

# ScrapeConfig CRD

“The Rising Star in the  
Prometheus Operator  
Ecosystem”

Jayapriya Pai  
Senior Software Engineer, Red Hat



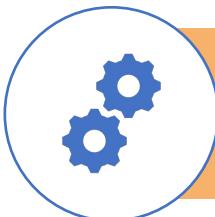
Supported By  
 CLOUD NATIVE  
COMPUTING FOUNDATION

# About me

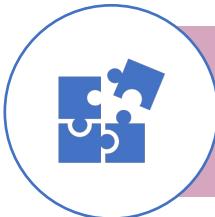


- **Jayapriya Pai (Jp)**
- Senior Software Engineer
- OpenShift In-cluster monitoring team Red Hat
- Contributor and member in Prometheus-Operator
- slashpai (GitHub)
- techieslash (Twitter/X)
- jayapriyapai (LinkedIn)

# Takeaways



Prometheus-Operator Brief and Why  
new ScrapeConfig CRD?



Intro to ScrapeConfig CRD



ScrapeConfig Examples

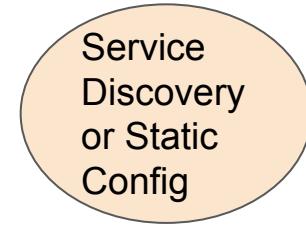
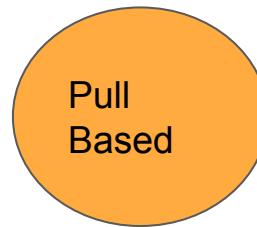
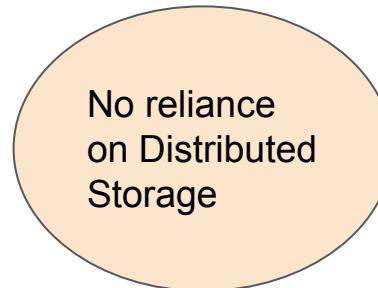
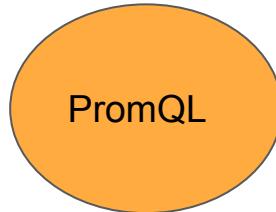
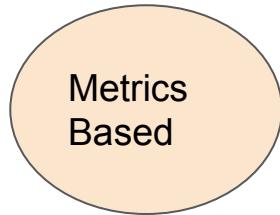
# Before we begin

# Prometheus Brief

“Prometheus is an open-source systems monitoring and alerting toolkit originally built at SoundCloud”

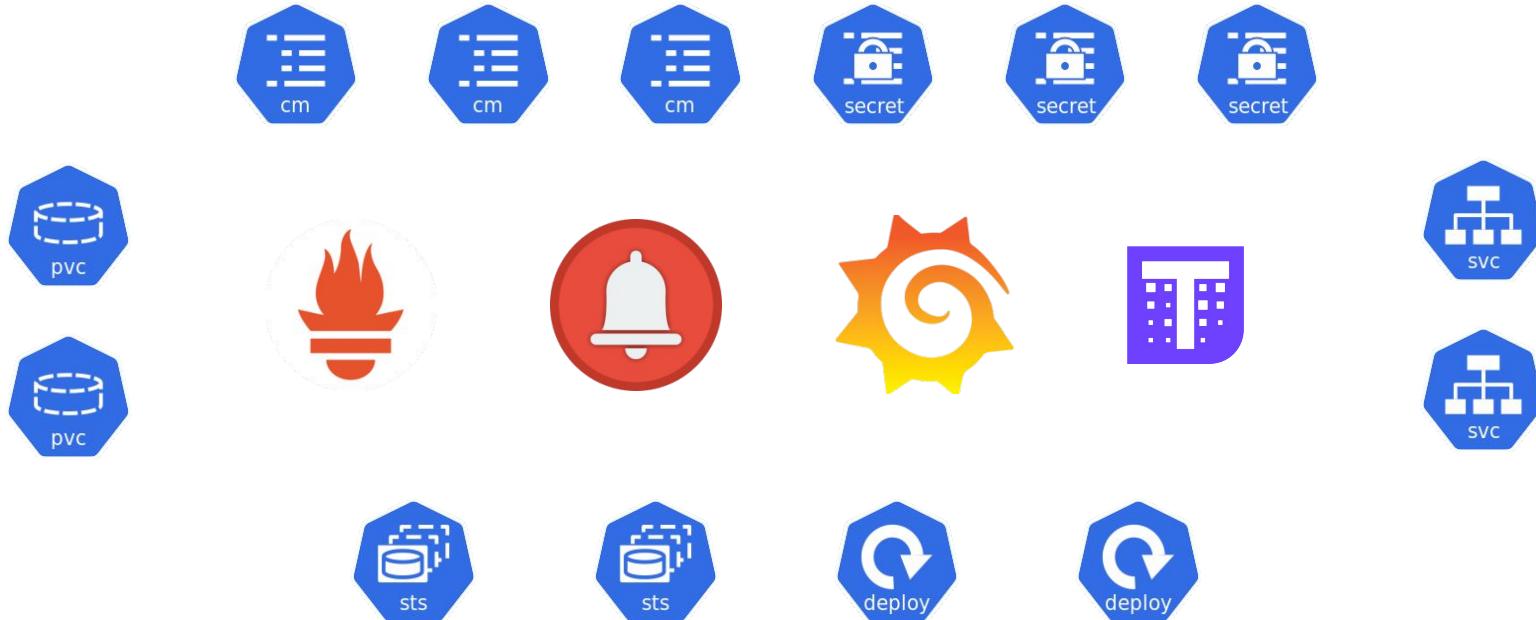


CNCF Project since 2016 and second graduated project after Kubernetes

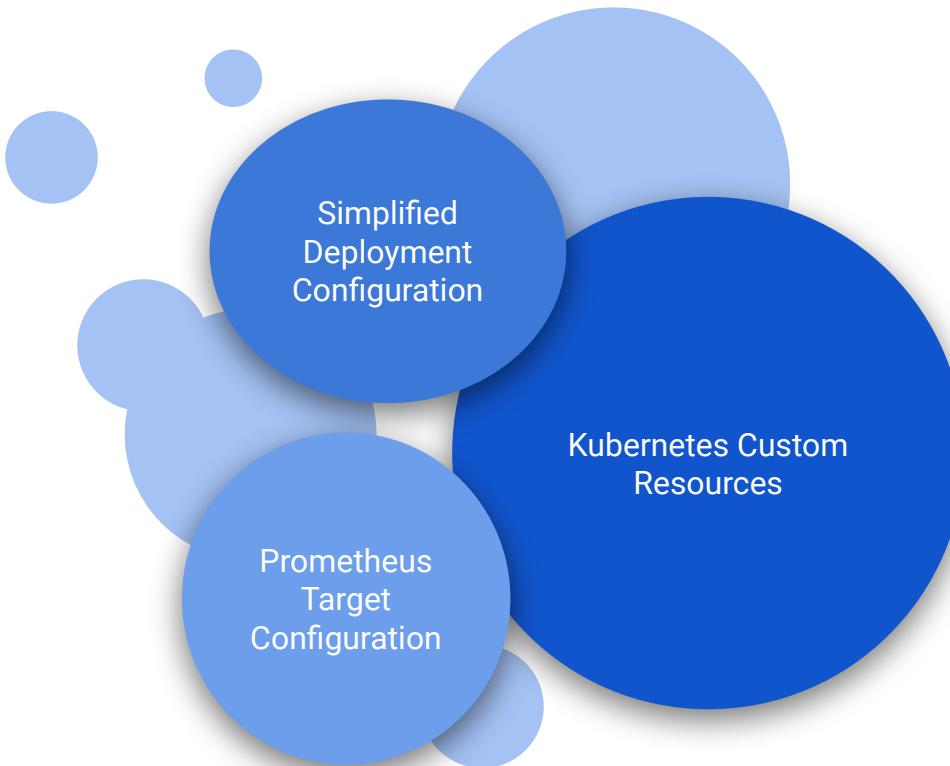


# Why Prometheus-Operator?

# Typical Monitoring Stack



# Prometheus-Operator



# Prometheus Operator

Screenshot of the GitHub repository page for `prometheus-operator`.

**Overview**   **Repositories** 5   **Projects**   **Packages**   **Teams** 4   **People** 14

**Unfollow**

**Popular repositories**

- prometheus-operator** (Public)  
Prometheus Operator creates/configures/manages Prometheus clusters atop Kubernetes  
Go   ⭐ 8.6k   📈 3.7k
- kube-prometheus** (Public)  
Use Prometheus to monitor Kubernetes and applications running on Kubernetes  
Jsonnet   ⭐ 6k   📈 1.8k
- runbooks** (Public)  
HTML   ⭐ 74   📈 121
- website** (Public)  
The Prometheus Operator & kube-prometheus website  
Shell   ⭐ 10   📈 16

**View as: Public** ▾  
You are viewing the README and pinned repositories as a public user.  
You can [create a README file](#) visible to anyone.

**People**



# Prometheus Custom Resource

## Important Fields:

- Selectors:
  - serviceMonitorNamespaceSelector, serviceMonitorSelector
  - podMonitorNamespaceSelector, podMonitorSelector
  - probeNamespaceSelector, probeSelector
  - scrapeConfigNamespaceSelector, scrapeConfigSelector
- version
- replicas
- alerting
- resources

```
apiVersion: monitoring.coreos.com/v1
kind: Prometheus
metadata:
  name: prometheus
  labels:
    prometheus: prometheus
spec:
  replicas: 2
  resources:
    requests:
      memory: 400Mi
  serviceMonitorSelector:
    matchLabels:
      team: frontend
  alerting:
    alertmanagers:
    - namespace: default
      name: alertmanager
      port: web
```

# Collecting Metrics

Service Monitor

Pod Monitor

Probe

Selects pods via service (and their endpoints) based on label selector

Selects pods directly based on label selector

Monitor static target or ingress objects (blackbox\_exporter needed)  
Selects ingress objects based on label selector

So what about targets  
outside Kubernetes?

# Additional Scrape Config

```
- job_name: "prometheus"
  static_configs:
  - targets: ["prometheus.demo.do.prometheus.io"]
```

**prometheus-additional.yaml**

```
kubectl create secret generic additional-scrape-configs  
--from-file=prometheus-additional.yaml
```

Prometheus

Targets

All scrape pools ▾ All Unhealthy Expand All Filter by endpoint or labels Unknown Unhealthy Healthy

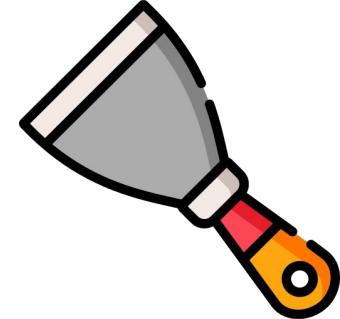
prometheus (1/1 up) Show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://prometheus.demo.do.prometheus.io/metrics	UP	<code>instances="prometheus.demo.do.prometheus.io:80" job="prometheus"</code> Discovered labels: <code>__address__="prometheus.demo.do.prometheus.io" __metrics_path__="/metrics" __scheme__="http" __scrape_interval__="30s" __scrape_timeout__="10s" job="prometheus"</code>	2.724s ago	1.885s	

```
apiVersion: monitoring.coreos.com/v1
kind: Prometheus
metadata:
  name: prometheus
  labels:
    prometheus: prometheus
spec:
  replicas: 2
  serviceAccountName: prometheus
  serviceMonitorSelector: {}
  additionalScrapeConfigs:
    name: additional-scrape-configs
    key: prometheus-additional.yaml
```

# But what about these scenarios?

- Validating config defined in additional scrape configs?
- Kubernative native way to declare targets outside Kubernetes?
- Monitor targets specifying config as in Prometheus?



v1alpha1  
(from v0.65)

# ScrapeConfig CRD

“ScrapeConfig CRD exemplifies Prometheus's reach beyond Kubernetes”

# A github issue that was logged 4 years ago..

The screenshot shows a GitHub issue page for a pull request. At the top, there's a navigation bar with links for Code, Issues (189), Pull requests (46), Discussions, Actions, Projects, Security, and Insights. Below the navigation is the title "Generic ScrapeConfig CRD #2787". A "New issue" button is located in the top right corner.

The issue is marked as "Closed" by "auhlig" on Oct 1, 2019, with 27 comments. The main comment from "auhlig" discusses the need for a "ScrapeConfig" CRD to monitor services and pods. It proposes adding a "NodeMonitor" that generates a "scrape config for nodes" based on its definition. The comment also mentions being willing to contribute a PR similar to #2566 if there's general agreement. The comment has 48 likes and 2 replies.

On Oct 1, 2019, "auhlig" added the "kind/feature" label to the issue. The issue details sidebar includes sections for Assignees (xiu), Labels (kind/feature), Projects (None yet), Milestone (No milestone), Development (No branches or pull requests), and Notifications (Customize, Unsubscribe). There are 15 participants listed at the bottom.

On Oct 2, 2019, "brancz" commented, asking for use cases and explaining why they can't be achieved with existing PodMonitor/ServiceMonitor objects. The comment has 1 like.



# feat: Add ScrapeConfig CRD #5335

[Code ▾](#)[Merged](#)philippgough merged 27 commits into [prometheus-operator:main](#) from [xiu:scrapeconfig\\_impl](#) on May 3, 2023[Conversation 219](#)[Commits 27](#)[Checks 16](#)[Files changed 51](#)

+5,088 -214



xiu commented on Feb 12, 2023 • edited

Member



## Description

This PR adds the `ScrapeConfig` CRD.

Proposal: [ScrapeConfig](#)

Issue: [#2787](#)

## Type of change

What type of changes does your code introduce to the Prometheus operator? Put an  in the box that apply.

- CHANGE (fix or feature that would cause existing functionality to not work as expected)
- FEATURE (non-breaking change which adds functionality)
- BUGFIX (non-breaking change which fixes an issue)
- ENHANCEMENT (non-breaking change which improves existing functionality)
- NONE (if none of the other choices apply. Example, tooling, build system, CI, docs, etc.)

## Changelog entry

Add ScrapeConfig CRD, allowing users to scrape targets outside the kubernetes cluster



5



8

### Reviewers



mrueg



slashpai



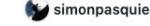
genofire



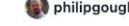
JoaoBraveCoding



ArthurSens



simonpasquier



philippgough



### Assignees

No one—assign yourself



### Labels

None yet



### Projects

None yet



### Milestone

No milestone



### Development

Last updated 11 hours ago

# ScrapeConfig CR

```
apiVersion: monitoring.coreos.com/v1alpha1
kind: ScrapeConfig
metadata:
  name: scrape-config-static-external
  namespace: default
  labels:
    app.kubernetes.io/name: scrape-config-static-external
spec:
  staticConfigs:
    - labels:
        job: node-exporter-external
      targets:
        - node.demo.do.prometheus.io
    - labels:
        job: prometheus-external
      targets:
        - prometheus.demo.do.prometheus.io
```

v1alpha1

**Provides a Kubernetes native API to create and manage scrape configurations**

**Monitor resources outside cluster as well (Say node exporter running outside k8s)**



## Targets

[All scrape pools ▾](#)[All](#)   [Unhealthy](#)   [Expand All](#)

Filter by endpoint or labels

 [Unknown](#)  [Unhealthy](#)  [Healthy](#)**scrapeConfig/default/scrape-config-static-external (2/2 up)** [show less](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
<a href="#">http://node.demo.do.prometheus.io/metrics</a>	<span>UP</span>	<a href="#">instance="node.demo.do.prometheus.io:80" job="node-exporter-external"</a> ▾	25.987s ago	1.216s	
<a href="#">http://prometheus.demo.do.prometheus.io/metrics</a>	<span>UP</span>	<a href="#">instance="prometheus.demo.do.prometheus.io:80" job="prometheus-external"</a> ▾	21.291s ago	812.331ms	

# EC2 SD Example

```
apiVersion: monitoring.coreos.com/v1alpha1
kind: ScrapeConfig
metadata:
  name: scrape-config-ec2
  namespace: default
  labels:
    app.kubernetes.io/name: scrape-config-ec2
spec:
  ec2SDConfigs:
    - region: us-east-1
      accessKey:
        key: accessKey
        name: aws-api-access
      secretKey:
        key: secretKey
        name: aws-api-access
      port: 9100
      refreshInterval: 30s
    relabelings:
      - sourceLabels: [__meta_ec2_public_ip]
        replacement: ${1}:9100
        targetLabel: __address__
```

- Create an EC2 instance, add the security group to open port 9100, deploy node-exporter (can use native cloud agent as well instead)
- Attach the policy `AmazonEC2ReadOnlyAccess` to IAM Role/User ([reference](#))
- Relabelling was required because only public ip will be accessible to scrape target metrics
- Created k8s secret with api access key and secret key
- Apply the ScrapeConfig



## Targets

All scrape pools ▾

[All](#) [Unhealthy](#) [Expand All](#)

Filter by endpoint or labels

 Unknown  Unhealthy  Healthy[scrapeconfig/default/scrape-config-ec2 \(2/2 up\)](#) [show less](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
<a href="#">http://52.23.225.92:9100/metrics</a>	<span>UP</span>	<code>instance="52.23.225.92:9100"</code> <code>job="scrapeconfig/default/scrape-config-ec2"</code>	8.560s ago	452.985ms	
<a href="#">http://54.91.28.35:9100/metrics</a>	<span>UP</span>	<code>instance="54.91.28.35:9100"</code> <code>job="scrapeconfig/default/scrape-config-ec2"</code>	2.561s ago	450.267ms	

**Note:** Don't expose /metrics endpoint without proper auth/TLS  
(this is example hence didn't use for sake of simplicity)

# Kubernetes SD Examples

```
apiVersion: monitoring.coreos.com/v1alpha1
kind: ScrapeConfig
metadata:
  name: scrape-config-kubernetes-sd-example
  namespace: default
  labels:
    app.kubernetes.io/name: scrape-config-kubernetes-sd-example
spec:
  scheme: HTTPS
  authorization:
    credentials:
      name: prometheus-secret
      key: token
  tlsConfig:
    ca:
      secret:
        name: default-server
        key: ca.crt
    insecureSkipVerify: true
  kubernetesSDConfigs:
    - role: Node
```

```
apiVersion: monitoring.coreos.com/v1alpha1
kind: ScrapeConfig
metadata:
  name: scrape-config-kubernetes-sd-example-pod
  namespace: default
  labels:
    app.kubernetes.io/name: scrape-config-kubernetes-sd-example-pod
spec:
  scheme: HTTP
  kubernetesSDConfigs:
    - role: Pod
      selectors:
        - role: pod
          label: app=sc.example
```



## Targets

All scrape pools ▾

All Unhealthy Expand All



Filter by endpoint or labels

 Unknown  Unhealthy  HealthyscrapeConfig/default/scrape-config-kubernetes-sd-example (3/3 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
<a href="https://172.18.0.3:10250/metrics">https://172.18.0.3:10250/metrics</a>	UP	<code>instance="multi-node-worker"</code> <code>job="scrapeConfig/default/scrape-config-kubernetes-sd-example"</code> ▼	28.19s ago	39.534ms	
<a href="https://172.18.0.2:10250/metrics">https://172.18.0.2:10250/metrics</a>	UP	<code>instance="multi-node-worker2"</code> <code>job="scrapeConfig/default/scrape-config-kubernetes-sd-example"</code> ▼	3.264s ago	34.965ms	
<a href="https://172.18.0.7:10250/metrics">https://172.18.0.7:10250/metrics</a>	UP	<code>instance="multi-node-control-plane"</code> <code>job="scrapeConfig/default/scrape-config-kubernetes-sd-example"</code> ▼	17.814s ago	37.108ms	

You have option to filter targets as well in kubernetesSDConfig



## Targets

All scrape pools ▾

All Unhealthy Expand All

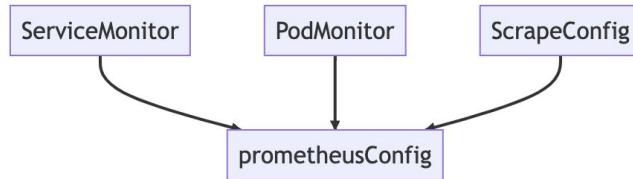


Filter by endpoint or labels

 Unknown  Unhealthy  HealthyscrapeConfig/default/scrape-config-kubernetes-sd-example-pod (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
<a href="http://10.244.2.8:8080/metrics">http://10.244.2.8:8080/metrics</a>	UP	<code>instances="10.244.2.8:8080"</code> <code>job="scrapeConfig/default/scrape-config-kubernetes-sd-example-pod"</code> ▼	13.381s ago	1.051ms	

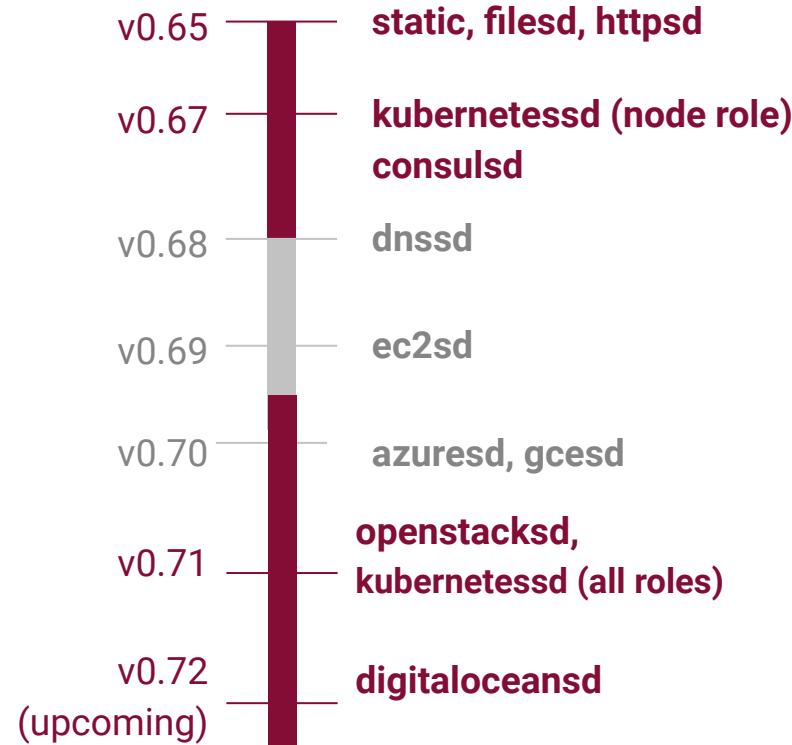
# ScrapeConfig



Act the same as the other CRDs and append scrape configurations to the configuration

# Service Discovery Configs in ScrapeConfig

Currently supported: 10



# Interested in Contributing?



- Documentation
- Tutorials
- Features
- Enhancements
- Bug Fixes

GitHub screenshot of the Prometheus Operator repository showing the Issues page.

The Issues page displays 194 open issues. The search bar shows the filter "is:issue is:open".

Issues listed:

- #4241: Alertmanager mTLS config (TamasNeumer) - Open, 2 comments
- #4200: Support basic\_auth\_users in web configuration for Prometheus ... (jack1902) - Open, 14 comments
- 194 Open, 2,459 Closed
- #6281: Enable kubernetes node labels (HenryGuo1019) - Open, 7 hours ago
- #6279: Enable minVersionKeepFiringFor for Thanos v0.34.0 (dragongel) - Open, 1 day ago
- #6277: Invalid child routes in AlertmanagerConfig CRD triggers a panic (siretar) - Open, 2 days ago
- #6275: reconciliation is not working for maximumStartupDurationSeconds (prudnitskiy) - Open, 3 days ago
- #6269: Is targetPort on the ServiceMonitor actually deprecated? (hotteat773) - Open, 3 days ago
- #6270: Allow turning off automountServiceAccountToken for Prometheus SA (oscarbm1998) - Open, last week
- #6266: The Prometheus pod launched from the image does not contain a file specifying the OS distribution information. (toshi0228) - Open, last week
- #6246: Support file paths for certificate, key and client CA in web TLS configuration (abohone) - Open, 2 weeks ago

See <https://github.com/prometheus-operator/prometheus-operator/blob/main/CONTRIBUTING.md>

# Links and References

- <https://github.com/prometheus-operator/prometheus-operator/tree/main/Documentation>
- <https://github.com/slashpai/prometheus-operator-examples>
- <https://www.youtube.com/watch?v=MuHPMXCGiLc>
- [https://www.youtube.com/watch?v=P\\_R4CFFxEQ](https://www.youtube.com/watch?v=P_R4CFFxEQ)
- [https://www.youtube.com/watch?v=UpH\\_Say4D3M](https://www.youtube.com/watch?v=UpH_Say4D3M)

# Thank you!



<https://github.com/prometheus-operator>



<https://prometheus-operator.dev>

