

Going Beyond CI/CD with Prow

@leodido



KubeCon



CloudNativeCon

Europe 2020

Whoami!

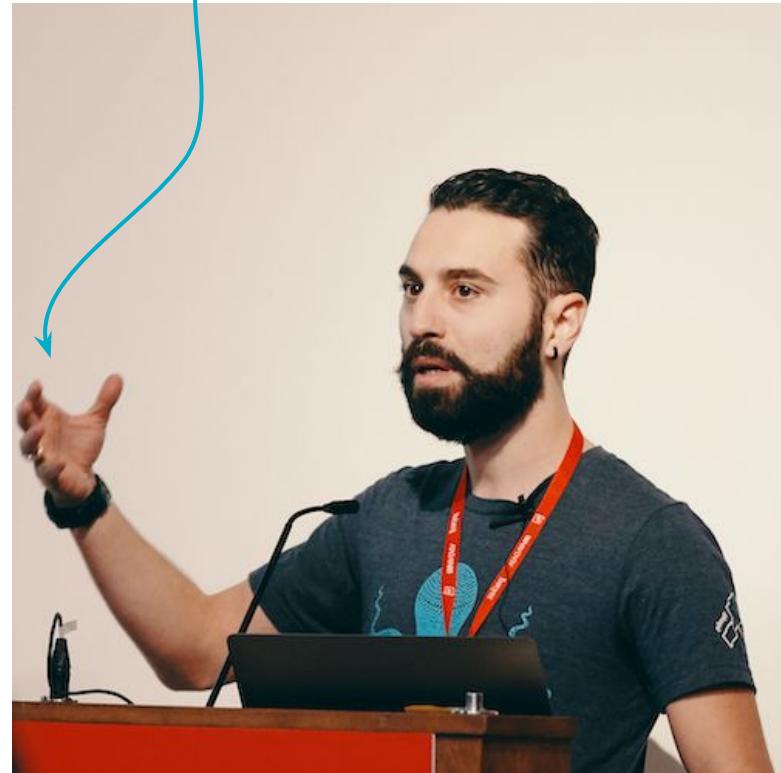
Leonardo Di Donato

*Open Source Software Engineer
Falco Maintainer*

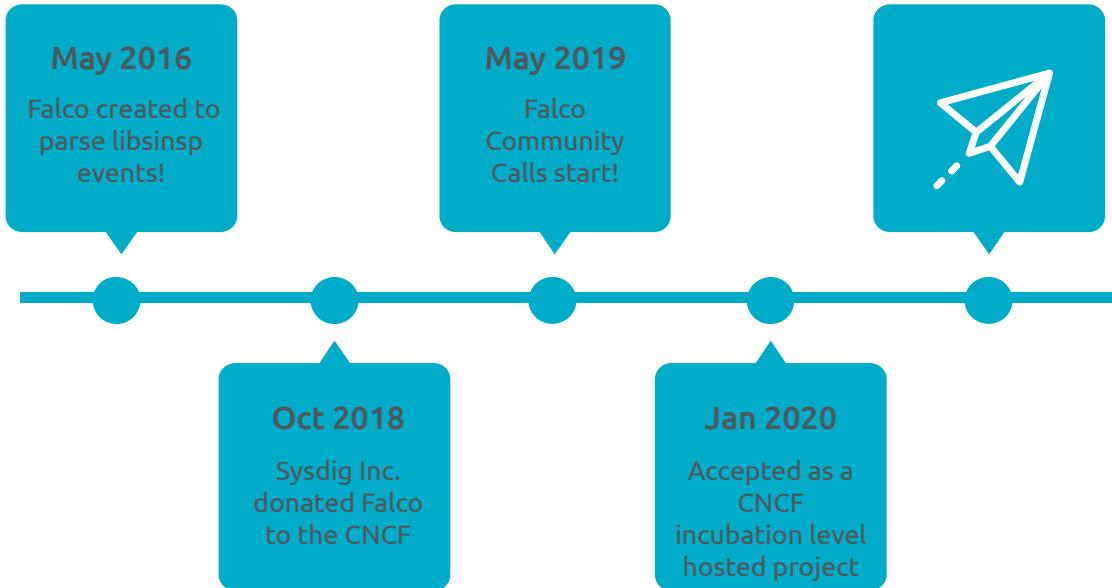
@leodido



extra points to who spots the meaning of this Italian hand-gesture! 😊



A timeline always works fine



Drama!

- ❑ CI was built on top of Travis CI
 - ❑ Barely working
 - ❑ Flaky
- ❑ Falco drivers building & testing infrastructure not public
 - ❑ Internal Jenkins

README.md

Falco

Latest release
v0.15.0 Read the [change log](#)

Dev Branch: [build failing](#)
Master Branch: [build failing](#)
CII Best Practices: [cii best practices](#) In progress 56%
[falcoresecurity / falco](#) [build failing](#)

Current Branches Build History Pull Requests > Build #1732 More options

ebpf eBPF support for Falco

- Commit 4effeefeed
- Compare 4effeefec88b2
- Branch ebpf

[# 1732 errored](#)
Ran for 10 min 38 sec
Total time 19 min 35 sec
2 years ago

Build jobs View config

Build #	Architecture	OS	Compiler	Build Type	Duration
1732.1	AMD64	Trusty	C	BUILD_TYPE=Debug	8 min 57 sec
1732.2	AMD64	Trusty	C	BUILD_TYPE=Release	10 min 38 sec



This is fine.



More drama!

- ❑ PRs merged even when GitHub status checks not passing
- ❑ No clear ownership
- ❑ Missing PRs reviews
- ❑ Some policies there but...
 - ❑ Not easily discoverable & auditable
 - ❑ Zero enforcement
 - ❑ Zero automation
- ❑ The list goes on...

A screenshot of a GitHub pull request list. It shows five merged pull requests (PR #617, #616, #613, #608, #603) that have status check failures. Each PR has a blue arrow pointing to its status check failure message. The messages indicate that the PRs were merged despite failing status checks.

- fix grep rule and ncat rule × area/rules
#617 by dms1lva was merged on 5 Jun 2019 • Changes requested
- fix travis ci badges ×
#616 by was merged on 25 May 2019 • Approved
- rule update: add exception for write below etc × area/rules
#613 by was merged on 30 May 2019 • Approved
- Add instruction for k8s audit support in >= 1.13 × kind/documentation
#608 by was merged on 24 May 2019 • Approved
- changelog missing reference to cri-o container metadata caching ✓
#603 by leodido was merged on 14 May 2019 • Approved

A screenshot of a GitHub pull request page. A specific merge commit (commit 0429883) is highlighted with a blue box. It shows a check failure message: "1 check failed". The commit message indicates that version 0.2.0 was released as part of testing an early version in #537, and can't be overwritten, so it was published as 0.3.0. The pull request has been merged into the dev branch on 21 Feb 2019.

0.2.0 was released as a part of testing an early version in #537, and can't be overwritten, so publish as 0.3.0.

Update version to 0.3.0 ...

merged commit 0429883 into dev on 21 Feb 2019
1 check failed

Merged mstemm merged 2 commits into falosecurity:dev from dms1lva:dev on 5 Jun 2019

Conversation 6 Commits 2 Checks 1 Files changed 1

dms1lva commented on 23 May 2019 • edited • Contributor

fixed the grep binaries list that had egrep instead of egrep.
added the ncalt command line arguments -c and -e that can be used to spawn remote shells.
Launching this: ncalt -l localhost 4443 -e "/bin/sh"
would not match anything when running: sudo sysdig "proc.name = \"/ncat\" and (proc.args contains \"--sh-exec\" or proc.args contains \"--exec\")"
using sudo sysdig "proc.name = \"/ncat\" and (proc.args contains \"--sh-exec\" or proc.args contains \"--exec\" or proc.args contains \"-e\")" fixes this

fix egrep rule and ncalt rule ...

fntnz self-requested a review on 24 May 2019

fntnz added the area/rules label on 29 May 2019

fntnz requested review from mstemm, Kaizhe and leodido on 29 May 2019

mstemm requested changes on 29 May 2019

rules/falco_rules.yaml Outdated

Kaizhe reviewed on 29 May 2019

rules/falco_rules.yaml

add space after arguments, add --lua-exec ...

mstemm commented on 5 Jun 2019

The cause of the test failures was a recent sysdig change that I fixed with #646, and not due to the rules, so I'll go ahead and merge.

merged commit 7a56f1c into falosecurity:dev on 5 Jun 2019

View details

Member

Comments 1

Reviewers: Kaizhe, mstemm, leodido, fntnz

Assignees: No one—assign yourself

Labels: area/rules

Projects: None yet

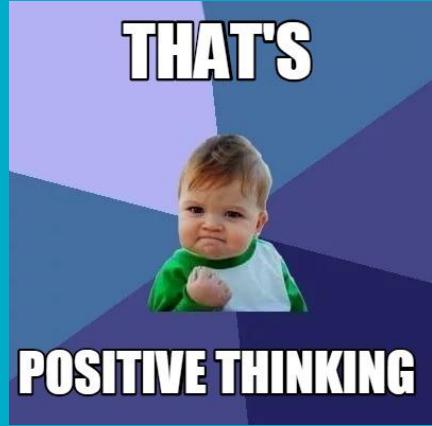
Milestone: No milestone

Linked issues: Successfully merging this pull request may close these issues. None yet

Notifications: You're receiving notifications because you're watching this repository.

4 participants: leodido, fntnz, mstemm, Kaizhe

Lock conversation



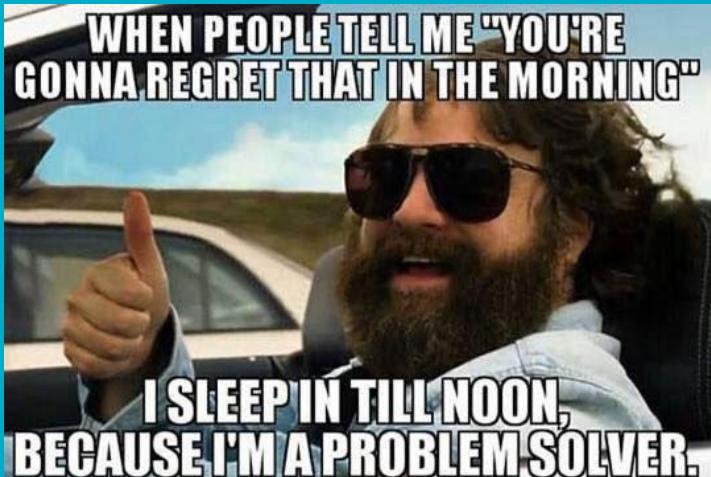
No blame. Just solutions.TM

We had some constraints...

- No time to build a custom CI/CD system from scratch
- No time to create an automatic policy enforcer
- Team of 2

While the main focus was to:

- Write eBPF
- Write C/C++ code



How Kubernetes does?

k8s-ci-bot commented 41 minutes ago

@latteczy: Adding the "do-not-merge/release-note-label-needed" label because no release-note block was detected, please follow our [release note process](#) to remove it.

▶ Details

k8s-ci-bot assigned deads2k and wojtek-t 41 minutes ago

k8s-ci-bot added [do-not-merge/release-note-label-needed](#) [size/S](#) labels 41 minutes ago

k8s-ci-bot commented 41 minutes ago

Thanks for your pull request. Before we can look at your pull request, you'll need to sign a Contributor License Agreement (CLA).

Please follow instructions at <https://git.k8s.io/community/CLA.md#the-contributor-license-agreement> to sign the CLA.

It may take a couple minutes for the CLA signature to be fully registered; after that, please reply here with a new comment and we'll verify. Thanks.

- If you've already signed a CLA, it's possible we don't have your GitHub username or you're using a different email address. Check your existing CLA data and verify that your [email is set on your git commits](#).
- If you signed the CLA as a corporation, please sign in with your organization's credentials at <https://identity.linuxfoundation.org/projects/cncf> to be authorized.
- If you have done the above and are still having issues with the CLA being reported as unsigned, please log a ticket with the Linux Foundation Helpdesk: <https://support.linuxfoundation.org/>
- Should you encounter any issues with the Linux Foundation Helpdesk, send a message to the backup e-mail support address at: login-issues@jira.linuxfoundation.org

▶ Details

prow

Way more than CI/CD.

- GitHub ChatOps
- Define and enforce policies
- Auto-merge bot
- OSS ❤️

Prow

Built for Kubernetes. On Kubernetes.



CRDs

- Custom controllers
- Resource management

Pods

- Scheduling
- Lifecycle
- A bunch of containers

ConfigMaps

- Auditable
- Configurations

Secrets

- Credentials management





Poiana stops the drama!



More time spent in
[Falco](#) coding.

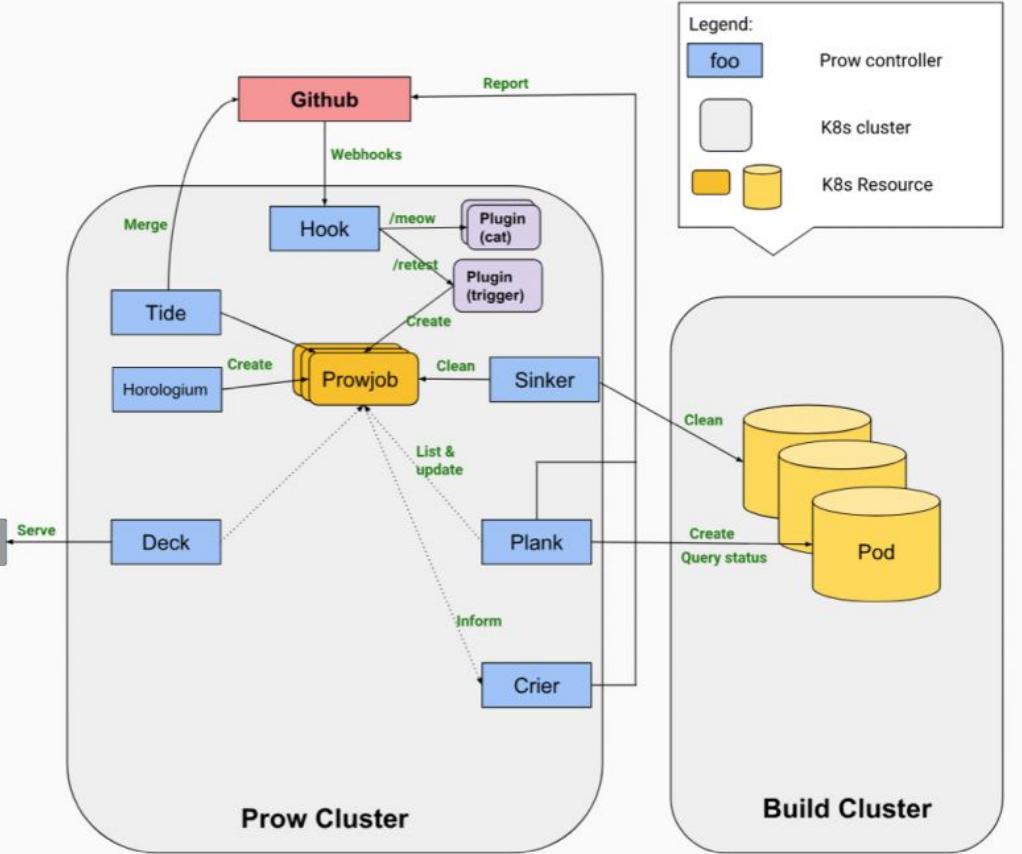
Healthier OSS

Auditable policies.
Clear rules.
Templated processes.
Auto enforcement.

Community

More contributors.
More projects.

Prow Architecture



- ❑ [ProwJob custom resource](#)
- ❑ Horologium
- ❑ Hook
- ❑ Plank
- ❑ Deck
- ❑ Tide
- ❑ Sinker
- ❑ Crier

```

// ProwJobStatus provides runtime metadata, such as when it finished, whether it is
running, etc.
type ProwJobStatus struct {
    // StartTime is equal to the creation time of the ProwJob
    StartTime metav1.Time `json:"startTime,omitempty"`
    // PendingTime is the timestamp for when the job moved from triggered to pending
    PendingTime *metav1.Time `json:"pendingTime,omitempty"`
    // CompletionTime is the timestamp for when the job goes to a final state
    CompletionTime *metav1.Time `json:"completionTime,omitempty"`
    State ProwJobState `json:"state,omitempty"`
    Description string `json:"description,omitempty"`
    URL string `json:"url,omitempty"`
}

// PodName applies only to ProwJobs fulfilled by
// plank. This field should always be the same as
// the ProwJob.ObjectMeta.Name field.
PodName string `json:"pod_name,omitempty"`

// BuildID is the build identifier vended either by tot
// or the snowflake library for this job and used as an
// identifier for grouping artifacts in GCS for views in
// TestGrid and Gubernator. Identifiers vended by tot
// are monotonically increasing whereas identifiers vended
// by the snowflake library are not.
BuildID string `json:"build_id,omitempty"`

// JenkinsBuildID applies only to ProwJobs fulfilled
// by the jenkins-operator. This field is the build
// identifier that Jenkins gave to the build for this
// ProwJob.
JenkinsBuildID string `json:"jenkins_build_id,omitempty"`

// PrevReportStates stores the previous reported prowjob state per reporter
// So crier won't make duplicate report attempt
PrevReportStates map[string]ProwJobState `json:"prev_report_states,omitempty"`
}

// ProwJobSpec configures the details of the prow job.
//
// Details include the podspec, code to clone, the cluster it runs
// any child jobs, concurrency limitations, etc.
type ProwJobSpec struct {
    // Type is the type of job and informs how
    // it's run.
    Type ProwJobType `json:"type,omitempty"`
    // Agent determines which controller fulfills
    // this specific ProwJobSpec and runs the job
    Agent ProwJobAgent `json:"agent,omitempty"`
    // Cluster is which Kubernetes cluster is used
    // to run the job, only applicable for that
    // specific agent
    Cluster string `json:"cluster,omitempty"`
    // Namespace defines where to create pods/resources.
    Namespace string `json:"namespace,omitempty"`
    // Job is the name of the job
    Job string `json:"job,omitempty"`
    // Refs is the code under test, determined at
    // runtime by Prow itself
    Refs *Refs `json:"refs,omitempty"`
    // ExtraRefs are auxiliary repositories that
    // need to be cloned, determined from config
    ExtraRefs []Refs `json:"extra_refs,omitempty"`
    // Report determines if the result of this job should
    // be reported (e.g. status on Github, message in Slack, etc.)
    Report bool `json:"report,omitempty"`
    // Context is the name of the status context used to
    // report back to GitHub
    Context string `json:"context,omitempty"`
    // RerunCommand is the command a user would write to
    // trigger this job via their pull request
    RerunCommand string `json:"rerun_command,omitempty"`
    // MaxConcurrency restricts the total number of instances
    // of this job that can run in parallel at once
    MaxConcurrency int `json:"max_concurrency,omitempty"`
    // ErrorOnEviction indicates that the ProwJob should be completed and given
    // the ErrorState status if the pod that is executing the job is evicted.
    // If this field is unspecified or false, a new pod will be created to replace
    // the evicted one.
    ErrorOnEviction bool `json:"error_on_eviction,omitempty"`

    // PodSpec provides the basis for running the test under
    // a Kubernetes agent
    PodSpec <corev1.PodSpec `json:"pod_spec,omitempty"`

    // JenkinsSpec holds configuration specific to Jenkins jobs
    JenkinsSpec *JenkinsSpec `json:"jenkins_spec,omitempty"`

    // PipelineRunSpec provides the basis for running the test as
    // a pipeline-crd resource
    // https://github.com/tektoncd/pipeline
    PipelineRunSpec *pipelinev1alpha1.PipelineRunSpec
    PipelineRunSpec `json:"pipeline_run_spec,omitempty"`

    // DecorationConfig holds configuration options for
    // decorating PodSpecs that users provide
    DecorationConfig *DecorationConfig `json:"decoration_config,omitempty"`

    // ReporterConfig holds reporter-specific configuration
    ReporterConfig *ReporterConfig `json:"reporter_config,omitempty"`

    // RerunAuthConfig holds information about which users can rerun the job
    RerunAuthConfig *RerunAuthConfig `json:"rerun_auth_config,omitempty"`

    // Hidden specifies if the Job is considered hidden.
    // Hidden jobs are only shown by deck instances that have the
    // --hidden-only=true or --show-hidden=true flag set.
    // Presubmits and Postsplits can also be set to hidden by
    // adding their Repository in Decks "hidden_repo" setting.
    Hidden bool `json:"hidden,omitempty"`
}

```



ProwJobs

- ❑ [Life of a ProwJob](#)
- ❑ [Custom resource](#)
- ❑ [Custom controllers](#)
- ❑ [Types](#)
 - ❑ presubmit
 - ❑ postsubmit
 - ❑ batch
 - ❑ periodic

```

// +k8s:deepcopy-gen:interfaces=k8s.io/apimachinery/pkg/runtime.Object

// ProwJob contains the spec as well as runtime metadata.
type ProwJob struct {
    metav1.TypeMeta `json:",inline"`
    metav1.ObjectMeta `json:"metadata,omitempty"`

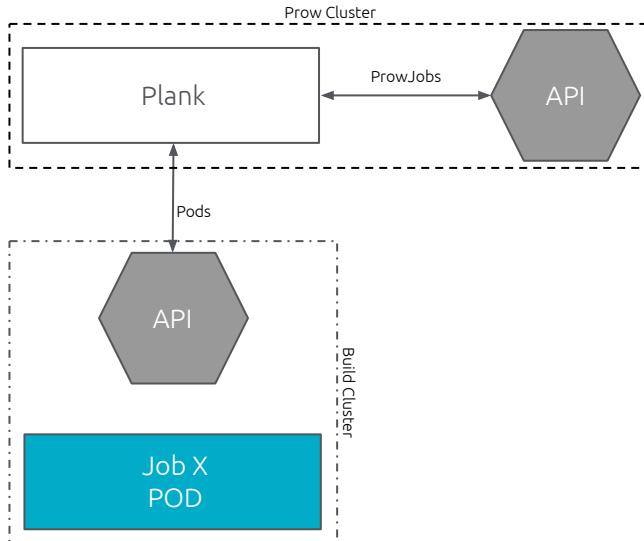
    Spec ProwJobSpec `json:"spec,omitempty"`
    Status ProwJobStatus `json:"status,omitempty"`
}

```



Plank

Controller that manages the lifecycle of jobs.



Horologium

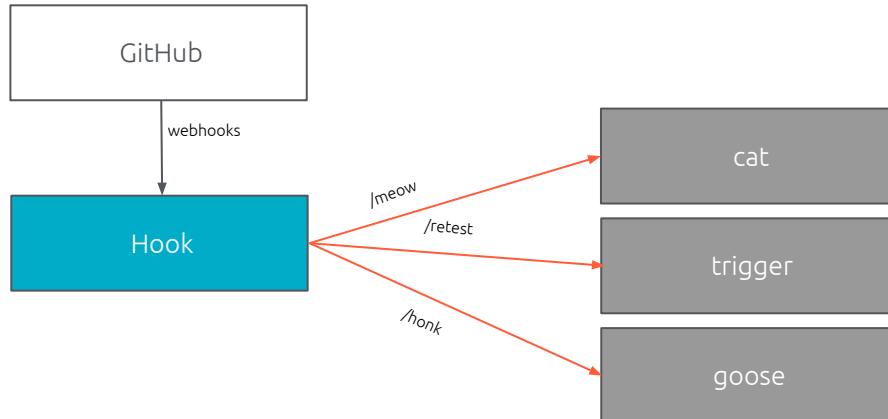
Periodic jobs... [👉 gh:falcosecurity/test-infra/pull/78](https://github.com/falcosecurity/test-infra/pull/78)

```
periodics:
  - name: periodic-stale
    interval: 1h
    decorate: false
    spec:
      containers:
        - image: gcr.io/k8s-prow/commenter:v20191202-824584dd0
          command:
            - /app/robots/commenter/app.binary
          args:
            - |-
              -query=org:falcosecurity
              -label:lifecycle/frozen
              -label:lifecycle/stale
              -label:lifecycle/rotten
              -updated=2160h
              --token=/etc/github/oauth
              -|
              -comment=Issues go stale after 90d of inactivity.
              Mark the issue as fresh with '/remove-lifecycle stale'.
              Stale issues rot after an additional 30d of inactivity and eventually
              close.
            If this issue is safe to close now please do so with '/close'.
            Provide feedback via https://github.com/falcosecurity/community.
            /lifecycle stale
          -template
          -confirm
      volumeMounts:
        - name: oauth
          mountPath: /etc/github
          readOnly: true
      volumes:
        - name: oauth
          secret:
            secretName: oauth-token
```



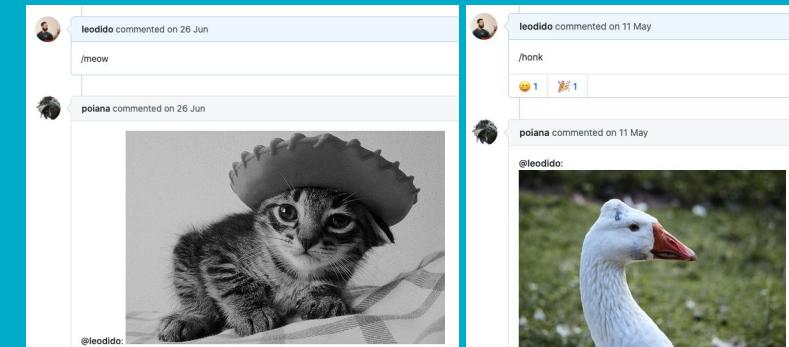
Payload

```
{  
  "id": "10303926682",  
  "sha": "571f8a28e7d338ecac5b12a492df97812e208045",  
  "name": "falcosecurity/falco",  
  "target_url": "https://circleci.com/gh/falcosecurity/falco/3469?utm_campaign=vcs-integration-link&utm_medium=referral&utm_source=github-pr",  
  "avatar_url": "https://avatars2.githubusercontent.com/u/48087?v=4",  
  "context": "ci/circleci: centos7",  
  "description": "Your tests passed on CircleCI!",  
  "state": "success",  
  "commit": {  
    "sha": "571f8a28e7d338ecac5b12a492df97812e208045",  
    "node_id": "MDY6Q29tbWl0NDk5ODYwNDY6NTcxZjhMjh1N2QzMzhLY2FjNWIxMmE0OTJkZjk30DEyZTIwODA0NQ==",  
    "commit": {  
      "author": {  
        "name": "kaizhe",  
        "email": "derek0405@gmail.com",  
        "date": "2020-07-23T23:22:48Z"  
      },  
      "committer": {  
        "name": "polana",  
        "email": "51138685+polana@users.noreply.github.com",  
        "date": "2020-07-25T06:53:06Z"  
      },  
      "message": "add macro user_read_sensitive_file_containers\n\nSigned-off-by: kaizhe <derek0405@gmail.com>",  
      "tree": {  
        "sha": "b19b618b1bfd5d0248ae3323ba3c8e938c2a2ad1",  
        "url": "https://api.github.com/repos/falcosecurity/falco/git/trees/b19b618b1bfd5d0248ae3323ba3c8e938c2a2ad1"  
      },  
      "url": "https://api.github.com/repos/falcosecurity/falco/git/commits/571f8a28e7d338ecac5b12a492df97812e208045"  
    }  
  }  
}
```



Hook

👉 [kubernetes/test-infra/prow/hook](#)



Plugins for everybody

👉 [kubernetes/test-infra/prow/plugins](https://github.com/kubernetes/test-infra/prow/plugins)

- ❑ config_updater
- ❑ trigger
- ❑ approve, [lgtm](#)
- ❑ hold, wip
- ❑ ...

The screenshot shows a GitHub pull request interface with several comments and annotations:

- Comment by leodido:** /hold
to be merged in after #139 (which needs required checks to be changed manually in order to be merged i
- Annotation by poiana:** poiana added size/XS do-not-merge/hold labels on 24 Jun
- Comment by poiana:** @leogr: Updated the configs configmap in namespace default at cluster default using the following files:
 - key configs.yaml using file prow/config/configs.yaml

// We need to empty import all enabled plugins so that they will be linked into any hook binary.

```
import (
    "k8s.io/test-infra/prow/plugins/approve" // Import all enabled plugins.
    "k8s.io/test-infra/prow/plugins/assign"
    "k8s.io/test-infra/prow/plugins/blockade"
    "k8s.io/test-infra/prow/plugins/blunderbuss"
    "k8s.io/test-infra/prow/plugins/branchcleaner"
    "k8s.io/test-infra/prow/plugins/bugzilla"
    "k8s.io/test-infra/prow/plugins/buildifier"
    "k8s.io/test-infra/prow/plugins/cat"
    "k8s.io/test-infra/prow/plugins/cherrypickunapproved"
    "k8s.io/test-infra/prow/plugins/cla"
    "k8s.io/test-infra/prow/plugins/dco"
    "k8s.io/test-infra/prow/plugins/dog"
    "k8s.io/test-infra/prow/plugins/golint"
    "k8s.io/test-infra/prow/plugins/goose"
    "k8s.io/test-infra/prow/plugins/heart"
    "k8s.io/test-infra/prow/plugins/help"
    "k8s.io/test-infra/prow/plugins/hold"
    "k8s.io/test-infra/prow/plugins/invalidcommitmsg"
    "k8s.io/test-infra/prow/plugins/label"
    "k8s.io/test-infra/prow/plugins/lgtm"
    "k8s.io/test-infra/prow/plugins/lifecycle"
    "k8s.io/test-infra/prow/plugins/merge-method-comment"
    "k8s.io/test-infra/prow/plugins/mergecommitblocker"
    "k8s.io/test-infra/prow/plugins/milestone"
    "k8s.io/test-infra/prow/plugins/milestoneapplier"
    "k8s.io/test-infra/prow/plugins/milestonestatus"
    "k8s.io/test-infra/prow/plugins/override"
    "k8s.io/test-infra/prow/plugins/owners-label"
    "k8s.io/test-infra/prow/plugins/pony"
    "k8s.io/test-infra/prow/plugins/project"
    "k8s.io/test-infra/prow/plugins/projectmanager"
    "k8s.io/test-infra/prow/plugins/releasenote"
    "k8s.io/test-infra/prow/plugins/require-matching-label"
    "k8s.io/test-infra/prow/plugins/requiresig"
    "k8s.io/test-infra/prow/plugins/retitle"
    "k8s.io/test-infra/prow/plugins/shrug"
    "k8s.io/test-infra/prow/plugins/signmention"
    "k8s.io/test-infra/prow/plugins/size"
    "k8s.io/test-infra/prow/plugins/skip"
    "k8s.io/test-infra/prow/plugins/slackevents"
    "k8s.io/test-infra/prow/plugins/stage"
    "k8s.io/test-infra/prow/plugins/trigger"
    "k8s.io/test-infra/prow/plugins/updateconfig"
    "k8s.io/test-infra/prow/plugins/verify-owners"
    "k8s.io/test-infra/prow/plugins/welcome"
    "k8s.io/test-infra/prow/plugins/wip"
    "k8s.io/test-infra/prow/plugins/yuks"
)
```



Command help for [all commands](#)

Command	Example	Description	Who Can Use	Plugin
★ /retest	<code>/retest</code>	Rerun test jobs that have failed.	Anyone can trigger this command on a trusted PR.	TRIGGER
★ /test (all)	<code>/test all</code> <code>/test pull-based-test</code>	Manually starts a/all test job(s).	Anyone can trigger this command on a trusted PR.	TRIGGER
★ /retitle	<code>/retitle New Title</code>	Edits the pull request or issue title.	Collaborators on the repository.	RETITLE
★ /lgtm [cancel] or GitHub Review action	<code>/lgtm</code> <code>/lgtm cancel</code> <code>'Approve' or 'Request Changes'</code>	Adds or removes the 'lgtm' label which is typically used to gate merging.	Collaborators on the repository. '/lgtm cancel' can be used additionally by the P author.	
★ /check-dco	<code>/check-dco</code>	Forces rechecking of the DCO status.	Anyone	

Deck

Tide History

Filter [all repositories](#) [all branches](#) >> [all pull requests](#) [all authors](#) [all actions](#) [all status](#)

Pool	Base Commit	Action	Target	Time	Error
✓ falcoresecurity/charts master	3ba32e9	MERGE	#62 (75ac0cb) by Whyeasy	Jul 28 18:32:53	
✓ falcoresecurity/falco master	571f8a2	MERGE	#1330 (1aeebcd) by antoineedeschenes	Jul 28 09:58:42	
✓ falcoresecurity/falco-website master	892bcfd	MERGE	#235 (9dce4f2) by TakaoShimizu	Jul 28 09:58:38	
! falcoresecurity/falco master	571f8a2	MERGE	#1330 (1aeebcd) by antoineedeschenes	Jul 28 09:56:38	failed merging [1330]: [PR is unmergeable. Do the Tide merge requirements match the GitHub settings for the repo? At least approving reviews are required by reviewers with write access.]



Repository	Revision	Job
ci-kubernetes-coverage-e2e-gci-gce		
ci-cri-containedr-windows-build		
ci-kubernetes-e2e-gci-gce-ingress		
ci-kubernetes-e2e-gce-taint-evict		
ci-kubernetes-e2e-gce-alpha-api		
ci-containerd-e2e-ubuntu-gce		
post-test-infra-push-bootstrap	master (ee80a02)	
post-test-infra-bazel		
ci-kubernetes-verify-master		
ci-test-infra-canary-echo-test		



```

tide:
  context_options:
    skip-unknown-contexts: true
    from-branch-protection: true
  merge_method:
    falcosecurity/falco: rebase
  queries:
    - repos:
        - falcosecurity/falco
      labels:
        - approved
        - lgtm
        - "dco-signoff: yes"
    missingLabels:
      - do-not-merge
      - do-not-merge/hold
      - do-not-merge/invalid-owners-file
      - do-not-merge/work-in-progress
      - do-not-merge/release-note-label-needed
      - needs-rebase

branch-protection:
  enforce_admins: true # rules apply to admins too!
  restrictions: # restrict who can push
    teams: ["maintainers", "machine_users"]
  required_pull_request_reviews:
    dismiss_stale_reviews: true # automat dismiss old reviews
    dismissal_restrictions: # allow review dismissals
    teams:
      - maintainers
      - machine_users
    require_code_owner_reviews: true # require a code owner approval
    required_approving_review_count: 1 # number of approvals
  required_status_checks:
    strict: false # we don't want to block any PR if it's not up to date, we have
    the rebase merge strategy and needs-rebase plugin
  orgs:
    falcosecurity:
      # dco check is required for all PRs of any falcosecurity project
      required_status_checks:
        contexts:
          - dco
    falco:
      required_pull_request_reviews:
      required_approving_review_count: 2
      required_status_checks:
        contexts:
          - ... # other required GitHub status check, even not ProwJob ones!

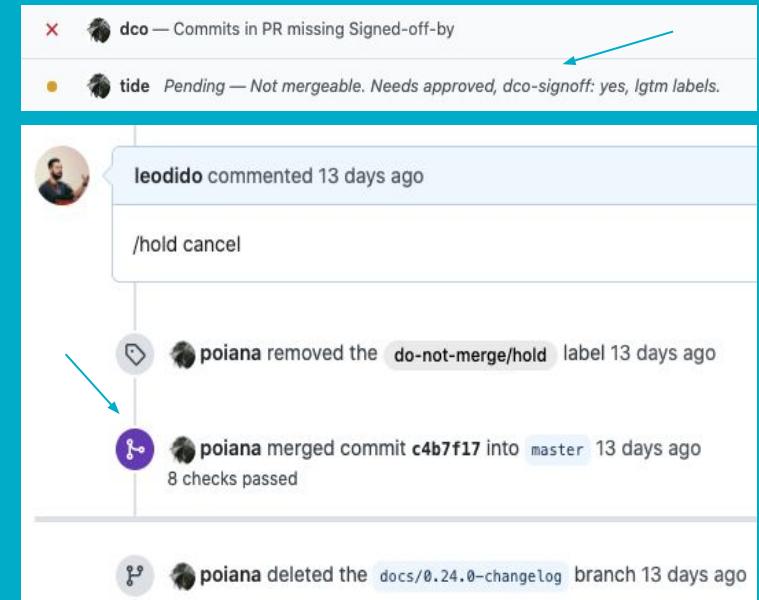
```

Poana requires these labels to put a Falco PR in merge queue!

These prevent a PR on Falco to be merged in!

Overrides!

Tide



Crier

1. Detects ProwJob status changes
2. Reports them
 - Gerrit
 - Pubsub
 - GitHub
 - Slack

📢 Disable report feature of Plank! ([`skip-report=true`](#))
More at  [test-infra/prow/cmd/crier](#)

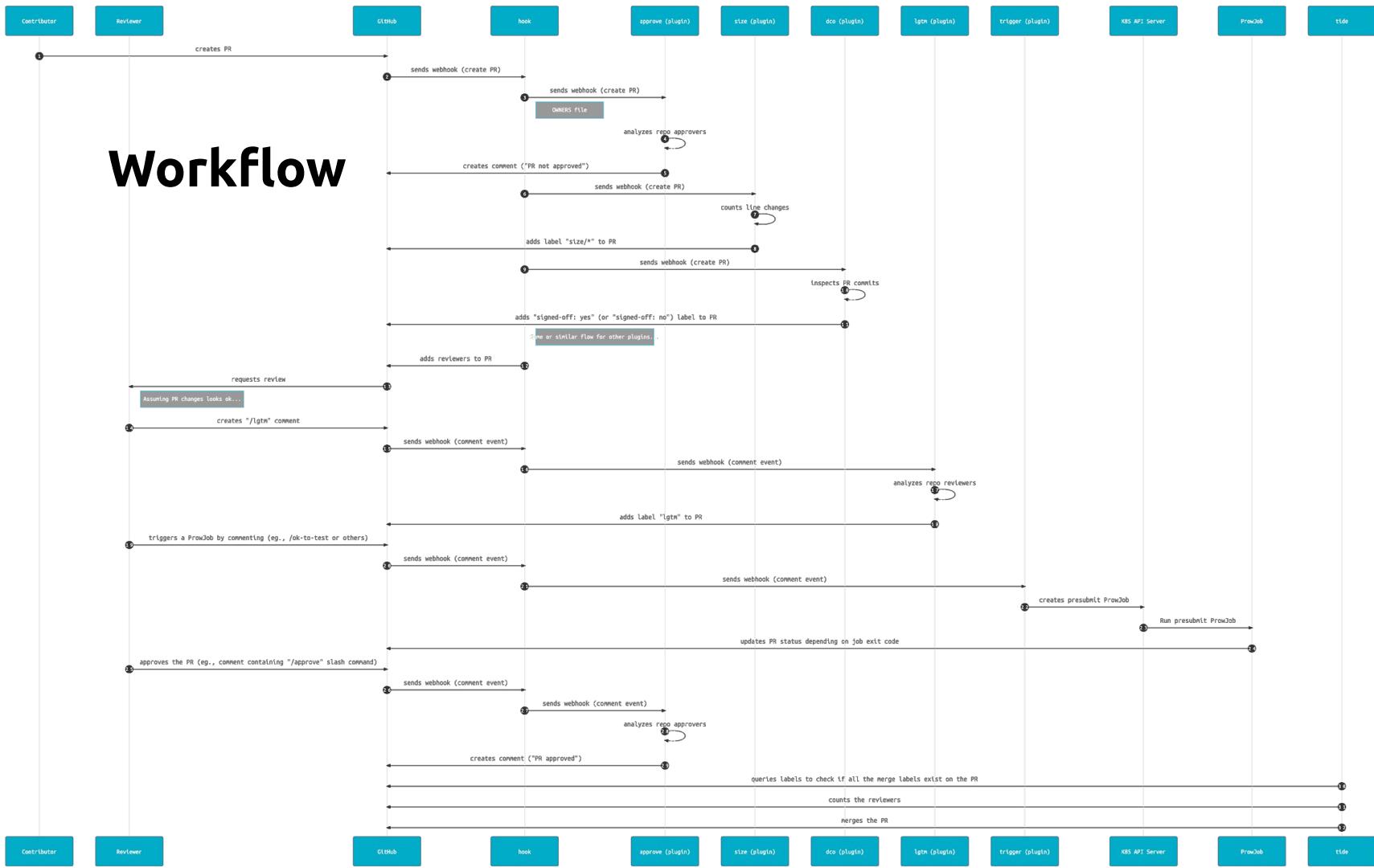
Sinker



garbage collector

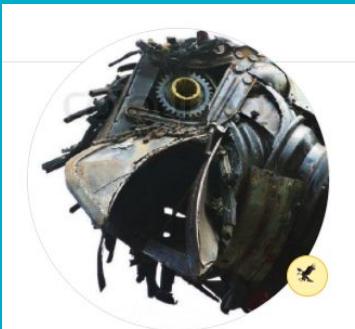
- > 48 hours completed jobs ([customisable](#))
- > 30 min completed pods ([customisable](#))

Workflow



I want it too!

👉 [getting started guide](#)



poiana
poiana

I fly around @falcosecurity projects.
For issues talk with @kris-nova ,
@leodido and @fntlnz

1

2

3

GitHub Bot Account

- Write access to the repos
- Owner access for the org(s)
- Create access token with `repo:status + public_repo` scopes
 - `admin:org_hook` if handling GitHub org

Become a YAML developer

1. Create the YAML manifests, cluster role bindings...
2. Create the secrets from tokens (`hmac + oauth2`)
3. Apply them on your Kubernetes cluster

Install the webhook

go get -u k8s.io/test-infra/experiment/update-hook

```
update-hook --hmac-path=path/to/hmac/secret \
--github-token-path=path/to/oauth/secret \
--hook-url http://1.2.3.4/hook --repo my_organization --confirm=true
```

falcosecurity/test-infra/prow a.k.a @poiana

master test-infra / prow / cluster /

	2 authors and poiana update(prow): compatibility with kubernetes 1.16	...
..		
 0-test.yaml	fix: loving yaml indentation	
 branchprotector.yaml	update(prow): compatibility with kubernetes 1.16	
 configs.yaml	new: specify status checks for branch protection	
 deck.yaml	update(prow): compatibility with kubernetes 1.16	
 hook.yaml	update(prow): compatibility with kubernetes 1.16	
 horologium.yaml	update(prow): compatibility with kubernetes 1.16	
 job.yaml	new: prow custom resource definition	
 needs-rebase.yaml	update(prow): compatibility with kubernetes 1.16	
 plank.yaml	update(prow): compatibility with kubernetes 1.16	
 plugins.yaml	update(prow): compatibility with kubernetes 1.16	
 sinker.yaml	update(prow): compatibility with kubernetes 1.16	
 tide.yaml	update(prow): compatibility with kubernetes 1.16	

```
SHELL := /bin/bash

cluster/oauth-token.yaml: oauth
    kubectl create secret generic --dry-run -o yaml oauth-token --from-file=< >> $@

cluster/hmac-token.yaml: hmac
    kubectl create secret generic --dry-run -o yaml hmac-token --from-file=< >> $@

cluster/unsplash-token.yaml: unsplash_secret
    kubectl create secret generic --dry-run -o yaml unsplash-token --from-file=< >> $@

prow: cluster/oauth-token.yaml cluster/hmac-token.yaml cluster/unsplash-token.yaml
cluster/*.yaml
    kubectl apply -f cluster

plugins: config/plugins.yaml
    kubectl create configmap plugins --from-file=plugins.yaml=config/plugins.yaml --dry-run -o yaml | kubectl replace configmap plugins -f -

configs: config/configs.yaml
    kubectl create configmap configs --from-file=configs.yaml=config/configs.yaml --dry-run -o yaml | kubectl replace configmap configs -f -

clean:
    rm -f cluster/oauth-token.yaml
    rm -f cluster/hmac-token.yaml
    rm -f cluster/unsplash-token.yaml
```



Bring Your Own Prow Plugin

STEPS

1. Write a Go server!
2. Wrap it into a docker image
3. Kubernetes service and deployment
4. kubectl apply -f them!
5. Add it to your Prow cluster

```
external_plugins:  
  <org>/<repo>:  
    - name: <name>  
      endpoint:  
        http://<name>.default.svc.cluster.local  
        events:  
          - issue_comment
```

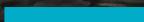
RESOURCES

- ❑ [custom external plugin intro](#)
- ❑ [phony](#) test utility
- ❑ help helpers
 - ❑ [k8s.io/test-infra/prow/pluginhelp](#)
 - ❑ [k8s.io/test-infra/prow/pluginhelp/externalplugins](#)
- ❑ GitHub client and utilities
 - ❑ [k8s.io/test-infra/prow/github](#)
- ❑ config agent and secrets
 - ❑ [k8s.io/test-infra/prow/config/secret](#)
 - ❑ [k8s.io/test-infra/prow/config](#)
- ❑ flag helpers
 - ❑ [k8s.io/test-infra/pkg/flagutil](#)
 - ❑ [k8s.io/test-infra/prow/flagutil](#)



👉 [leodido/saymyname-prow-plugin](https://github.com/leodido/saymyname-prow-plugin) 👈

Read The Source Code





leodido commented now

Member Author 😊 ...

/poiana



poiana commented now

Member 😊 ...

@leodido: Poiana stops the drama!

▶ Details



leodido commented 30 seconds ago

Member 😊 ...

/poiana



poiana commented 28 seconds ago

Member 😊 ...

@leodido: I'm Poiana, I stop the drama!

▶ Details



leodido commented 17 seconds ago

Member 😊 ...

/poiana



poiana commented 15 seconds ago

Member 😊 ...

@leodido: I'm Poiana, I'm here to look at Kris's commit messages...

▶ Details

@leodido



Speculations about the future!

- Enable the [Prow GitHub API proxy](#)
- Falco Drivers Test Grid through ProwJobs
- Code Coverage
- Code Linting
- Custom plugin for externals CI
- Spyglass
- Plugin that automatically creates a release PR when a milestone is done
- Periodic job to cleanup old unused Falco version and prebuilt Falco drivers
- ...



Thanks!

Questions and feedback welcome

-
- ❑ twitter.com/leodido
 - ❑ github.com/leodido
 - ❑ github.com/falcosecurity/falco
 - ❑ slack.k8s.io, #falco channel
 - ❑ github.com/falcosecurity/test-infra
 - ❑ github.com/kubernetes/test-infra
 - ❑ thanks to Apulia ❤ for inspiration

