer Evangelist at GitLab



EveryoneCanContribute.com



KubeCon



CloudNativeCon

Europe 2022

An SRE Tale



KubeCon



CloudNativeCon

Europe 2022



Photo by [Michael Friedrich](#)

Turn back time

State blackbox monitoring

Traditional SLA reporting

State changes over time

Metric data points and trends

SLOly adding metrics

Whitebox monitoring

Prometheus /metrics in Docker in 2016

Add metrics output to docker #25820

Merged

thaJeztah merged 1 commit into moby:master from crosbymichael:prom on 28 Oct 2016

<https://github.com/moby/moby/pull/25820>

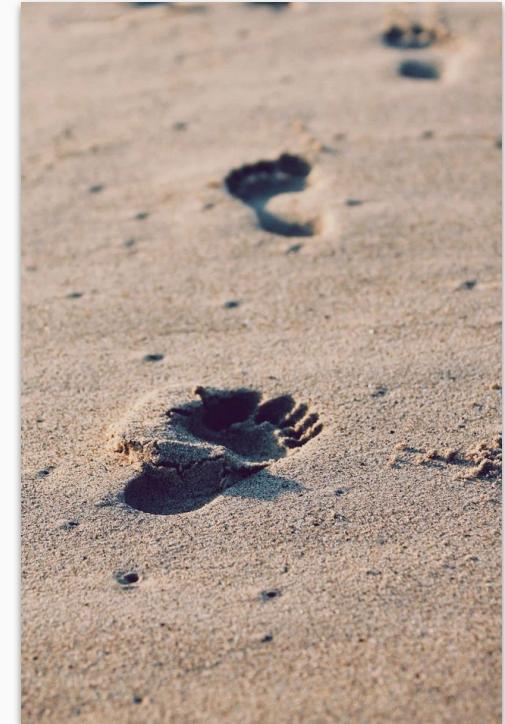


Photo by Christopher Sardegna on [Unsplash](#)

Moving into SL*

Service Level

Agreement - 99.5% availability

Objective - 99.9% availability

Indicator - errors, latency, ...

Error budgets

<https://landing.google.com/sre/workbook/chapters/implementing-slos/>

<https://engineering.bitnami.com/articles/implementing-slos-using-prometheus.html>

<https://grafana.com/blog/2019/11/27/kubecon-recap-how-to-include-latency-in-slo-based-alerting/>

<https://github.com/google/prometheus-slo-burn-example/blob/master/prometheus/slos.rules.yml>

<https://github.com/prometheus/prometheus/issues/6209>

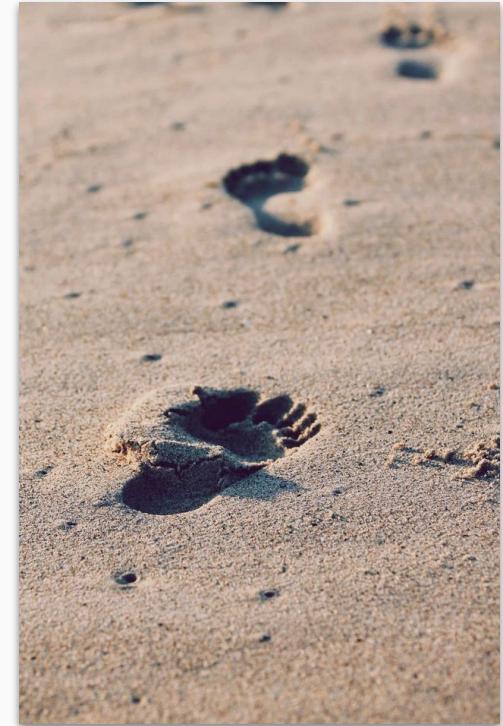


Photo by Christopher Sardegna on [Unsplash](#)

Golden Signals

Latency

Traffic

Errors

Saturation

Code instrumentation needed

SRE solved everything?



A Developer's Tale

Story time: Development gone wrong



KubeCon



CloudNativeCon

Europe 2022



Photo by [Michael Friedrich](#)

Devs building software

🔥 Slow REST & JSON-RPC API response

CPU overload with threads

💡 Go has light-weight goroutines

🤔 Co-routines in C++

Stackless

Function ptr on the heap

Stack unwinding continuation

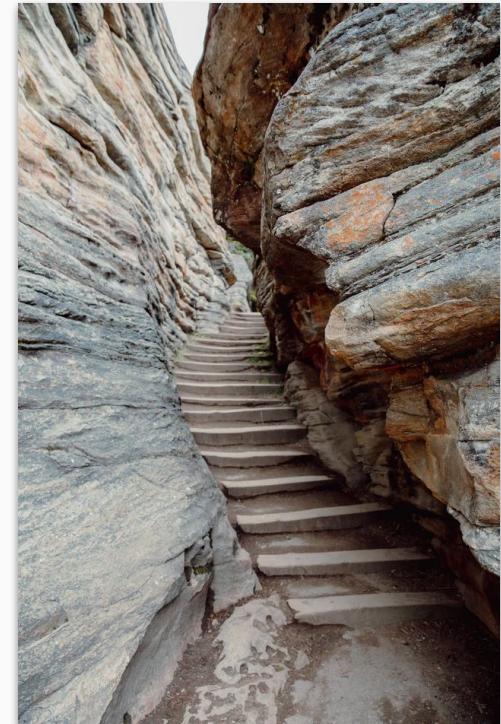


Photo by [Moriah Wolfe](#) on [Unsplash](#)

Devs debugging software

Crash happens in production

1000+ API clients

Memory

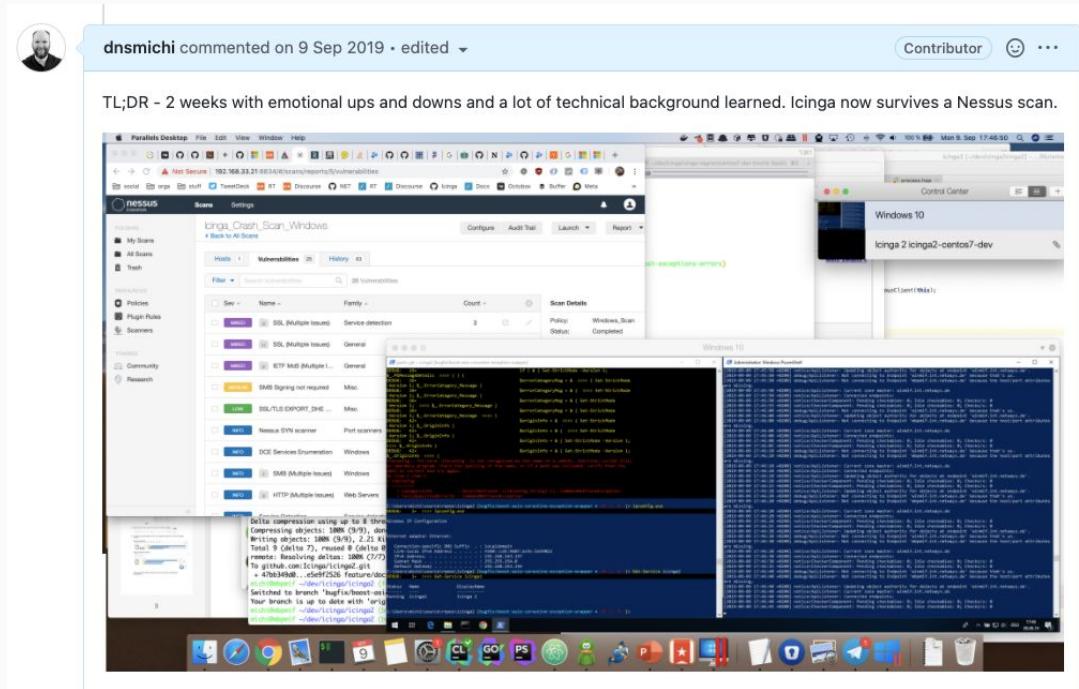
corruption

exhausted - leaks?

Windows specific crashes

Stack guards

Nessus security scanner



Screenshot by Michael Friedrich

Retrospective: Devs need to go SLO

Corrective action

Stack/heap memory meets Ops requirements



SLI

mem_usage level



SLO

x < 10% increase



Metrics monitoring in CI/CD merge requests / production



Chaos engineering/Fuzzing for API requests



An Ops tale

Story time: It's always DNS



KubeCon



CloudNativeCon

Europe 2022



Photo by [Michael Friedrich](#)

DNSSEC with .at

Turn back time to 2011

.at ccTLD got DNSSEC

Signing hardware

State machine of steps

Friday afternoon script change

Deployed to production

No more signing

No DNS updates: Domain delegation

Monitoring?

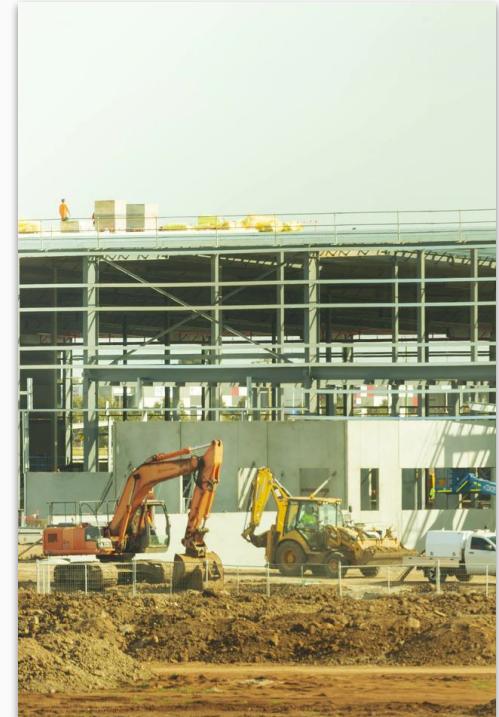


Photo by Troy Mortier on [Unsplash](#)

DNS outdated - who you gonna call

DNS zone serial is Unix timestamp

Serial + offset < now?

1st alarm at 3am via email

2nd alarm: Escalation at 4am via SMS

(from all nameservers, no alert groups)

Xth alarm: Debugging at 5am is not fun

RCA: Change was persisted in Git

Then rolled to production

No CI/CD or quality gates



Photo by [Martin Widenka](#) on [Unsplash](#)

Retrospective: DNS outdated corrective actions

Staging signing hardware

Changes rolled with IaC / GitOps

 SLI

zone_serial_age

 SLO

now() - zone_serial_age < 1h

 Chaos engineering with nameservers

Deny zone updates

Return different zone serials



Photo by [Markus Spiske](#) on [Unsplash](#)

A DevOps Tale

Story time: Containers everywhere



KubeCon



CloudNativeCon

Europe 2022



Photo by [Michael Friedrich](#)

Rate limits for container image pulls

Turn back time to Sep 2020

Docker announced Hub rate limits

Possibly affected

CI/CD pipelines

Cloud native deployments

Organisations behind NAT

Cloud providers



Known state

Limits applied

Values?

Simulate a pull

Header response

Script to parse

Prometheus Exporter

Pull simulation changed to HEAD request

not affecting the remaining count later.



Unknown state

“Docker pull” environments

Kubernetes clusters

CI/CD pipelines

“429 - too many requests” in logs?

App presents new prices only to 33% of customers ...

because release deployment failed ...

because “docker pull” reached the limit?



<https://www.honeycomb.io/blog/so-you-want-to-build-an-observability-tool/>
https://docs.gitlab.com/ee/user/project/clusters/kubernetes_pod_logs.html

Photo by Karen Zhang on Unsplash

Retrospective: SLI and SLO for Docker Limits



SLI

`pull_counts_{remaining,limit}`



SLO

`pull_counts_remaining > 10` (arbitrary number)



Gates

SLO failed: don't start a deployment

Developers need to know why CI/CD and reviews are blocked before the limit is reached. Mitigation: Local registry & dependency proxy



Photo by NOAA on [Unsplash](#)

Go SLO



KubeCon



CloudNativeCon

Europe 2022



Photo by [Michael Friedrich](#)

As a SRE/Dev/Ops/DevOps, where to?

SLO

Monitoring

Metrics

key/tag

Values

?



Photo by Silas Baisch on [Unsplash](#)

Metrics: Prometheus and PromQL

```
# HELP name_counter Count of names
# TYPE name_counter counter
name_counter{name="michael"} 104
name_counter{name="michi"} 1
name_counter{name="niclas"} 1
name_counter{name="niclas"} 16
name_counter{name="olik"} 1
# HELP rocket_http_requests_duration_seconds HTTP request duration in
requests
# TYPE rocket_http_requests_duration_seconds histogram
rocket_http_requests_duration_seconds_bucket{endpoint="/hello/<name>?
<caps>",method="GET",status="200",le="0.005"} 123
```

```
# Latest sample
metric_name

# Range
metric_name[5m]

# Labels
metric_name{label1="a",label2="b"}

# Functions
rate(metric_name[5m])
sum(metric_name)
delta(metric_name[5m])

# Comparisons
metric_name > 10*1024
```

<https://everyonecancontribute.com/post/2021-06-30-cafe-36-rust-rocket-prometheus/>
<https://prometheus.io/docs/prometheus/latest/querying/examples/>
<https://promqls.com/promql-cheat-sheet/>
<https://demo.promlens.com/>

Screenshot from the
#EveryoneCanContribute cafe meetup

Understand metrics sources

Infrastructure

Memory, CPU, IO, ...

Exporter on the node / pod / cluster

Services

Prometheus Exporters

App instrumentation in the code



<https://prometheus.io/docs/instrumenting/exporters/>

<https://training.promlabs.com/>

Photo by Márton Szalai on [Unsplash](#)

Describe SLO with PromQL and Alert Rules

Define the allowed errors in the error budget

Otherwise alert, SLO violated

```
spec:  
  groups:  
    - name: kube-state-metrics  
      rules:  
        - alert: KubeStateMetricsListErrors  
          annotations:  
            description: kube-state-metrics is experiencing errors at an elevated rate in list operations. This is likely causing it to not be able to expose metrics about Kubernetes objects correctly or at all.  
            runbook_url: https://github.com/prometheus-operator/kube-prometheus/wiki/kubestatemetricslisterrors  
            summary: kube-state-metrics is experiencing errors in list operations.  
          expr: |  
            (sum(rate(kube_state_metrics_list_total{job="kube-state-metrics",result="error"}[5m])) /  
             sum(rate(kube_state_metrics_list_total{job="kube-state-metrics"}[5m])))  
            > 0.01  
          for: 15m  
          labels:  
            severity: critical
```

Learn app instrumentation

Learn with playful examples

Dockerfile

CI/CD build image

Container registry

Prometheus Operator

ServiceMonitor CRD

Inspect metrics

Prometheus

OpenTelemetry

```
$ git clone https://github.com/prometheus-operator/kube-prometheus
$ cd kube-prometheus

# 1. Create monitoring namespace and custom resource definitions

$ kubectl create -f manifests/setup

# 2. Wait until the ServiceMonitor CRD is available

$ until kubectl get servicemonitors --all-namespaces ; do date; sleep 1; echo "";
done

# 3. Apply remaining manifests

$ kubectl create -f manifests/
```

```
$ kubectl create -f ./manifests/ecc-python-service.yml
$ kubectl create -f ./manifests/ecc-python-service-monitor.yml
```

https://github.com/prometheus/client_python#three-step-demo

https://gitlab.com/everyonecancontribute/observability/prometheus_python_service



Left shift SLOs



KubeCon



CloudNativeCon

Europe 2022



Photo by [Michael Friedrich](#)

SLOs with Prometheus

Metrics & Alerts calculation

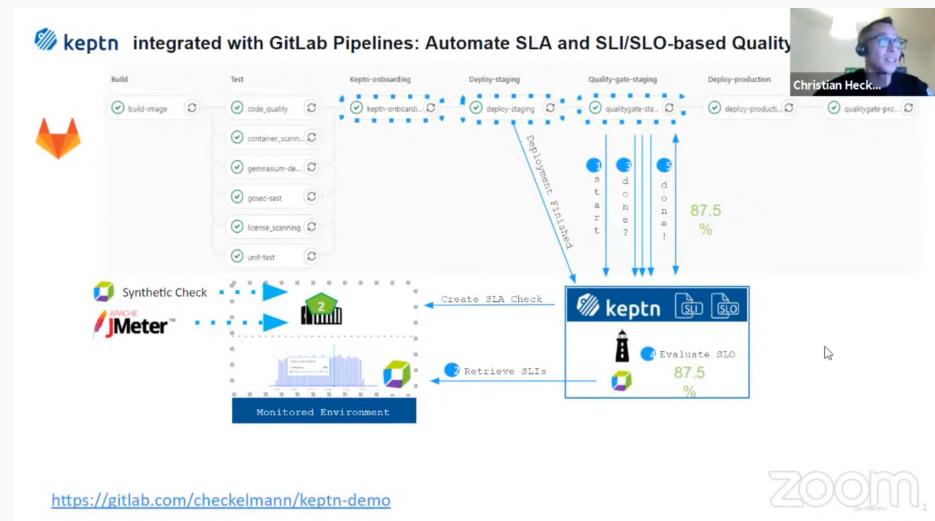
PromQL queries in OpenSLO format

CI/CD

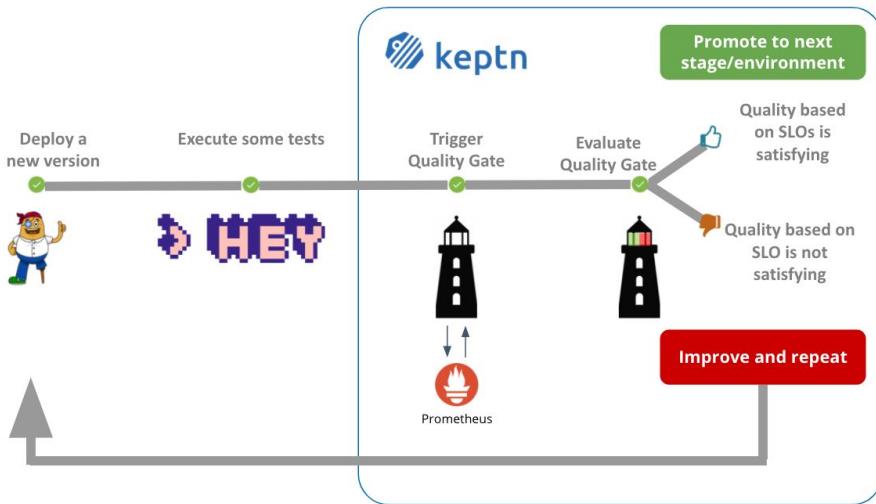
Environments

Metrics Monitoring

Keptn with SLOs for quality gates



Oh Keptn, my Keptn



Screenshot of the Keptn UI showing the "evaluation failed" event details:

evaluation failed

Context: eb1b18ad-2a22-4b9c-aa67-1e6b2e59e605
helloservice

quality-gate

Started today at 11:55 and took 0 seconds
Last task: get-sli

evaluation succeeded

Started today at 11:49 and took 0 seconds
Last task: get-sli

Configure monitoring succeeded

Started today at 11:47 and took 0 seconds
Last task: Configure monitoring

create succeeded

Started today at 11:47 and took 0 seconds
Last task: create

Last time fetched: today at 11:55:29

<https://everyonecancontribute.com/post/2020-11-11-cafe-8-keptn/>

https://keptn.sh/docs/0.9.x/quality_gates/slo/

<https://tutorials.keptn.sh/tutorials/keptn-quality-gates-prometheus-08/#0>

https://www.youtube.com/watch?v=7ksL0V0tN_M

Image credit: [Keptn.sh](https://keptn.sh)

Shift left with quality gates & SLO

Continuous Delivery with

🔒 Keptn as Quality Gate

📈 Prometheus for SLOs

❗ Simulating production (incidents) for apps is hard.

💡 Add chaos to your

Staging environments

Production deployments

🔥 Trigger alerts and SLOs

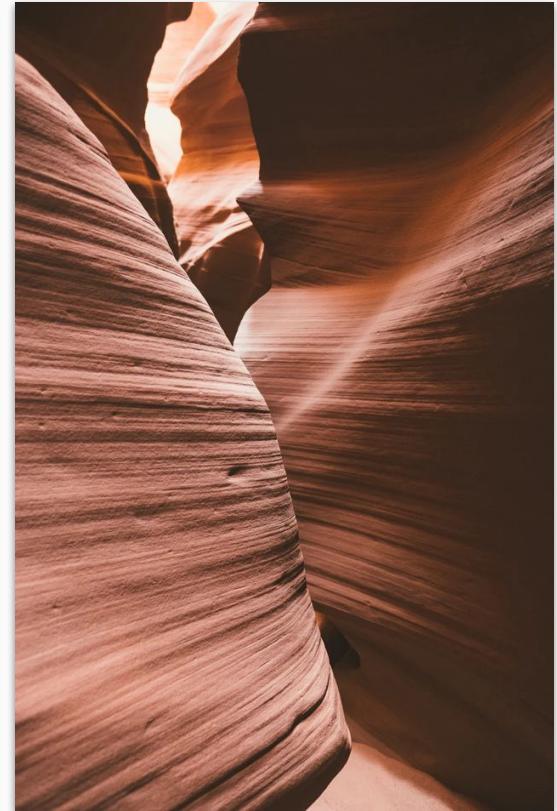


Photo by [Matteo Di Iorio](#) on [Unsplash](#)

Left shift with Chaos



KubeCon



CloudNativeCon

Europe 2022



Photo by [Michael Friedrich](#)

As a SRE/Dev/Ops/DevOps, where to?

Cloud Native

Clusters

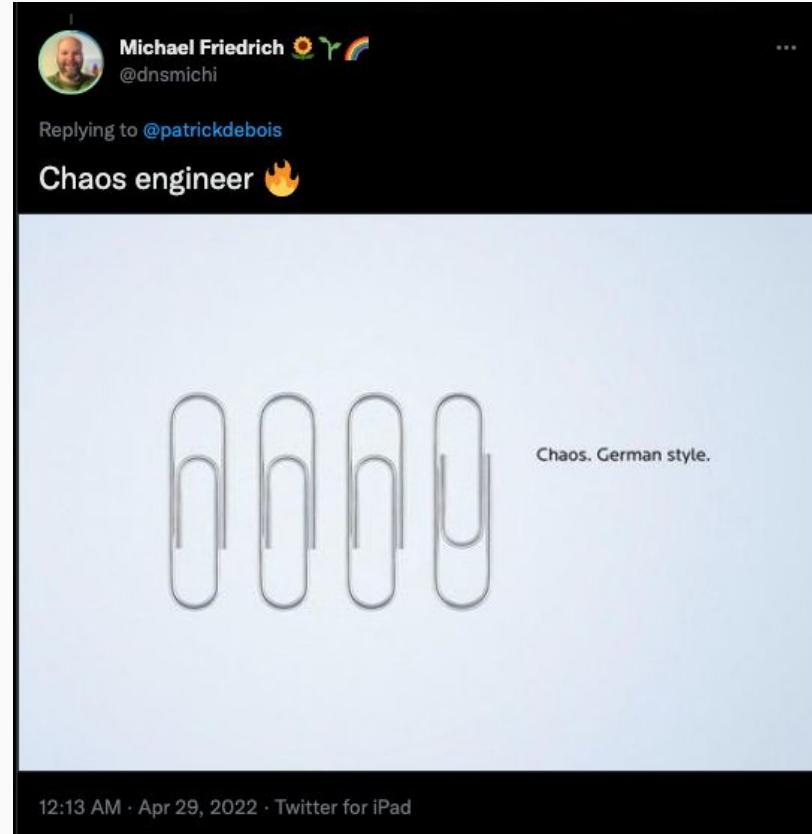
Deployments

Chaos Framework

Experiments

Instrumentation SDKs

<https://o11y.love/topics/chaos-engineering/>



[Tweet source](#)

Litmus Chaos

Fail your infrastructure/cluster

See how the app behaves

See if SLOs still match

Define actions & improvements

Experiments and workflows

The screenshot shows the 'Chaos Center' interface with a purple header bar. On the left is a sidebar with navigation links: Home, Litmus Workflows (which is selected and highlighted in blue), ChaosAgents, ChaosHubs, Analytics, Settings, Usage Statistics, Litmus Docs, Litmus API Docs, and Community. The main content area has a title 'Schedule a new Litmus workflow'. Below it is a horizontal progress bar with steps: 'Choose Agent', 'Choose a workflow', 'Workflow settings', 'Tune workflow' (which is green), 'Reliability score', 'Schedule', 'Back', and 'Next'. A sub-section titled 'Tune the selected workflow' shows a message: 'You have selected a pre-defined workflow Podtato Head. Fine tune the application names and other variables to suit your needs.' It includes a 'Edit Sequence' button and a table for 'Sequence', 'Name', 'Namespace', 'Application', and 'Probes'. The table contains one row: Sequence 1, Name podtato-main-pod-delete-chase, Namespace litmus, Application name-podtato-main, and Probes 1.

The screenshot shows the 'Chaos Center' interface with a purple header bar. On the left is a sidebar with navigation links: Home, Litmus Workflows (selected), ChaosAgents, ChaosHubs, Analytics, Settings, Usage Statistics, Litmus Docs, Litmus API Docs, and Community. The main content area has a title 'Workflow dashboard of podtato-head-1629126291'. It includes tabs for 'Graph View' (selected) and 'Table View'. Below is a graph showing a sequence of events: 'podtato-he-1629126291' (with a circular icon) → 'install-application' (with a checkmark icon) → 'install-chaos-experiments' (with a checkmark icon) → 'pod-delete' (with a circular icon). The status for each step is shown as green with a checkmark.

<https://twitter.com/dnsmichi/status/1427287749065953291>
<https://docs.litmuschaos.io/docs/getting-started/run-your-first-workflow/>

Screenshots by [Michael Friedrich](#)

Chaos Mesh

Fail Kubernetes or Hosts

Run Chaos Experiments

Once

Continuously

Fail DNS as 1st experiment

It's always DNS in prod too

<https://chaos-mesh.org/docs/quick-start/>

<https://chaos-mesh.org/docs/simulate-dns-chaos-on-kubernetes/>

The screenshot shows the Chaos Mesh web interface. On the left is a sidebar with icons for Dashboard, Workflows, Schedules, Experiments (which is selected and highlighted in purple), Events, Archives, and Settings. The main area has a header with 'New experiment', 'Load from', and 'By YAML' buttons, and a search bar. Below this is a section titled 'Experiment Type' with two tabs: 'Kubernetes' (selected) and 'Hosts'. Under 'Kubernetes', there are nine fault types arranged in a 3x3 grid: AWS Fault, DNS Fault, GCP Fault; IO Injection, HTTP Fault, Kernel Fault; Network Attack, Pod Fault, Stress Test; and Clock Skew, JVM Fault. At the bottom are two buttons: 'Error' and 'Random'. Further down are sections for 'Patterns' (with 'o11y+', 'cncf.io', and 'gitlab.com' listed) and 'Affected container names' (with a note about generating container names). A 'Submit' button is at the very bottom.

Screenshot by [Michael Friedrich](#)

Chaos Mesh

UI or CLI

Preview

Scheduling strategies

Preview of Pods to be injected
Checking or unchecking Pods to further limit the scope.

Name	Namespace	IP address	Status
cpp-dns-leaker-service-o11y-55d9498b7f-h5wnp	default	10.42.1.3	Running
cpp-dns-leaker-service-cncf-75d756dcdf-khq26	default	10.42.1.4	Running
cpp-dns-leaker-service-o11y-55d9498b7f-v2kkm	default	10.42.2.5	Running
cpp-dns-leaker-service-gitlab-5ddf58f457-77gjf	default	10.42.2.6	Running

Events

 dns-chaos-example	30 SECONDS AGO
Successfully update records of resource	
 dns-chaos-example	30 SECONDS AGO
Successfully apply chaos for default/cpp-dns-leaker-service-gitlab-5ddf58f457-77gjf/cpp-dns-leaker-service-gitlab	
 dns-chaos-example	30 SECONDS AGO
Successfully apply chaos for default/cpp-dns-leaker-service-cncf-75d756dcdf-khq26/cpp-dns-leaker-service-cncf	
 dns-chaos-example	30 SECONDS AGO
Successfully apply chaos for default/cpp-dns-leaker-service-o11y-55d9498b7f-v2kkm/cpp-dns-leaker-service-o11y	
 dns-chaos-example	31 SECONDS AGO
Successfully apply chaos for default/cpp-dns-leaker-service-o11y-55d9498b7f-h5wnp/cpp-dns-leaker-service-o11y	

<https://opensource.com/article/21/6/chaos-mesh-kubernetes>

<https://medium.com/nerd-for-tech/chaos-engineering-in-kubernetes-using-chaos-mesh-431c1587ef0a>

Screenshot by [Michael Friedrich](#)

Chaos for SRE, Dev, Ops, DevOps Tales

SRE: CPU overload, HTTP requests blocked (golden signals)

Dev: Many API clients not closing the connections

Ops: Intercept DNS traffic, do not resolve

DevOps: Registry proxy limiting container pulls



<https://chaos-mesh.org/docs/simulate-dns-chaos-on-kubernetes/>

Photo by [Markus Spiske](#) on [Unsplash](#)

Your own Chaos

Experiments & SDKs

Integration into CI/CD, tutorials and documentation

Know the limits of Chaos

Evaluate resource usage

Do not harm existing workflows or teams

Chaos Engineering does not solve all reliability issues

It can help with new perspectives 

<https://litmuschaos.github.io/litmus/experiments/categories/contents/>
<https://sharpend.io/the-limitations-of-chaos-engineering/>



Photo by [Diego PH](#) on [Unsplash](#)

My Chaos

Let's relive the DNS connection error memory leak story together

C++ code demo

Kubernetes deployment

kube-prometheus

Chaos mesh

SLOs, alerts, etc.



Use case

App tries to resolve DNS, leaks memory when unsuccessful

Container build in GitLab CI/CD, deployed to Kubernetes

Kube-prometheus metrics collection & alert rules

Add DNS failure as chaos experiment

Verify memory going up, triggering alerts, SLOs, MR updates, etc.

Demo 

<https://gitlab.com/everyonecancontribute/observability/cpp-dns-leaker>



Demo hosted by Civo

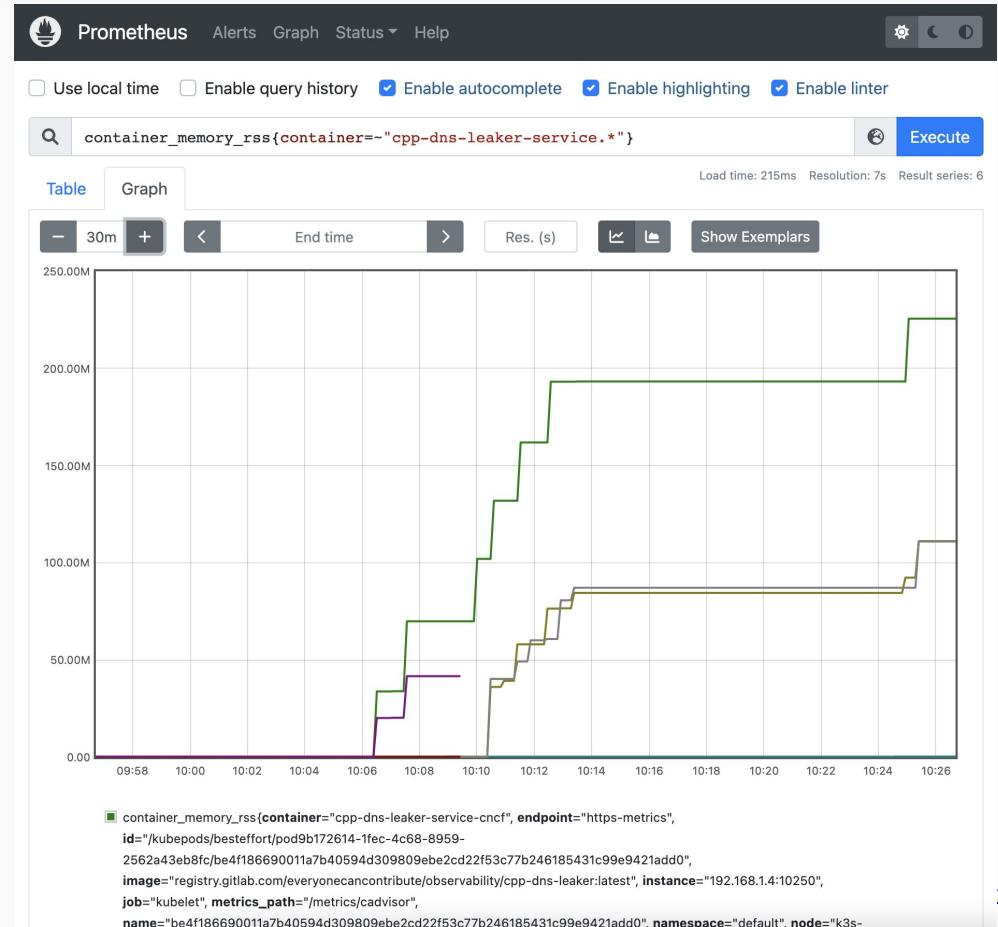
Demo - Chaos Mesh

```
kind: Schedule
apiVersion: chaos-mesh.org/v1alpha1
metadata:
  namespace: default
  name: schedule-dns-chaos-kubecon
  annotations:
    experiment.chaos-mesh.org/pause: 'true'
    kubectl.kubernetes.io/last-applied-configuration: >-
      {"apiVersion":"chaos-mesh.org/v1alpha1","kind":"Schedule","metadata":{},"name":"schedule-dns-chaos-kubecon","namespace":"default","spec":{"concurrencyPolicy":"Allow","dnsChaos":{"action":"error","duration":"60s","mode":"all","patterns":["olly.*","cncf.io","gitlab.com"]},"selector":{"namespaces":["default"]}},"historyLimit":2,"schedule":*****,"type":"DNSChaos"}}
spec:
  schedule: '* * * * *'
  startingDeadlineSeconds: null
  concurrencyPolicy: Allow
  historyLimit: 2
  type: DNSChaos
  dnsChaos:
    action: error
    selector:
      namespaces:
        - default
    mode: all
    duration: 60s
    patterns:
      - olly.*
      - cncf.io
      - gitlab.com
```

Events

	schedule-dns-chaos-kubecon	12 SECONDS AGO
	Successfully update lastScheduleTime of resource	
	schedule-dns-chaos-kubecon	12 SECONDS AGO
	Create new object: schedule-dns-chaos-kubecon-nnzv7	
	schedule-dns-chaos-kubecon	1 MINUTE AGO
	Successfully update lastScheduleTime of resource	
	schedule-dns-chaos-kubecon	1 MINUTE AGO
	Create new object: schedule-dns-chaos-kubecon-k2q8w	
	schedule-dns-chaos-kubecon	2 MINUTES AGO
	Successfully update lastScheduleTime of resource	
	schedule-dns-chaos-kubecon	2 MINUTES AGO
	Create new object: schedule-dns-chaos-kubecon-c94tc	

Demo - DNS failure & memory leak



From DIY Monitoring to Observability



KubeCon



CloudNativeCon

Europe 2022



Photo by [Michael Friedrich](#)

From Metrics Monitoring to Observability?

Metrics

Logs/Events

(Distributed) Tracing

(Continuous) Profiling

Error Tracking, RUM agents, etc.

Shifting from monolith to microservices.

Breath.

💡 Start with metrics and tracing.

<https://samnewman.io/books/monolith-to-microservices/>

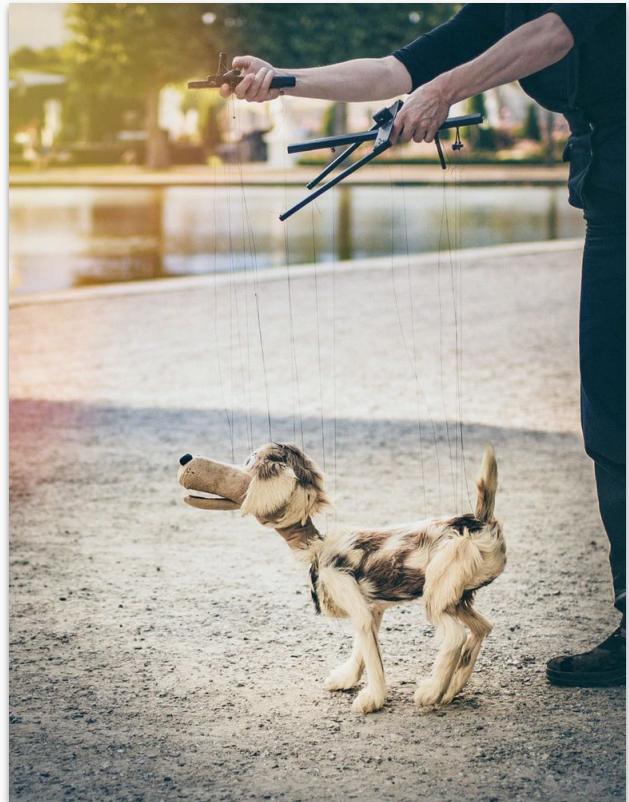


Photo by [Eric Masur](#) on [Unsplash](#)

Traces

Different to logs

Spans with start/end time

More metadata context

App code instrumentation

Spec

OpenTelemetry

Bring your own backend (Jaeger, ES/Clickhouse, etc.)

Build your own OTel distribution

Evaluate auto-instrumentation

<https://o11y.love/topics/tracing/>

<https://o11y.love/topics/collections-specs/#opentelemetry>

<https://www.jaegertracing.io/docs/1.25/architecture/>

<https://docs.google.com/presentation/d/15CzbqO3leXOnH3Pwz94zYRzeOT8g92YQK7wC-li8HzU/edit>

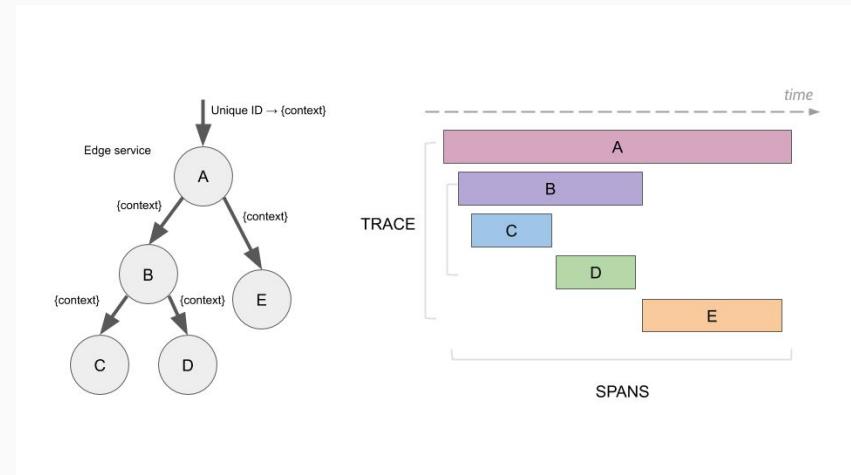


Image credit: [Jaeger Tracing Docs](#)

eBPF - new idea?

"Instead of relying on static counters and gauges exposed by the operating system, eBPF enables the collection & in-kernel aggregation of custom metrics and generation of visibility events based on a wide range of possible sources."

- Cilium, network connectivity security and observability
- Falco, Kubernetes threat detection engine
- Parca, Continuous Profiling
- Tracee, Runtime Security and Forensics

?

Source for SLOs

<https://o11y.love/topics/collections-specs/#ebpf>

<https://ebpf.io/>

<https://cilium.io/>

<https://falco.org/>

<https://www.parca.dev/>

<https://aquasecurity.github.io/tracee/latest/>

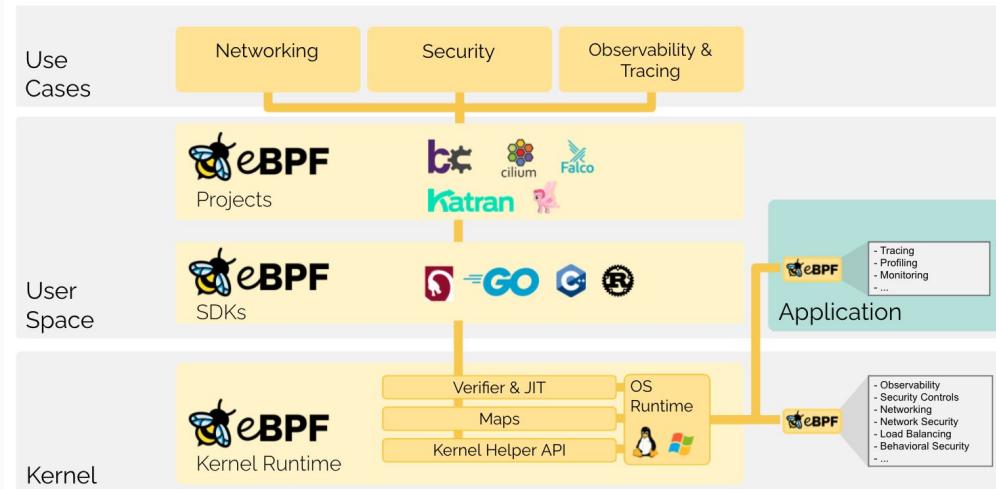


Image by <https://ebpf.io/>

Left Shift

Step-by-step: Iterate faster



KubeCon



CloudNativeCon

Europe 2022



Photo by [Michael Friedrich](#)

See value in ~~metrics, logs, traces~~ Observability

App insights for !Devs

Help everyone not familiar with your code

SREs, Dev, Ops, DevOps

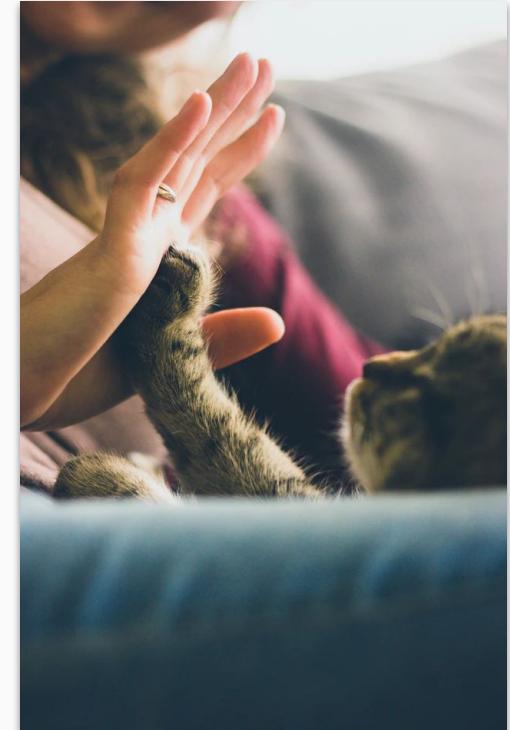
Is the problem code or something else?

Use boring solutions

Start with /metrics and Prometheus

Add tracing with OpenTelemetry

Collect more Observability data



Left Shift SLO

Observability everywhere

- Collect metrics, and more events

- Learn app instrumentation & educate DevOps teams

DevOps workflows

- SLOs and alerts in CI/CD, Merge Requests in staging envs

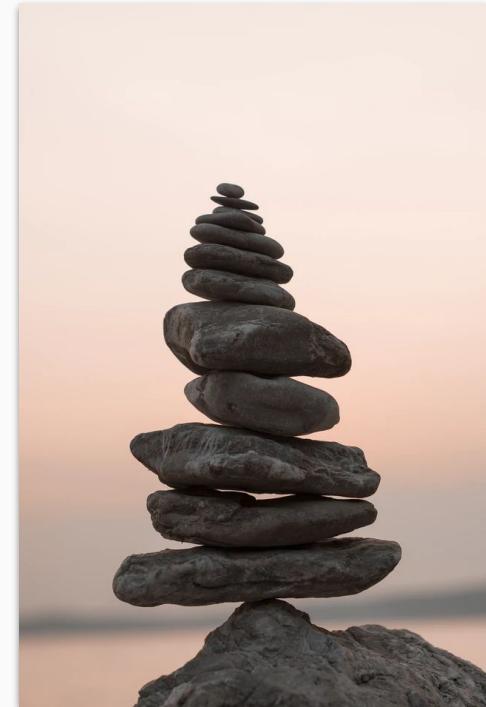
- Alert channels, incident management on a DevOps platform

Cloud-native benefits

- Deployments and auto-scaling

- CNCF projects integrations & best practices

- Kubernetes, Prometheus, OpenTelemetry, Litmus, etc.



Metrics with Prometheus for Kubernetes: <https://kubernetes.io/docs/concepts/cluster-administration/system-metrics/>

OpenTelemetry tracing in Kubernetes: <https://github.com/kubernetes/kubernetes/pull/94942>

OpenTelemetry tracing in Prometheus: <https://github.com/prometheus/prometheus/pull/9724>

Photo by [Bekir Dönmez](#) on [Unsplash](#)

Left Shift with Chaos

Bring chaos into Observability

Chaos workflows

Built-in and custom experiments

Verify

SLOs

Quality Gates

Reliability

Iterate and innovate

So many great chaos engineering tools

<https://o11y.love/topics/chaos-engineering/>

Photo by [Pavel Neznanov](#) on [Unsplash](#)



Wishlist

ML to correlate events and auto-generate SLOs

Chaos out-of-the-box

Accessible to ~~SREs, Dev, Ops, DevOps~~ everyone

OpenTelemetry adoption and use cases

CI/CD Observability: <https://gitlab.opentelemetry.love/> research



Photo by [Bernard Hermant](#) on [Unsplash](#)

Recap

App instrumentation with metrics & traces

PromQL and SLOs

Quality gates with Keptn & Prometheus

Shift Left

Chaos Engineering with Litmus, Chaos Mesh

Observability everywhere

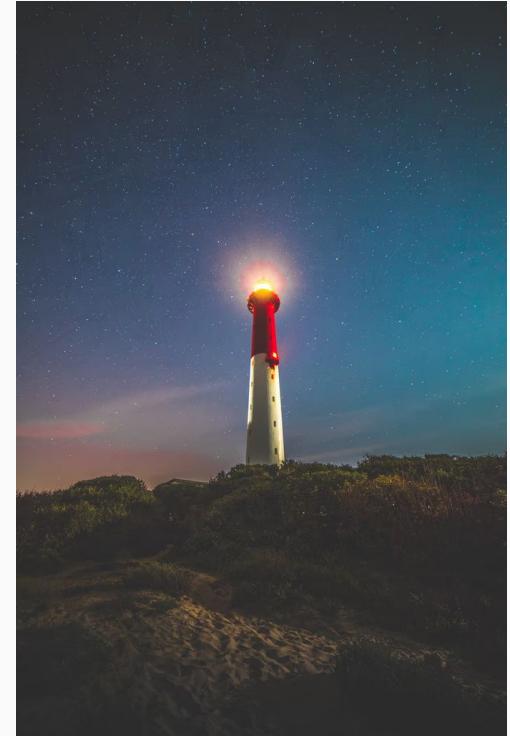


Photo by Valentin G on [Unsplash](#)

Q&A

Learn more at
<https://olly.love/>



KubeCon



CloudNativeCon

Europe 2022



From Monitoring to Observability: Left Shift your SLOs with Chaos



Michael Friedrich
Senior Developer Evangelist at GitLab



EveryoneCanContribute.com



KubeCon



CloudNativeCon

Europe 2022