

# Prow Under the Hood: CI/CD for Kubernetes Development

**Arka Saha**

Software Engineer, VMware by  
Broadcom

**Rayan Das**

Senior SRE, OneTrust



**Supported By**



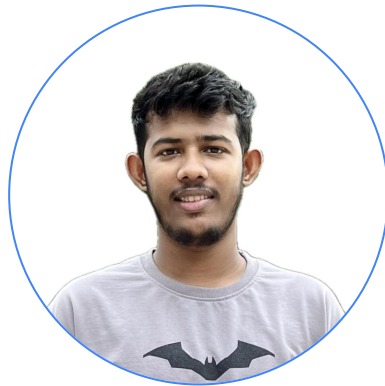
# Who are we?



**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?

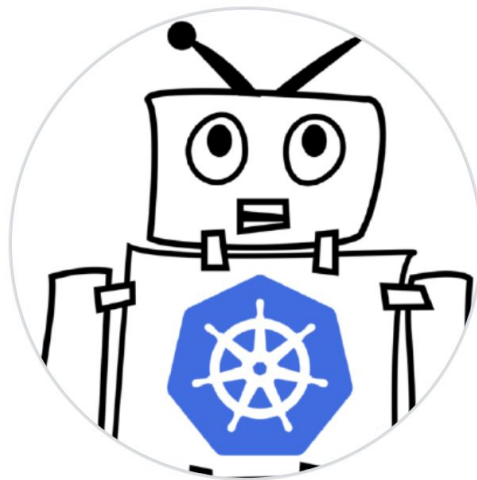
- **Kubernetes based CI/CD System**

# 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:

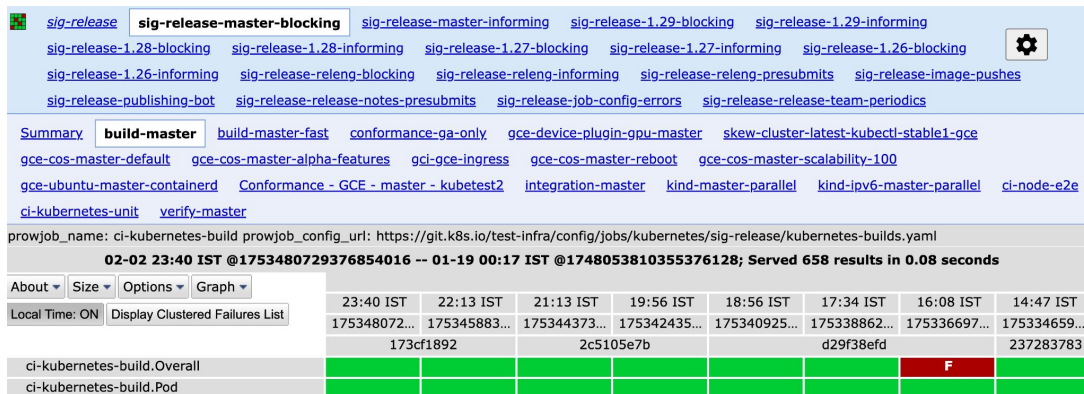


► Details



# What is Prow?

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



The screenshot displays the Prow web interface for a job named 'ci-kubernetes-build'. The top navigation bar includes links for various jobs like 'sig-release', 'sig-release-master-blocking', and 'sig-release-1.29-blocking'. The main content area shows a summary of the job, including the build-master and build-master-fast jobs. Below the summary, there is a table showing the results of the job, including the overall status and the status of individual pods.

Summary: **build-master** build-master-fast conformance-ga-only gce-device-plugin-gpu-master skew-cluster-latest-kubectl-stable1-gce gce-cos-master-default gce-cos-master-alpha-features gci-gce-ingress gce-cos-master-reboot gce-cos-master-scalability-100 gce-ubuntu-master-containerd Conformance - GCE - master - kubetest2 Integration-master kind-master-parallel kind-ipv6-master-parallel ci-node-e2e ci-kubernetes-unit verify-master

prowjob\_name: ci-kubernetes-build prowjob\_config\_url: https://git.k8s.io/test-infra/config/jobs/kubernetes/sig-release/kubernetes-builds.yaml

**02-02 23:40 IST @1753480729376854016 -- 01-19 00:17 IST @1748053810355376128; Served 658 results in 0.08 seconds**

About Size Options Graph

Local Time: ON Display Clustered Failures List

	23:40 IST	22:13 IST	21:13 IST	19:56 IST	18:56 IST	17:34 IST	16:08 IST	14:47 IST
ci-kubernetes-build,Overall	175348072...	175345883...	175344373...	175342435...	175340925...	175338862...	175336697...	175334659...
ci-kubernetes-build.Pod	173cf1892	2c5105e7b				d29f38efd	F	237283783

# What is Prow?

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

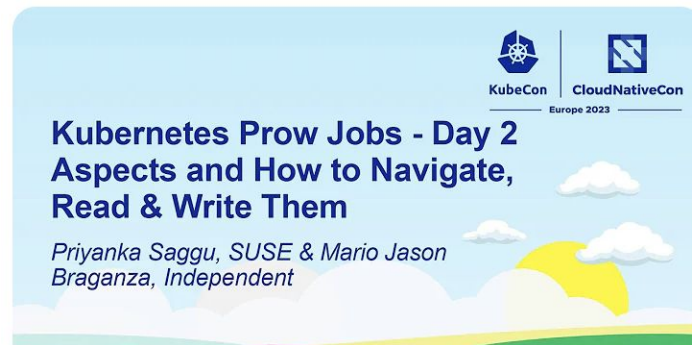
```
periodics:
- interval: 1h
  name: ci-kubernetes-unit
  annotations:
    fork-per-release: "true"
    testgrid-num-failures-to-alert: '3'
    testgrid-alert-stale-results-hours: '24'
    testgrid-days-of-results: '1'
    testgrid-create-test-group: 'true'
    testgrid-dashboards: sig-release-master-blocking
    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

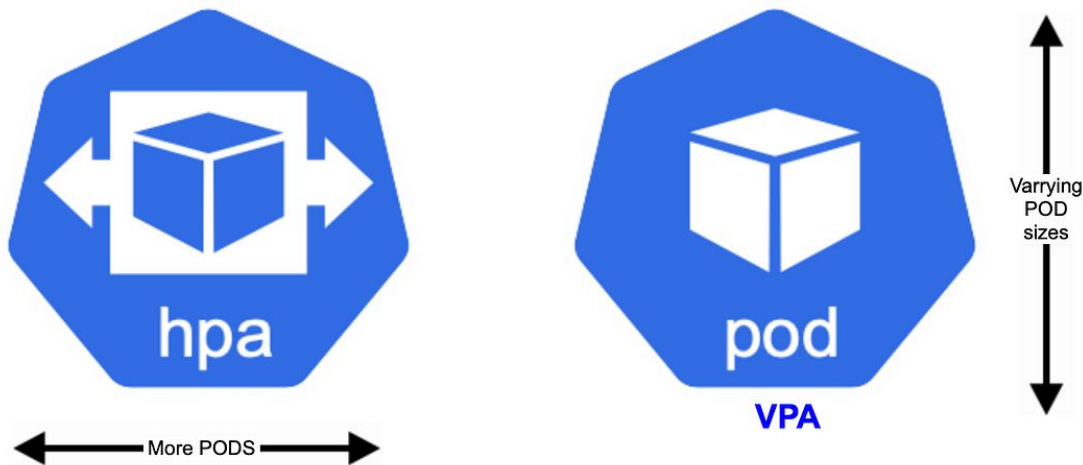


# Why Prow for CI/CD?

- Platform familiarity
- **Reliability**

# Why Prow for CI/CD?

- Platform familiarity
- Reliability
  - Scalability



# Why Prow for CI/CD?

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



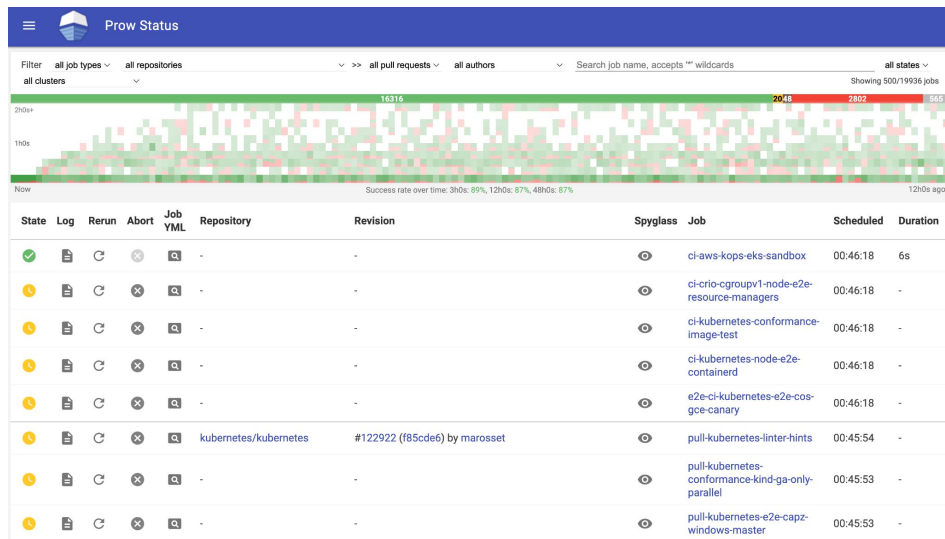
# Why Prow for CI/CD?

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



# Why Prow for CI/CD?

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



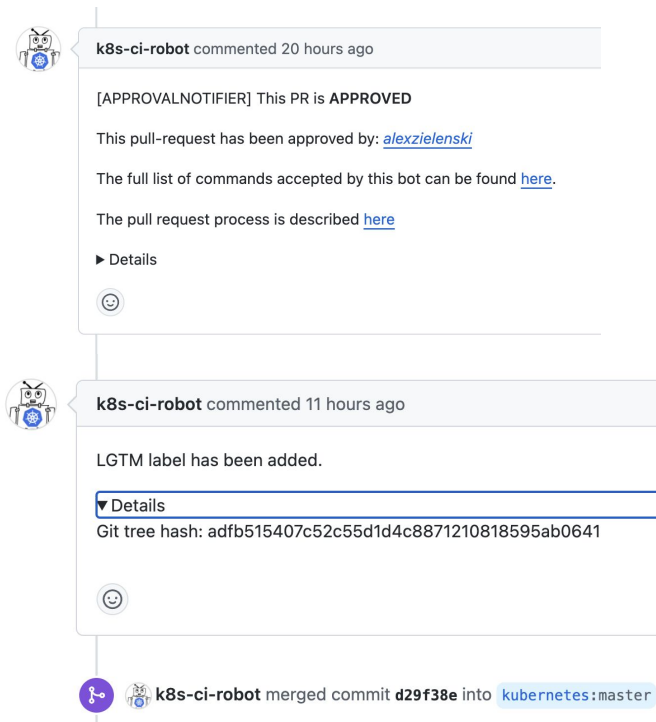
# Why Prow for CI/CD?

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




# Why Prow for CI/CD?

- Platform familiarity
- Reliability
- **Automation**



The screenshot displays a vertical sequence of GitHub Actions log entries. The first entry, from the 'k8s-ci-robot' bot, is a comment posted 20 hours ago. It contains an approval notification stating 'This PR is APPROVED', followed by information about the approving user 'alexzielenski', a link to the bot's command list, and a link to the pull request process. The second entry, also from 'k8s-ci-robot', is a comment posted 11 hours ago stating 'LGTM label has been added.' and includes a collapsed 'Details' section showing a Git tree hash. The final entry shows the 'k8s-ci-robot' bot 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

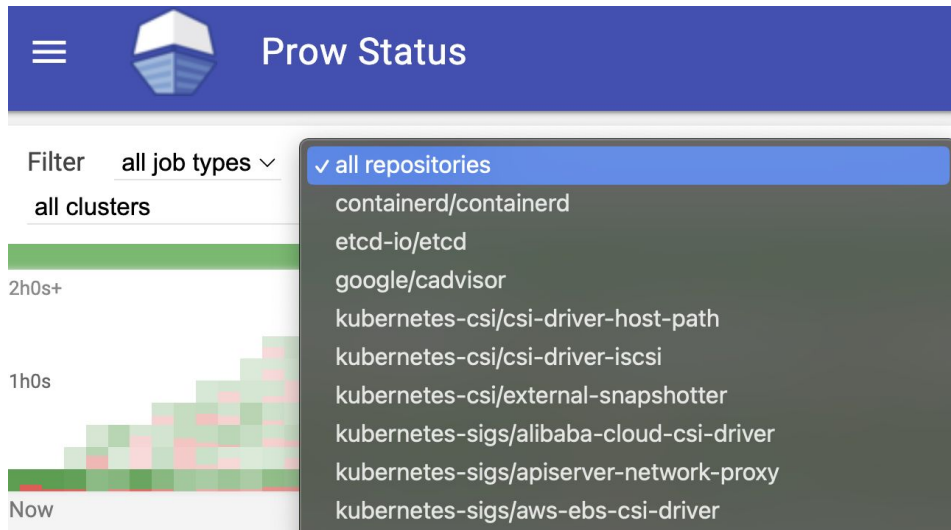
Git tree hash: adfb515407c52c55d1d4c8871210818595ab0641



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

# Why Prow for CI/CD?

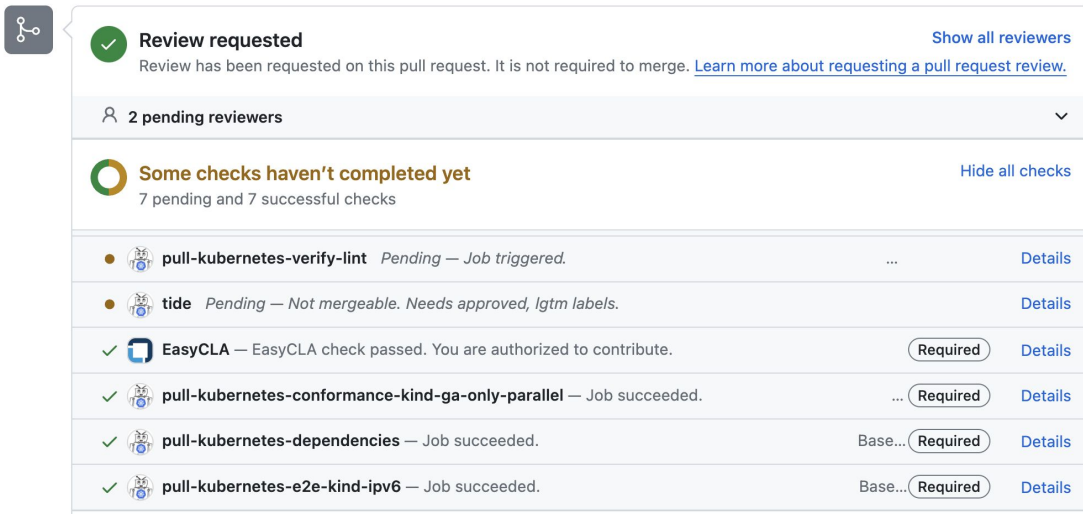
- Platform familiarity
- Reliability
- Automation
- **Multi-org/Multi-repo**



# 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 indicates a 'Review requested' status, with a link to 'Show all reviewers' and a note that review is required for merge. Below this, a section shows '2 pending reviewers'. A yellow circle icon indicates 'Some checks haven't completed yet', with a link to 'Hide all checks' and a note about 7 pending and 7 successful checks. A list of jobs follows, including 'pull-kubernetes-verify-lint' (pending), 'tide' (pending), 'EasyCLA' (passed), 'pull-kubernetes-conformance-kind-ga-only-parallel' (succeeded), 'pull-kubernetes-dependencies' (succeeded), and 'pull-kubernetes-e2e-kind-ipv6' (succeeded). Each job entry includes a status icon, name, description, and a 'Details' link.

**Review requested** [Show all reviewers](#)  
Review has been requested on this pull request. It is not required to merge. [Learn more about requesting a pull request review.](#)

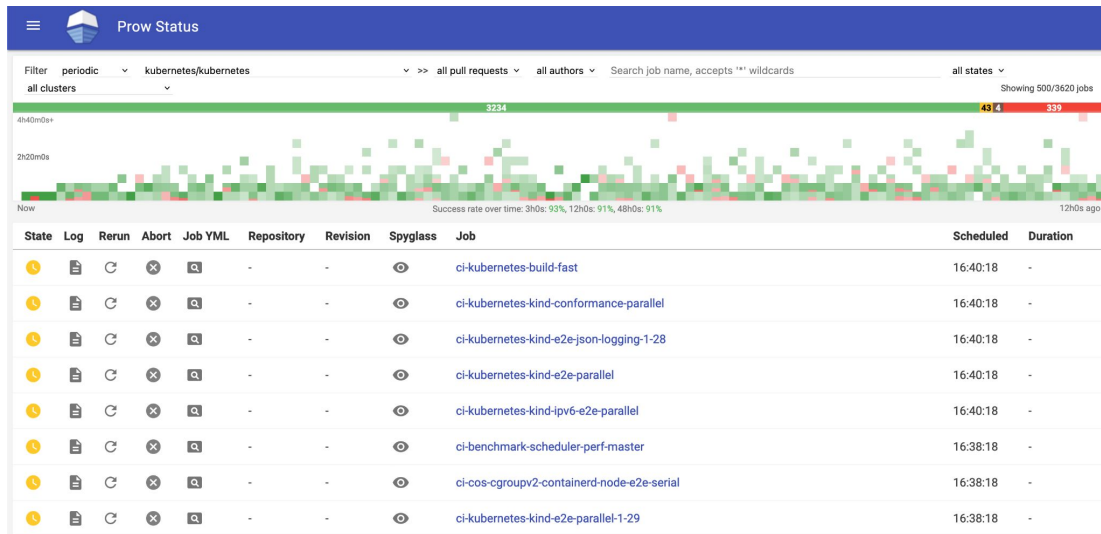
2 pending reviewers

**Some checks haven't completed yet** [Hide all checks](#)  
7 pending and 7 successful checks

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

# Types of Prowjobs

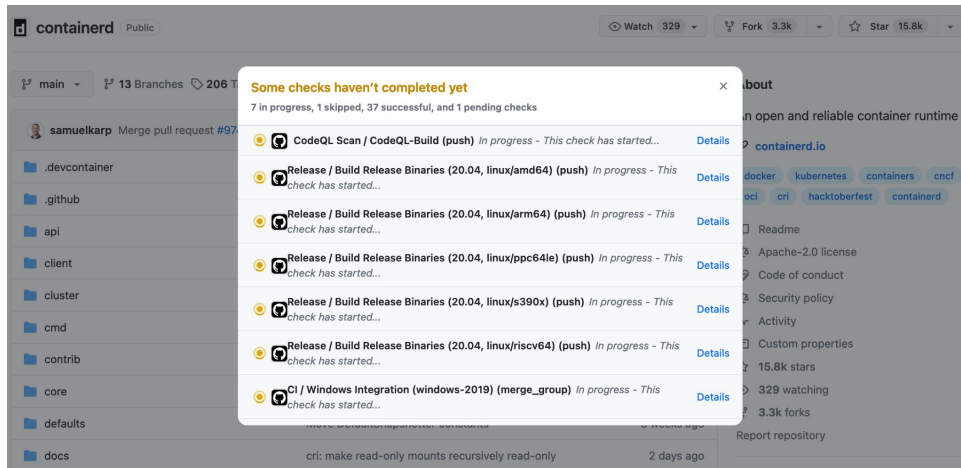
- Presubmits  
(triggered against PRs)
- Periodics  
(triggered recurrently)





# 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}"
```

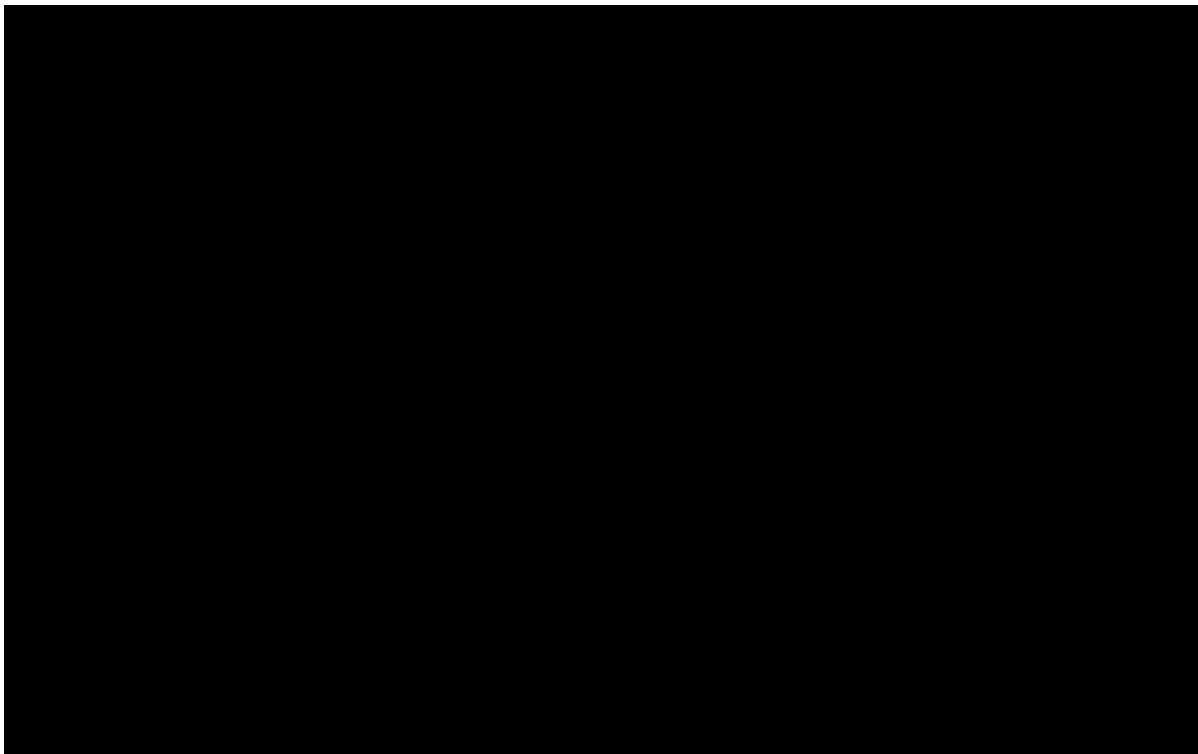
```
+ out_dir="${OUT_DIR:-/Users/arkas1/Documents/mnt/disks/prowjob-out}"
```

```
    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?



# Thank You