

Prow Under the Hood: CI/CD for Kubernetes Development

Arka Saha

Software Engineer, VMware by
Broadcom

Rayan Das

Senior SRE, OneTrust



Supported By



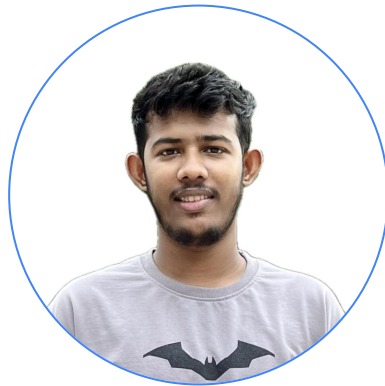
Who we are?



Arka Saha

Software Engineer, VMware by Broadcom

 [ArkaSaha30](#)  [arkasaha30](#)



Rayan Das

Senior SRE, OneTrust

 [rayandas](#)  [raydeeam](#)

Agenda

- Prow: What and Why?
- Types of Prowjobs
- Prowjob: How to write yours?
- Test your Prowjob
- Key Takeaways

Prow: What and Why?

What is Prow?



What is Prow?

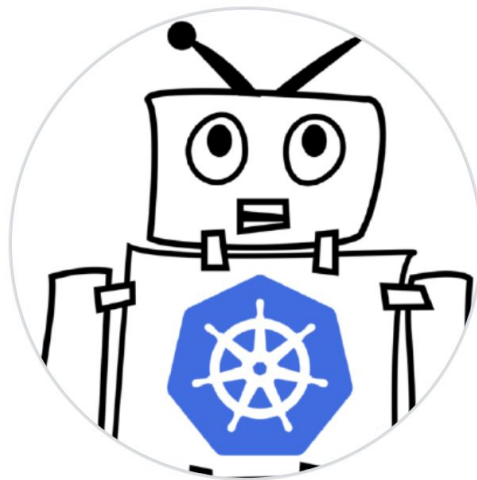
- **Kubernetes based CI/CD System**

What is Prow?

- Kubernetes based CI/CD System
- **Extensible features**

What is Prow?

- Kubernetes based CI/CD System
- Extensible features
 - Github Automation



Kubernetes Prow Robot
k8s-ci-robot

What is Prow?

- Kubernetes based CI/CD System
- Extensible features
 - Github Automation
 - Chatops



rayandas commented on Nov 30, 2023

Member ...

Being a part of the v1.29 enhancements team as a shadow was a fantastic experience! A big shoutout to [@npolshakova](#) and [@Priyankasaggu11929](#) for their guidance.

Heartfelt congratulations to [@salehsedghpour](#) for stepping into the role of Enhancements Lead for v1.30 🎉
Kudos to [@katcosgrove](#) and [@gracennng](#) as well! 🙌

Looking forward to v1.30 🦋
/honk



5



k8s-ci-robot commented on Nov 30, 2023

Member ...

@rayandas:



► Details



What is Prow?

- Kubernetes based CI/CD System
- Extensible features
 - Github Automation
 - Chatops
 - Test Grid

[illegible]

What is Prow?

- Kubernetes based CI/CD
- Extensible features
- Configuration as Code

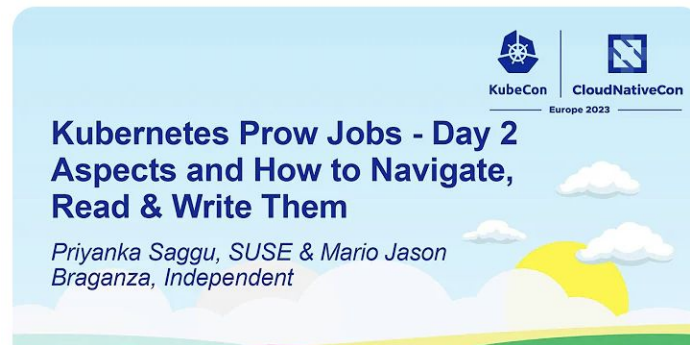
```
periodics:
- interval: 1h
  name: ci-kubernetes-unit
  annotations:
    fork-per-release: "true"
    testgrid-num-failures-to-alert: '3'
    testgrid-alert-stale-results-hours: '24'
    testgrid-days-of-results: '1'
    testgrid-create-test-group: 'true'
    testgrid-dashboards: sig-release-master-blocking
    testgrid-alert-email: kubernetes-sig-testing-alerts@googlegroups.com
  decorate: true
  cluster: k8s-infra-prow-build
  extra_refs:
  - org: kubernetes
    repo: kubernetes
    base_ref: master
    path_alias: k8s.io/kubernetes
  spec:
    # unit tests have no business requiring root or doing privileged operations
    securityContext:
      # NOTE: these are arbitrary non-root values. They don't exist in the
      # image and don't need to, the unit tests should only write to TMPDIR
      runAsUser: 2001
      runAsGroup: 2010
    containers:
    - image: gcr.io/k8s-staging-test-infra/kubekins-e2e:v20240111-cf1d81388e-master
      securityContext:
        allowPrivilegeEscalation: false
      command:
      - make
      - test
      # TODO: direct copy from pull-kubernetes-bazel-test, tune these
      resources:
        limits:
          cpu: 4
          memory: "36Gi"
        requests:
          cpu: 4
          memory: "36Gi"
```

<https://github.com/kubernetes/test-infra/blob/master/config/jobs/kubernetes/sig-testing/make-test.yaml#L149-L188>

Learn More ...



<https://www.youtube.com/watch?v=sdU-3cqjMg>



<https://www.youtube.com/watch?v=gKKfXTZd7VU>

Why Prow for CI/CD?

- Platform familiarity

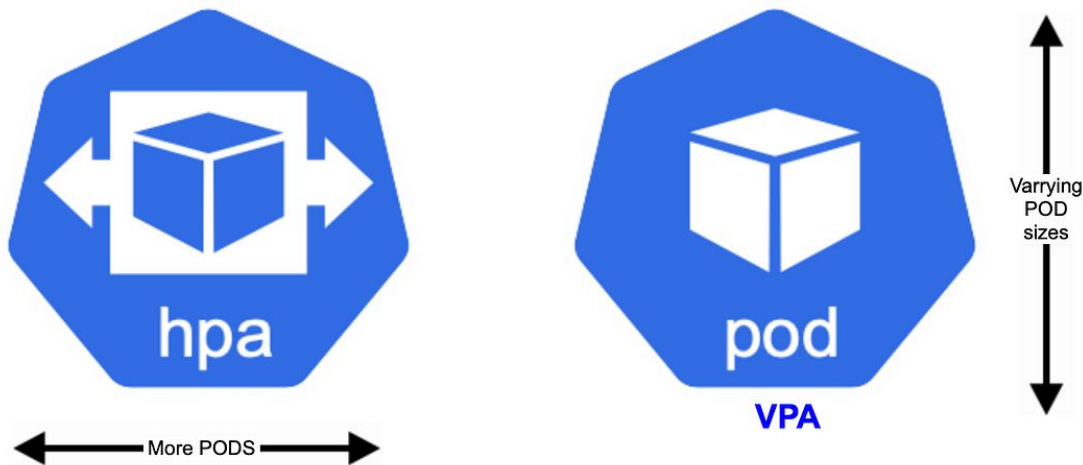


Why Prow for CI/CD?

- Platform familiarity
- **Reliability**

Why Prow for CI/CD?

- Platform familiarity
- Reliability
 - Scalability



Why Prow for CI/CD?

- Platform familiarity
- **Reliability**
 - Scalability
 - **Monitoring**

Why Prow for CI/CD?

- Platform familiarity
- Reliability
 - Scalability
 - Monitoring
- Infrastructure - *Prometheus/Grafana*



Why Prow for CI/CD?

- Platform familiarity
- Reliability
 - Scalability
 - Monitoring
 - Infrastructure - *Prometheus/Grafana*
 - Workload - *Spyglass/Testgrid*



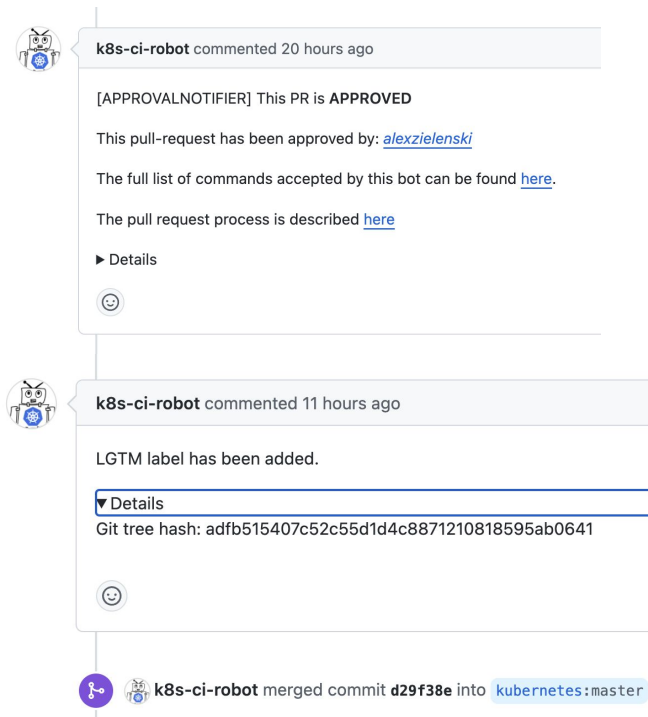
Why Prow for CI/CD?

- Platform familiarity
- **Reliability**
 - Scalability
 - Monitoring
 - **Multi-cloud**



Why Prow for CI/CD?

- Platform familiarity
- Reliability
- **Automation**



The screenshot displays a sequence of GitHub Actions log entries. The first entry, from the **k8s-ci-robot**, is a comment made 20 hours ago stating that a pull request is **APPROVED**. It includes links to the list of accepted commands and the pull request process. The second entry, also from **k8s-ci-robot** and made 11 hours ago, reports that the **LGTM** label has been added and provides a Git tree hash. The final entry shows the **k8s-ci-robot** merging commit **d29f38e** into the **kubernetes:master** branch.

k8s-ci-robot commented 20 hours ago

[APPROVALNOTIFIER] This PR is **APPROVED**

This pull-request has been approved by: [alexzielenski](#)

The full list of commands accepted by this bot can be found [here](#).

The pull request process is described [here](#)

► Details

☺

k8s-ci-robot commented 11 hours ago

LGTM label has been added.

▼ Details

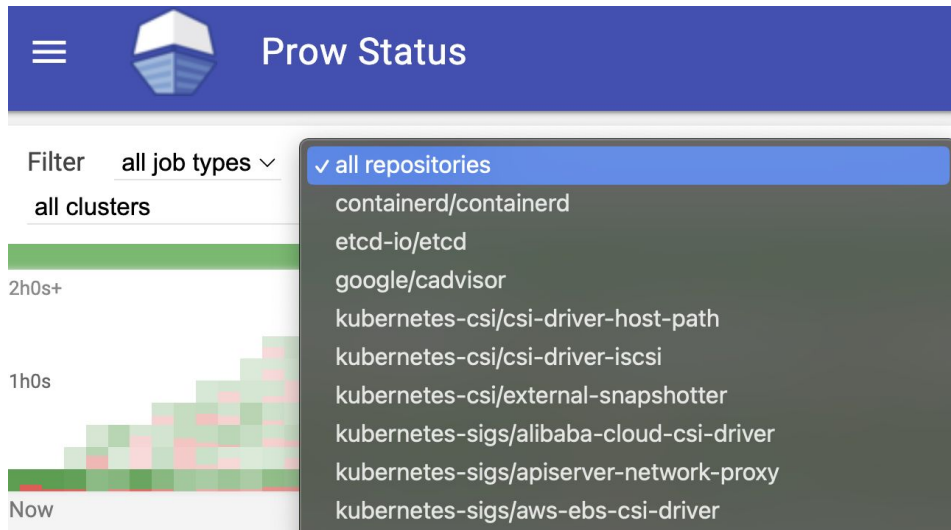
Git tree hash: adfb515407c52c55d1d4c8871210818595ab0641

☺

k8s-ci-robot merged commit **d29f38e** into **kubernetes:master**

Why Prow for CI/CD?

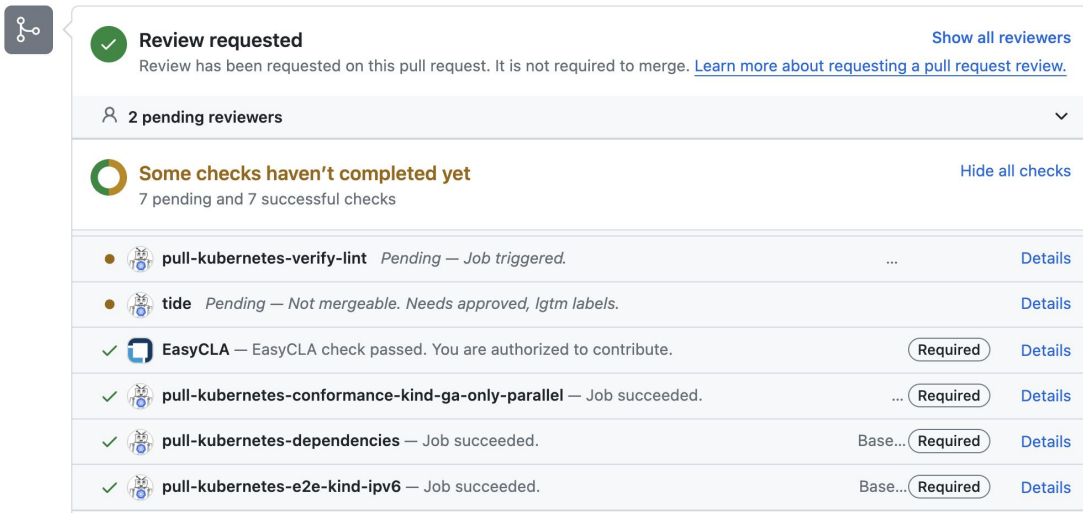
- Platform familiarity
- Reliability
- Automation
- **Multi-org/Multi-repo**



Types of Prowjobs

Types of Prowjobs

- **Presubmits**
(triggered against PRs)

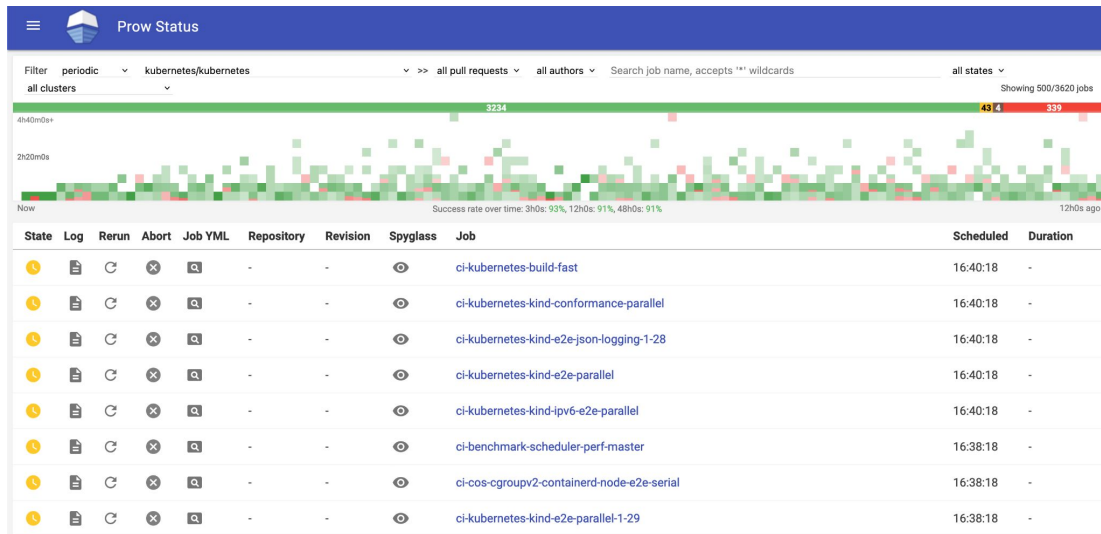


The screenshot displays a GitHub pull request interface with a Prow status summary. At the top, a green checkmark icon is next to the text "Review requested", with a link to "Show all reviewers" on the right. Below this, a message states "Review has been requested on this pull request. It is not required to merge." followed by a link "Learn more about requesting a pull request review.". A section titled "2 pending reviewers" is shown with a dropdown arrow. Below that, a yellow circle icon is next to the text "Some checks haven't completed yet", with a link to "Hide all checks" on the right. A message indicates "7 pending and 7 successful checks". A list of checks follows:

●		pull-kubernetes-verify-lint	Pending — Job triggered.	...	Details
●		tide	Pending — Not mergeable. Needs approved, lgtm labels.		Details
✓		EasyCLA	— EasyCLA check passed. You are authorized to contribute.	Required	Details
✓		pull-kubernetes-conformance-kind-ga-only-parallel	— Job succeeded.	Required	Details
✓		pull-kubernetes-dependencies	— Job succeeded.	Base... Required	Details
✓		pull-kubernetes-e2e-kind-ipv6	— Job succeeded.	Base... Required	Details

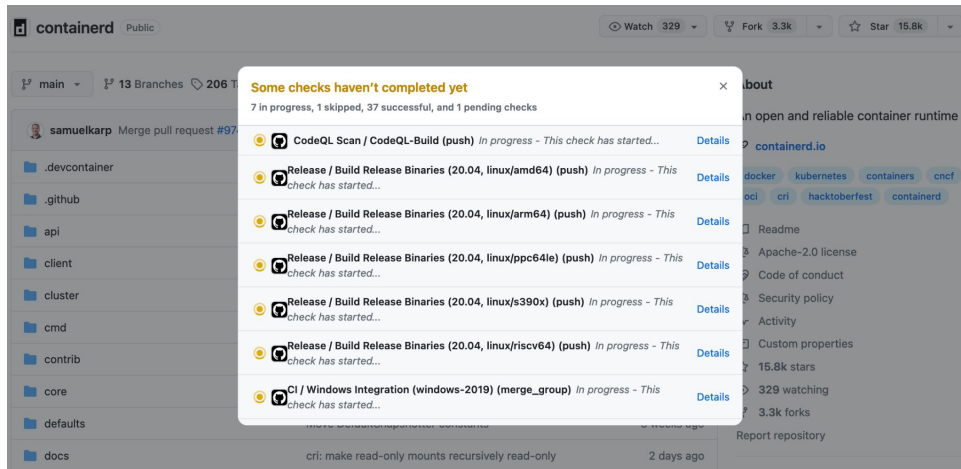
Types of Prowjobs

- Presubmits
(triggered against PRs)
- Periodics
(triggered recurrently)



Types of Prowjobs

- Presubmits
(triggered against PRs)
- Periodics
(triggered recurrently)
- Postsubmits
(triggered post-merge of PRs)



Prowjob: How to write yours?

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}

interval: 1h

run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []

trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits:
 org/repo:

postsubmits:
 org/repo:

periodics: } — Specifies the type of prowjob

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

`presubmits:` `postsubmits:` `periodics:`
 `org/repo:` `org/repo:` **Specifies the repository to clone**

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits:

org/repo:

postsubmits:

org/repo:

periodics:

} Specifies the name of prowjob
matching regex: `^[A-Za-z0-9-_.]+$`

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

Common to ALL

Only Presubmits
and Postsubmits

Only Periodics

Only Presubmits

Prowjob: How to write yours?

presubmits:
 org/repo:

postsubmits:
 org/repo:

periodics:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

} — Enables addition of this
preset to the job container

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits:
 org/repo:

postsubmits:
 org/repo:

periodics:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

} — Enables addition of this preset to the job container

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

```
presets:
- labels:
    preset-foo-bar: "true"
  env:
    - name: F00
      value: BAR
  volumes:
    - name: foo
      emptyDir: {}
    - name: bar
      secret:
        secretName: bar
  volumeMounts:
    - name: foo
      mountPath: /etc/foo
    - name: bar
      mountPath: /etc/bar
      readOnly: true
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}

interval: 1h

run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []

trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

— Enables podutils

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

<https://docs.prow.k8s.io/docs/components/pod-utilities/>

Prowjob: How to write yours?

presubmits:
 org/repo:

postsubmits:
 org/repo:

periodics:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

Org, repo, branch to clone

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

— Specifies to run on every PR/run request

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

} — Run max this many job instances concurrently

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

}] — Valid Kubernetes pod spec

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

```
interval: 1h
```

— Frequency of the periodic job to run

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

Run if regex condition is met

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

— Skip(or not) setting a Github status

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

} — Status context, same as job name.

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

] — List branches to run/skip, not used together

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

Prowjob: How to write yours?

presubmits: postsubmits: periodics:
 org/repo: org/repo:

```
name: qux-job
labels:
  preset-foo-bar: "true"
decorate: true
extra_refs:
  - org: org
    repo: repo
    base_ref: main
always_run: true
max_concurrency: 10
spec: {}
```

```
interval: 1h
```

```
run_if_changed: "qux/.*"
skip_report: true
context: qux-job
branches: []
skip_branches: []
```

```
trigger: "(?m)qux test this( please)?"
rerun_command: "qux test this please"
```

- Common to ALL
- Only Presubmits and Postsubmits
- Only Periodics
- Only Presubmits

The trigger regex should match the rerun_command for use to trigger via chatops

Test your Prowjob

Local Test: pj-on-kind

- Runs prowjobs as pods in a kind cluster

Local Test: pj-on-kind

- Runs prowjobs as pods in a kind cluster
- **Prerequisites/configuration:**
<https://docs.prow.k8s.io/docs/build-test-update/#running-a-prowjob-locally>

Local Test: pj-on-kind

- Runs prowjobs as pods in a kind cluster

- Prerequisites/configuration:

<https://docs.prow.k8s.io/docs/build-test-update/#running-a-prowjob-locally>

- Minimal configuration:

```
config/jobs/kubernetes/sig-security/govulncheck-presubmit.yaml
```

```
@@ -1,16 +1,12 @@
```

```
presubmits:
```

```
- kubernetes/kubernetes:
```

```
+ ArkaSaha30/kubernetes:
```

```
- name: check-govulncheck-results
```

```
prow/pj-on-kind.sh
```

```
@@ -46,9 +46,9 @@ function parseArgs() {
```

```
  job="${1:-}"
```

```
  config="${CONFIG_PATH:-}"
```

```
  job_config_path="${JOB_CONFIG_PATH:-}"
```

```
- out_dir="${OUT_DIR:-/mnt/disks/prowjob-out}"
```

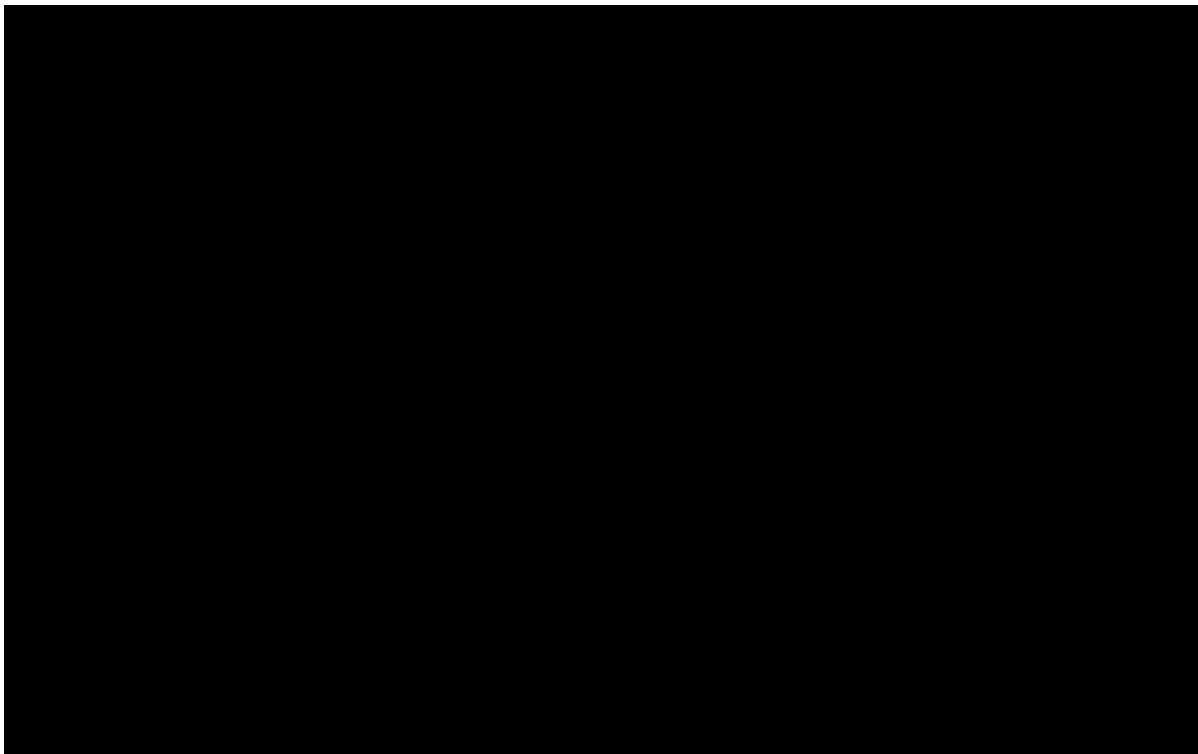
```
+ out_dir="${OUT_DIR:-/Users/arkas1/Documents/mnt/disks/prowjob-out}"
```

```
  kind_config="${KIND_CONFIG:-}"
```

```
- node_dir="${NODE_DIR:-/mnt/disks/kind-node}" # Any pod hostPath mounts
```

```
+ node_dir="${NODE_DIR:-/Users/arkas1/Documents/mnt/disks/kind-node}" #
```

Local Test: pj-on-kind



Key Takeaways

Key Takeaways

- Getting familiar with test-infra
- Automation scopes
- Writing and testing prowjobs
- Reliable and quality code via Prow

Questions?



<https://github.com/ArkaSaha30/kcd-kerala-2024>

Thank You