Kubernetes Operator SIG

See details about this call: https://github.com /open-telemetry/community

Notes from Kubernetes Operator SIG meeting, join our slack channel or visit opentelemetry-operator repository.

Meeting URL: https://zoom.us/j/94709025247?pwd=WVo1WXNSQ2hMRnhnd1FBUzlYQnVVUT09
Meeting recordings:
https://docs.google.com/spreadsheets/d/1SYKfjYhZdm2Wh2Cl6KVQalKg_m4NhTPZqq-8SzEVO6s/edit#gid=0

Nov 20, 2025

Attendees (company):

- Mikołaj Świątek (Elastic)
- Israel Blancas (Coralogix)
- Joe Sirianni (Bindplane)
- Jacob Aronoff (Tero)
- Benedikt Bongartz (Red Hat)
- David Ashpole (Google)
- Pavol Loffay (Red Hat)

Agenda:

- Managed CRD
    - https://github.com/open-telemetry/opentelemetry-operator/pull/4475
- Stabilizing prometheus receiver config: https://github.com/open-telemetry/opentelemetry-collector-contrib/issues/44182
- Pavol Loffay TA feature flag operator.targetallocator.mtls
    - It only makes the secure connection between TA and collector
    - https://github.com/open-telemetry/opentelemetry-operator/tree/main/cmd/otel-allocator#service--pod-monitor-endpoint-credentials
    - Secrets from the SA/pod are exposed in the TA endpoint (not related to the FF)
- Pavol Loffay Instrumentation v1/v1beta1
    - https://github.com/open-telemetry/opentelemetry-operator/milestone/5
    - Avoid conversion webhook - issues with namespace defined in the CRD for the webhook
    - CRDs should be installed before the operator?
        - https://github.com/open-telemetry/opentelemetry-helm-charts/issues/1184
    -
- [all] Issues to discuss at sig (always last)

Nov 6, 2025

Attendees (company):

- Benedikt Bongartz (Red Hat)
- Pavol Loffay (Red Hat)
- Mikołaj Świątek (Elastic)

Agenda:

- TBD
- [all] Issues to discuss at sig (always last)

Oct 23, 2025

Attendees (company):

- Mikołaj Świątek (Elastic)
- Antoine Toulme (Splunk)
- Benedikt Bongartz (Red Hat)
- David Ashpole (Google)
- Jacob Aronoff
- Pavol Loffay (Red Hat)

Agenda:

- Go through feature gates and check if any need to move forward or backward.
    - Move the sidecar featuregate to stable
        - https://github.com/open-telemetry/opentelemetry-operator/issues/4451
    - Gomemlimit & maxprocs
        - https://go.dev/blog/container-aware-gomaxprocs
        - Let's enable this by default
        - https://github.com/open-telemetry/opentelemetry-operator/issues/4452
    - Config defaulting
        - Let's move it to Stable.
        - https://github.com/open-telemetry/opentelemetry-operator/issues/4453
- Start a milestone for beta Instrumentation (https://github.com/open-telemetry/opentelemetry-operator/milestone/5)
    - Track instrumented workload
        - https://github.com/open-telemetry/opentelemetry-operator/issues/544
        - https://github.com/open-telemetry/opentelemetry-operator/issues/1142
        - https://github.com/open-telemetry/opentelemetry-operator/pull/1228
        -
- [all] Issues to discuss at sig (always last)

Oct 9, 2025

Attendees (company):

- Jacob Aronoff
- Timo Johner (SAP)
- David Ashpole (Google)
- Yuri Oliveira (OllyGarden)
- Pavol Loffay (Red Hat)

Agenda:

- [dashpole] I'm your new TC liaison!
- Adding spec.HostPID for all collector modes PR #4280 (picking up from 28th Aug)
    - Elastic may have something to say about this?
    - Privileged: true
    - Is there a way to get the audit logs without setting the hostPID: true? Or do it with the security context?
- [all] Issues to discuss at sig (always last)

Sep 25, 2025

Attendees (company):

- **matyas.vegh@ericsson.com (Ericsson)**
- Jacob Aronoff
- Mikołaj Świątek (Elastic)
- Antoine Toulme (Splunk)
- Pavol Loffay (Red Hat)

Agenda:

- Discussion about usage of Finalizers and side effects on namespace deletion. See also Finalizer in the open telemetry collector CustomResource prevents successful namespace deletion when managed operator is uninstalled prior to CustomResource uninstall/deletion · Issue #4367 · open-telemetry/opentelemetry-operator
  [matyas] Finalizers should be conditional on whether cluster-resources are actually used rather than unconditional as they are now. It would be possible to have an ownerReference from the operator's own clusterrrole(binding) to the created clusterrole(binding)s. That way, complete removal would clean up generated clusterrole(binding)s, but not if only the CR is deleted. Tracking the responsible CR would be viable, but requires more engineering.
- Discuss https://github.com/jinja2/opentelemetry-operator/blob/test/docs/cluster-observability.md
- [all] Issues to discuss at sig (always last)

Sep 11, 2025

Attendees (company):

- Pavol Loffay (Red Hat)
- Mikołaj Świątek (Elastic)
-

Agenda:

- Collector has an RFC about configuration merging using yaml tags, and they wanted to know if that affects the operator.
-

Aug 28, 2025 9 AM PT/6 PM CET

Attendees (company):

- Benedikt Bongartz (Red Hat)
- Mikołaj Świątek (Elastic)
- Vincent Desbois (Ericsson)
- Jacob Aronoff
- Pavol Loffay (Red Hat)
- Simon Olander (SAP)

Agenda:

- Skip release v0.132.0
  - https://github.com/open-telemetry/opentelemetry-operator/pull/4320
  - Dotnet auto-instr. Failing?

- Discuss Support range of OpenTelemetry Collector versions · Issue #4307 · open-telemetry/opentelemetry-operator
- Feedback on hostPID addition https://github.com/open-telemetry/opentelemetry-operator/issues/4214, https://github.com/open-telemetry/opentelemetry-operator/pull/4280
  - Try out to see if a work around is to run a Deployment with sidecar injection that has the correct settings (hostPID, securityContext etc).
  - Auditlogs part of Managed CRD?
    - https://github.com/open-telemetry/opentelemetry-operator/issues/3818
      - Config:
        - Platform:
          - Logs:
            - Audit: enabled
          - Metrics:
            - Hostmetrics: enabled
          - Traces:
        - Applications:
          - Logs:
            - Selector: => namespace: abc
        - Instrumentation:
        - Export:
          - Otlp:
            - endpoint: something

```
apiVersion: v1
kind: Pod
metadata:
  labels:
    sidecar.opentelemetry.io/inject=true
status:
  containerStatuses:
  - ready: true
    started: true
  phase: Running
spec:
  hostPID: true
```

Aug 14, 2025 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Antoine Toulme (Splunk)
- Jacob Aronoff

Agenda:

- Discuss the intent of https://github.com/open-telemetry/opentelemetry-operator/pull/4201
- Discuss the managed CR implementation
  - Have the CR spawn CRs
    - Preferred as we have hooks for upgrades and maintenance.
  - Have the CR run the whole deployment

- Discuss https://github.com/open-telemetry/opentelemetry-operator/issues/3340 and a possible solution https://github.com/open-telemetry/opentelemetry-operator/compare/main...jaronoff97:opentelemetry-operator:instrumentation-refactor

Jul 31, 2025 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Antoine Toulme (Splunk)
- Jacob Aronoff

Agenda:

- Review feature gate stability
- [all] Issues to discuss at sig (always last)

Jul 31, 2025 9 AM PT/6 PM CET

Attendees (company):

- Benedikt Bongartz (Red Hat)
- Jacob Aronoff
- Pavol Loffay (Red Hat)
- Yuri Oliveira (OllyGarden)

Agenda:

- [Antoine, cannot attend in person, will review recording] Looking for review on https://github.com/open-telemetry/opentelemetry-operator/pull/4201
    - [jacob] im not opposed to embedding a global instrumentations object in the operator YAML
        - I don’t love the DRY stuff
    - [pavol] there is some precedence for this – have heard from users they want cluster scoped resources
        - We will need to heavily consider precedence/duplicate targets and how that will work
    - [yuri] this also relates to precedence for namespace vs pod instrumentation annotations, how will this work with that?
    - [yuri] maybe we should change the helm chart to have a deliberate template for instrumentation with the operator install
    - [jacob] (going off of yuri’s point) if the goal is to be a single install that gets you instrumentations, we should change the helm charts to do extra manifests like the otel collector chart
    - [pavol] would it instead be better to have a “global instrumentation config” that we embed in the operator config, not a list of instrumentations to install
    - [yuri] using the otel-kube-stack as an example, can we not install these things altogether?
- [Pavol Loffay] Network policies for operator and operands https://github.com/open-telemetry/opentelemetry-operator/issues/4231
    - [jacob] definitely for it, can we just install it as part of the bundle rather than the operator creating it itself?
        - [pavol] OLM doesn’t support network policies as of now, so we need the operator to make it
        - [pavol] also we’ll need this code for when we want to support collector network policies

    - [yuri] do we have any concerns about the cloud constraints? In azure or AWS, we don’t see any issues usually, but in GCP we had some extra work to get it running correctly
        - [pavol] it’s probably the webhook that’s problematic
    - [pavol] Operator will use a featuregate for itself to auto-install this
        - CRDs will have a flag (initial default false) to create network policies
        - Create a featureflag to eventually set default to true
    - [jacob] it would be good to have an e2e to verify failure and success
- Review feature gate stability
- [all] Issues to discuss at sig (always last)

Jul 17, 2025 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Elastic)
- Benedikt Bongartz (Red Hat)
- Yuri Oliveira (OllyGarden)

Agenda:

- [Mikołaj] Let’s resolve the HTTP semantic convention breaking change somehow. https://github.com/open-telemetry/opentelemetry-operator/issues/2542
    - Not upgrading the Java/.NET versions but using latest versions for new CRDs is something we can do without much risk
- [Mikołaj] Run E2E test with the latest contrib image daily and report errors in an issue
    - We’ve had two release blocking bugs recently that weren’t caught upstream. Both of them would’ve been caught by the operator’s E2E tests.
    - Mikołaj Świątek will create an issue to do this
    - We should also bump the collector version asap, using renovate
- Review feature gate stability
- [all] Issues to discuss at sig (always last)

Jul 3, 2025 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Elastic)

Agenda:

- Resolving the breaking change in http semantic conventions #2542
- Review feature gate stability
- [all] Issues to discuss at sig (always last)

Jun 19, 2025 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Mikołaj Świątek (Elastic)

Agenda:

- Review feature gate stability
- [all] Issues to discuss at sig (always last)

Jun 5, 2025 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Elastic)
- Imma Valls (Grafana Labs)
- Antoine Toulme (Splunk)

- Jacob Aronoff (Omlet)
- Pavol Loffay (Red Hat)

Agenda:

- [Mikołaj] Use gotestsum as a test runner (PR)
  - ✅
- [Mikołaj] Add a test report for E2E tests (example)
  - ✅
- [Mikołaj] Run E2E tests on instrumentation images built from the PR branch (PR)
  - ✅
- [Antoine] opAmpBridge CRD missing - see PR
  - ✅
- [Antoine] Road to harmonized config - see PRs
  - https://github.com/open-telemetry/opentelemetry-operator/pull/4052
  - https://github.com/open-telemetry/opentelemetry-operator/pull/4053
  - https://github.com/open-telemetry/opentelemetry-operator/pull/4054
  - https://github.com/open-telemetry/opentelemetry-operator/pull/4055
  - https://github.com/open-telemetry/opentelemetry-operator/pull/4056
- [Antoine] migrate to Otel metrics (PR)
- [Evan] What's the motivation for using Prometheus Operator CRDs to configure the Target Allocator?

May 22, 2025 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Elastic)
- Luca Lanziani (Nearform)
- Benedikt Bongartz (Red Hat)
- Israel Blancas (Coralogix)

Agenda:

- Replace instances of gopkg.in/yaml.v3
  - We're good using this library
  -

May 8, 2025 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Omlet)
- Mikołaj Świątek (Elastic)
- Antoine Toulme (Splunk)
- Pavol Loffay (Red Hat)
- Alex Arnell (Heroku / Salesforce)
- Benedikt Bongartz (Red Hat)
- Israel Blancas (Coralogix)

Agenda:

- [jacob] target allocator as collector extension (link)
  - [antoine] helm chart is up
    - Relatively lean go code and can run on its own, could run on its own as a collector extension
    - End of feb/march became a maintainer
    - Doesn't want another person to feel the burden

        - Idea is to test it out and see if it makes sense
        - Not looking to replace the TA from the operator right now
        - Wants to chart a path to see if it could be a part of the collector
        - Could also use the leader election extension to do federation, one option is leader election
        - May improve code-reuse
    - [mikołaj] nothing necessarily wrong with doing this as an extension
        - Concern:
            - How do we manage sharing the code i.e. what is the relationship between the code and how do we prevent code drift
            - How does configuration work? Is the configuration combinable both ways? Does one depend on another? Will we do automatic rewriting?
        - Would happily accept a port from client-go to otel-sdk
    - [jacob] maybe we could move it to the /cmd dir in contrib like the supervisor
        - This way we could run it in the same binary
    - [antoine]
        - In contrib we get some codecov, better lifecycle management, stability is more present, etc.
        - Would probably make sense to move it over for that reason
- [mikołaj] Instrumentation automatic configuration proposal needs operator maintainer engagement
    - Due diligence document here
    - [antoine]
        - A lot of this is michele's work thats going to be impactful
        - Already a draft PR for the Zig approach using LD preload
        - This would overhaul the approach we do from the operator
        - Solves a long standing operator issue
        - LD_Preload makes sense for this
        - Operator defines it as a dependency
        - Long process / roadmap event
    - [jacob]
        - Should this be a feature flag? I think yes
        - Should this be in the config spec? I think no...
        - IMO i think it should be a feature flag that a user opts into initially and if they set their own image it's on them to use it
        - Clauses:
            - Isolated repo
            - Opt-in for users until a v1beta1 for instrumentation?
            - If a user doesn't use a custom auto-instrumentation image, we can gradually migrate them to the new method
            - We delete the old method when we do v1beta1
            - Incredibly extensive e2e tests
            - [antoine] eventually this results in spec requirements for languages, if you want to work with LD preload you MUST do these things.
    - [mikołaj]
        - We don't give any guarantees for users building their own images
        - It's going to be a breaking change, but the blast radius is hopefully small

- [mikołaj] Do we want to bump the K8s requirement to 1.25? Our policy permits it.
    - [jacob] yes. Beyond time.
    - Openshift colleagues will get back to us
    - We want to use CEL for validations
- [antoine] TA helm chart (link)
- [mikołaj] Massive thank you to Ishwar for all the great tests for instrumentation!! Hooray!!
- [all] Issues to discuss at sig (always last)

Apr 24, 2025 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Omlet)
- Pavol Loffay (Red Hat)
- Josh Voravong (Splunk)
- Mikołaj Świątek (Elastic)
- Yuri Oliveira (OllyGarden)

Agenda:

- [all] Issues to discuss at sig (always last)
    - Downgrade mechanism?

Apr 10, 2025 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Elastic)
- Antoine Toulme (Splunk)
- Hans Kristian Flaatten (Nav)
- Benedikt Bongartz (Red Hat)
- Pavol Loffay (Red Hat)
- Yuri Oliveira (OllyGarden)

Agenda:

- Prometheus remote write exporter, memory leaks
    - Use OTLP with Prometheus https://prometheus.io/docs/guides/opentelemetry/
- Managed CRD https://github.com/open-telemetry/opentelemetry-operator/pull/3830
    - Weaver discussion https://github.com/open-telemetry/weaver
- [all] Issues to discuss at sig (always last)

Apr 4, 2025 KubeCon London, 11:30 BST

Agenda:

- Medium-term roadmap
    - Improve documentation
        - Documentation for the most popular use-cases
            - Target Allocator, how to collect prometheus metrics
                - Can we rename (in documentation) to e.g. prometheus bridge?
                    - Maybe even the CRD?
            - Collecting logs
            - TLS
            - Traces from instrumentation
                - Tailsampling
            - Missing docs for RBAC of the components
                - For all components in the kubernetes distro
            - Reference architectures for deployments

        - Stabilize the Instrumentation CRD
            - Add semantic tests
                - [jacob] I think we could use weaver validate here and run a server that is being sent the OTLP data and validating against what the language instrumentations define for semconv
                - [jacob] I will work with the project incra sig that i am a part of to recommend that they begin to have the semconv definitions in their repos
                    - We could make this easier with some totaling that checks various instrumentation or <lang>-contrib for the instrumentations to automate here
                    - FWIW i think the languages would want this themselves
                    - I think we should check it for stable semconv
                - [mikołaj] at the very least we should write some basic asserts on the data to check that we get *anything* out. It's not our problem if something there is broken.
                    - [jacob] I do think that the languages should maintain semconv adherence and it would be good for the project to do this. Not the operator's scope, but I (Jacob) can look into this.
            - Use a label instead of an annotation
                - [jacob] yes this should be a requirement
                    - We need to change the mutating webhook configuration
                    - Relevant proposal
                    - Found the performance issue
                    - Other issue
            - Make the pod mutation function pure
                - [jacob] I would very much like to simplify this code, i've begun a POC of this
            - Resolve the HTTP semantic convention breaking change
                - [jacob] Could we couple this into the v1beta1 change?
            - Involve the SDK SIGs in maintaining the auto-instrumentation images
                - [jacob] I would love to ask the TC for guidance/assistance here, given Juraci is our sponsor, we should ask him
        - Make the operator more observable
            - Add a schema for existing metrics
                - [jacob] yes
                - [mikołaj] use weaver, unsure if we'll use the auto-generated stuff
            - Use the Otel SDK for metrics, then also for logs
                - [jacob] yes, it's a bit more than a "good-first-issue" but a bit less than a massive project
            - Use tracing?
                - [jacob] yes, kubernetes itself has otel tracing and should have support for this
            - [jacob] Should we have a CRD for operator configuration? Many other operators do this so they can set things via a config file (istio)
                - [jacob] we could use viper to assist with this
- [Mikołaj] We need to resolve https://github.com/open-telemetry/opentelemetry-operator/issues/3855 before 0.122.0

Mar 27, 2025 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Omlet)
- Pavol Loffay (Red Hat)
- Israel Blancas (Coralogix)
- Mikołaj Świątek (Elastic)
- Raphaël Thériault (SolarWinds)

Agenda:

- [raph] Request for second look at https://github.com/open-telemetry/opentelemetry-operator/pull/3416
- [jacob] antoine's RFC (link)
  - Let's all review before KubeCon
- [Pavol] https://github.com/open-telemetry/opentelemetry-collector-contrib/issues/38682
- [all] Issues to discuss at sig (always last)

Mar 13, 2025 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Omlet) is OOO
- Israel Blancas (Coralogix)
- Antoine Toulme (Splunk)
- Xuan Cao
- Josh Voravong (Splunk)

Agenda:

- **Friends, it seems many maintainers are out today...**
- [Mikołaj] Are we ok with this breaking change without the ability to switch back to old behaviour: https://github.com/open-telemetry/opentelemetry-operator/pull/3797?
  - From SIG:
    - We agreed about suggesting adding a feature gate and promote to stable after some time.
    - Add a new entry to the changelog. Also, provide some recommendations about how to upgrade?
    - Were these changes released already?
    - Is this something we can do in a way that we don't break users? Or in a gradual way?
- [Mikołaj] There's a proposed Ruby autoinstrumentation: https://github.com/open-telemetry/opentelemetry-operator/issues/3762
  - How do we want to go about accepting this?
  - My proposal:
    - Approvers of ruby instrumentation should own the docker image definition
    - There should be a test verifying that it actually sends data
    - There should be a defined process for bumping the version
- [all] Issues to discuss at sig (always last)

Feb 27, 2025 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Elastic)
- Jacob Aronoff (Omlet)
- Tyler Helmuth (Honeycomb)
- Pavol Loffay (Red Hat)

- Michele Mancioppi (Dash0)

Agenda:

- [Michele] Injector PoC
- [Tyler] Helm chart bump (link)
  - What is the relationship between the TA feature gate and the TA CRD in v0.118.0
- [Mikołaj] Prometheus upgrade to 0.30x (#3739)
  - Prometheus apparently did some breaking changes to scrape protocols, does this affect us?
  - This ties into prometheus-operator upgrades, which exposed an issue in target-allocator
- [all] Issues to discuss at sig (always last)

Feb 13, 2025 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Elastic)
- Jacob Aronoff (Omlet)
- Josh Voravong (Splunk)
- Pavol Loffay (Red Hat)

Agenda:

- [Antoine (cannot attend, have a conflict)] Busybox replacement https://github.com/open-telemetry/opentelemetry-operator/pull/3682
  - We'd rather not host this ourselves
  - Maybe otel would be willing to host it as a shared infra tool?
  - Busybox specifically is up for replacement
- [Antoine/Jacob] Does the TA still need to be it's own component? Can the collector just do this now?
  - Collector could theoretically do it, but performance would be bad in large clusters
  - There'd also be additional load on the API Server for each collector doing this
- [all] Issues to discuss at sig (always last)

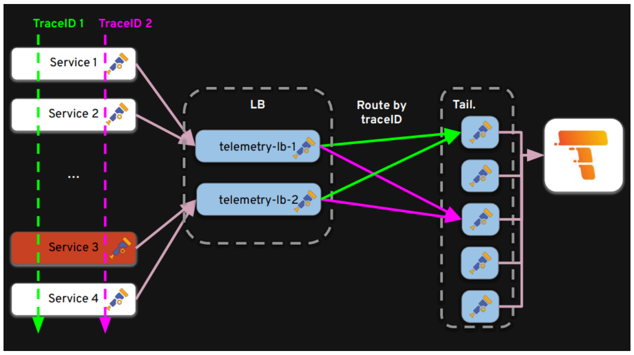
Jan 30, 2025 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Elastic)
- Josh Voravong (Splunk)
- Nikhil Thomas (Red Hat)
- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)

Agenda:

- [all] Issues to discuss at sig (always last)
  - https://github.com/open-telemetry/opentelemetry-operator/issues/3622
    - [Bene] Can it maybe something separate from the target allocator?
    - [Mikołaj] Current approach looks fine, what do others think?
- [Pavol] Stateful storage deployment patterns with Operator
  - Similar to the tailbased sampling setup.
  - E.g. usage for translation from delta to cumulative metrics

  - 

  - Bene and Pavol will discuss a tweak of the tailsampling proposal
- Review feature gates - should we change the stage for any of them?
  - Operator.sidecarcontainers.native
    - Bene wants to make improvements to it first
  - Operator.observability.prometheus
    - Let’s move it up.
  - Operator.collector.default.confg
    - Let’s move it up.
  - Operator.golang.flags
    - Candidate for enabling, but potentially a major change due to GOMEMLIMIT. Would love Jacob’s opinion here.

Jan 16, 2025 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Elastic)
- Josh Voravong (Splunk)
- Gregor Zeitlinger (Grafana)
- Nikhil Thomas (Red Hat)
- Jacob Aronoff (Omlet)
- Benedikt Bongartz (Red Hat)
- Pavol Loffay (Red Hat)
- Luca Casonato (Deno)

Agenda:

- Antoine’s issues relating to making the operator more modular:
  - https://github.com/open-telemetry/opentelemetry-operator/issues/3569
  - https://github.com/open-telemetry/opentelemetry-operator/issues/3568
  - https://github.com/open-telemetry/opentelemetry-operator/issues/3567
  - https://github.com/open-telemetry/opentelemetry-operator/issues/3566
  - https://github.com/open-telemetry/opentelemetry-operator/issues/3565
  - POC: https://github.com/otel-warez/whitegloves-operator
  - Discussion
    - Overall... yes
    - Cert manager
      - We can probably remove the hard dependency here (we already sort of do) but we should document how to do this and why
    - Remove dependencies on CRs
      - Move to use the availability package

        - RBAC
            - Yes, we should *remove* some things, but also this is a packaging problem and we can’t fit the needs/desires of all users
            - We should probably update the helm chart
            - Remove some functionality in there and the helm chart produces the necessary RBAC
    - TODO: Go through and prioritize, label, and begin work where applicable
        - Jacob can take the CR dependencies
- [Gregor] reading config about what to instrument from a config map
    - Where some other service can be the puppet master
    - Notes:
        - Gregor will attempt using a webhook to add in operator labels
        - Also maybe open a PR for his POC
- [Mikołaj] Last call for reviews for adding the TargetAllocator CR
    - Planning to merge right after the next release
- [Michele] Upstreaming LD_PRELOAD and OTEL_RESOURCE_ATTRIBUTES injection by Dash0 (see https://github.com/open-telemetry/opentelemetry-operator/issues/2375)
    - Would love to see a POC for the LD_PRELOAD
    - Michele and Mikolaj will discuss
- [all] Issues to discuss at sig (always last)
- Review feature gates - should we change the stage for any of them?

Dec 19, 2024 9 AM PT/6 PM CET

Attendees (company):

- Benedikt Bongartz (Red Hat)
- Jacob Aronoff (Omlet)
- Pavol Loffay (Red Hat)
- Mikołaj Świątek (Elastic)

Agenda:

- Non-K8s collector discovery in the target allocator #3317
    - Maybe we should also look into splitting the TA into its own repo???
    - Decision?
        - Allow for a static file to be used to pull ips from with a refresh endpoint
- Bene: v1alpha1 Sampling Controller
    - Punt to next year? yes
- Bug-releases - related to the issue above, what is the threshold for releasing a bug fix
    - I think we talked about this already?
- [all] Issues to discuss at sig (always last)
- Review feature gates - should we change the stage for any of them?

Dec 5, 2024 9 AM PT/6 PM CET

Attendees (company):

- Josh Voravong (Cisco/Splunk)
- Mikołaj Świątek (Elastic)
- Jacob Aronoff
- Gregor Zeitlinger (Grafana)
- Pavol Loffay (Red Hat)

Agenda:

- [Gregor, Josh Sueresh] https://github.com/open-telemetry/opentelemetry-operator/issues/3495
    - How to get instance is to collector
        - Join on container, pod, namespace instead
        - Pull from label values (prom scaping)
            - Entities sig will also make this available
            - Could also take labels from OTLP
        - Run ID?
            - Dynamic
        - Store in k8s attribute
        - Use <namespace name><pod name><container name>
            - What to do with multiple clusters? Can happen even if the pod name has a unique part
                - Create separate issue for this
    - Result: keep existing logic - just don't use label to set instance ID
- [Mikołaj] Automatic tool upgrades with renovate (#3450)
- [Mikołaj] Getting rid of kube-rbac-proxy (#3369)
    - Pavol Loffay will take a look
- [Mikołaj] Review feature gates - should we change the stage for any of them?
    - Do this every SIG meeting? Every other one?
    - Open issues for each feature gate as well.
- ~~Bene: v1alpha1 Sampling Controller~~
- Pavol Loffay : Collector upgrade issues (#3515)
    - Current upgrade procedure runs twice - at startup and during reconciliation. The reconciliation upgrade changes only status
    - Unmanaged instances are never upgraded when flipped back to managed
    - Proposal is: run upgrade only as part of the reconciliation
- Pavol Loffay publish CRDs with descriptions and without (what we do right now). OLM uses create and it should use the CRD with descriptions
- Bug-releases - related to the issue above, what is the threshold for releasing a bug fix
- [Mikołaj] Should we split api docs into separate files per CRD?
    - Adding the TargetAllocator CRD makes the file too big for GitHub to render in diffs
    - Yep, let's do it
- [Mikołaj] How do we handle optional resource deletion?
    - See https://github.com/open-telemetry/opentelemetry-operator/pull/3494
    - What about ClusterRoles? Roles in different namespaces that Israel is adding for prometheus receiver?
- [Jacob] Minor announcement
- [all] Issues to discuss at sig (always last)
- [Mikołaj] Review feature gates - should we change the stage for any of them?

Nov 21, 2024 9 AM PT/6 PM CET

Attendees (company):

- Benedikt Bongartz (Red Hat)
- Mikołaj Świątek (Elastic)
- Lukas Hundt Petersen

Agenda:

- Adding a fallback allocation strategy to per-node: issue

    - We're not sure if/how we want to make strategies configurable, so maybe start with a feature flag?
- [all] Issues to discuss at sig (always last)

Nov 7, 2024 , 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)
- Mikołaj Świątek (Elastic)
- Israel Blancas (Red Hat)

Agenda:

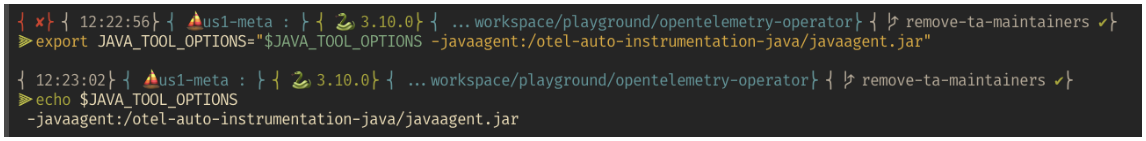
- [all] Issues to discuss at sig (always last)

Oct 24, 2024 , 9 AM PT/6 PM CET

Attendees (company):

- Benedikt Bongartz (Red Hat)
- Josh Voravong (Cisco/Splunk)
- Pavol Loffay (Red Hat)
- Jacob Aronoff (Lightstep)
- Mikołaj Świątek (Elastic)
- David Haja (Itron)
- Yuanfan Peng (Cisco/Splunk)
- Israel Blancas (Red Hat)

Agenda:

- [Josh] Just curious how people decided to manage CRDs in the operator chart? These are not the easiest to use if you use the operator chart as a subchart.
    - https://github.com/open-telemetry/opentelemetry-helm-charts/blob/main/charts/opentelemetry-operator/UPGRADING.md#0560-to-0570
    - https://github.com/open-telemetry/opentelemetry-helm-charts/tree/main/charts/opentelemetry-operator/conf/crds
        - You have to manage them on your own
        - Use a method within your vendor specific chart to install the chart. Check out kubestack.
    - https://github.com/open-telemetry/opentelemetry-helm-charts/blob/b8406fc3f5814a26556e8267182f3fd5d36d7d83/Makefile#L62
- Pavol Loffay https://github.com/open-telemetry/opentelemetry-operator/pull/3373 and https://github.com/open-telemetry/opentelemetry-operator/pull/3379
    - 

```
{ ✗} { 12:22:56} { ⚠us1-meta : } { 🐍 3.10.0} { …workspace/playground/opentelemetry-operator} { ᚴ remove-ta-maintainers ✓}
⮞export JAVA_TOOL_OPTIONS="$JAVA_TOOL_OPTIONS -javaagent:/otel-auto-instrumentation-java/javaagent.jar"

{ 12:23:02} { ⚠us1-meta : } { 🐍 3.10.0} { …workspace/playground/opentelemetry-operator} { ᚴ remove-ta-maintainers ✓}
⮞echo $JAVA_TOOL_OPTIONS
 -javaagent:/otel-auto-instrumentation-java/javaagent.jar
```

- ~~[Bene] Collecting some thoughts:~~
    - ~~https://github.com/open-telemetry/opentelemetry-operator/issues/3356~~
- Yuanfan Peng https://github.com/open-telemetry/opentelemetry-operator/issues/3340
- [all] Issues to discuss at sig (always last)

Oct 10, 2024 , 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep/ServiceNow)

- David Haja (Itron)
- Benedikt Bongartz (Red Hat)
- Josh Voravong (Cisco/Splunk)
- Pavol Loffay (Red Hat)
- Yuanfan Peng (Cisco/Splunk)
- Kristina Pathak (Nvidia)
- Israel Blancas (Red Hat)

Agenda:

- [David] PR review: https://github.com/open-telemetry/opentelemetry-operator/pull/3324
    - [Pavol] is there a bug this is fixing?
        - [David] Operator cannot remove certain fields in the created deployment/daemonset/statefulset
        - Pavol Loffay issues was introduced by https://github.com/open-telemetry/opentelemetry-operator/pull/2941 - we should revert it and just handle annotation and labels separatetelly
        - [Pavol] Just copy the spec over, merge only the annotations and labels together
- [Bene] Native sidecar support
    - https://github.com/open-telemetry/opentelemetry-operator/pull/2801#discussion_r1790997764
    - [Bene] Would be easy to enable for all tests 1.28+, problem is the resulting output in e2e will change
        - This will split our e2e tests :/
        - Add extra folder for only native sidecar
    - [Jacob] Let's do this, but only if we write down the kubernetes version support policy
    - [Josh] Operator has the most flexible support policy I've seen and I think we would need to make it more strict.
        - https://endoflife.date/red-hat-openshift
            - Last supported version is OS 4.12 or K8s 1.25
            - OS Doc: https://access.redhat.com/solutions/4870701 (requires login)
        - https://endoflife.date/amazon-eks
            - Last supported version is K8s 1.23
- [Yuanfan] https://github.com/open-telemetry/opentelemetry-operator/issues/3340
    - In a large enough cluster, pods can get created very quickly at the same time which can OOM the operator
    - Unable to reproduce easily because it requires a huge cluster?
    - Label vs Annotations: https://github.com/open-telemetry/opentelemetry-operator/issues/821
    - [Jacob] The way to do this is via mutating webhook configuration selectors
        - I think this would go well with the instrumentation support select
    - [Jacob] A rate limit wouldn't be a terrible thing if its opt-in, it would cause disruptions but maybe that's acceptable?
    - [Jacob] Let's test out a rate limit, but also scale up/out the deployment to see if that assists
        - I am working on some changes to improve our performance
- [Kristina] https://github.com/open-telemetry/opentelemetry-operator/pull/3343
- [Pavol] https://github.com/open-telemetry/opentelemetry-operator/pull/3338
- [all] Issues to discuss at sig (always last)

Sep 26, 2024 , 9 AM PT/6 PM CET

Attendees (company):

- Josh Voravong (Cisco/Splunk)
- Jacob Aronoff (Lightstep/ServiceNow)
- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)
- David Haja (Itron)
- Israel Blancas (Red Hat)

Agenda:

- Mergo issue https://github.com/open-telemetry/opentelemetry-operator/issues/2947
  - [david] in this case, if the operator removes anyway the manually added stuff, wouldn't it be easier/cleaner if the operator would just replace the spec of the resource, except the propagated labels and annotations?
  - [david] or we want to keep/consider those fields that are not supported in the opentelemetrycollector's specification
  - [Josh] go with uniformity
  - Keep labels and annotations
  - [pavol] set existing = desired (maybe with mergewithoverwritewithemptyvalue)
- Jacob Aronoff Sampling / Instrumentation story
  - Jacob going to maybe take over the instrumentation select story
  - Jacob and Bene will meet and discuss this
- Pavol Loffay Instrumentation CR support TLS, injecting certs
  - Going to secure from SDK to TLS via the instrumentation CR
  - [Josh] Willing to review!
  - Pavol going to review the mTLS PR prior to push
- Pavol Loffay FIPS blacklisted components
  - Some minor go things need to change
  - exCrypto is not handled by some SSL libs
  - Some of these are not FIPS compliant
  - Check for env var, parse collector config for FIPS validation
  - [bene] create a general denylist?
    - Flag will only work with check for FIPS
  - Syntax: Receivers.prometheus, exporters.snowflake
- [jacob] PLz review :) https://github.com/open-telemetry/opentelemetry-operator/pull/3250
- [all] Issues to discuss at sig (always last)

2024/09/12, 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Elastic)
- Pavol Loffay (Red Hat)
- Israel Blancas (Red Hat)
- Benedikt Bongartz (Red Hat)
- Jacob Aronoff (Lightstep/ServiceNow)

Agenda:

- EOL collector v1alpha1
  - Stop support post kubecon NA ‘24?

        - [Pavol] Wants for longer than that, and more expectations around when it will be fully dropped
        - Drop v1alpha1 https://github.com/open-telemetry/opentelemetry-operator/issues/2727
- https://github.com/open-telemetry/opentelemetry-operator/issues/2798
    - We want roughly the following:
        - We always support the K8s versions supported by upstream
        - We make an effort to support as many versions outside the above as possible, and a particular effort to support the major distributions (for example OpenShift)
        - If we drop support for a version, we give at least 1 month worth of notice
- [all] Issues to discuss at sig (always last)
- Good to merge https://github.com/open-telemetry/opentelemetry-operator/pull/3232 ?
- Please review https://github.com/open-telemetry/opentelemetry-operator/pull/3015
-

2024/08/29, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep/ServiceNow)
- Benedikt Bongartz (Red Hat)
- Israel Blancas (Red Hat)
- Gregor Zeitlinger (Grafana)
- Josh Voravong (Splunk)
- Pavol Loffay (Red Hat), I watched recording, I will comment in the PRs

Agenda:

- [Jacob] Resource annotations (link, link)
    - [bene] Maybe we can add an action field similar how its done in the attribute processor?
        - It would help to indicate if fields should be overwritten or not.
        - https://github.com/open-telemetry/opentelemetry-collector-contrib/blob/main/processor/attributesprocessor/README.md
    - [tyler] Should we have a flag on the instrumentation resource that indicates if we want set attributes automatically.
        - Alternatively a featuregate
    - Add this to instrumentation
        - defaults:
        -   useLabelsForResource: true
    - Document precedence in readme
- [Jacob] Where should we be putting runbooks? (link)
    - Let's put it in the docs site so that we can keep it in a static location and if we need to move it in the future the docs site can handle auto-rerouting so as to not break user runbooks
- [Jacob] Quick aside... Do we want to check out https://dosu.dev/ ?
    - Nvm... comments incorrectly all the time and clutters the issues with useless info.
    - That's a no!
- PRs to review (in order of priority)
    - https://github.com/open-telemetry/opentelemetry-operator/pull/3074
    - https://github.com/open-telemetry/opentelemetry-operator/pull/3232
    - https://github.com/open-telemetry/opentelemetry-operator/pull/3206
- [all] Issues to discuss at sig (always last)

2024/08/15, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep/ServiceNow)
- Mikołaj Świątek
- Pavol Loffay (Red Hat)

Agenda:

- RBAC issues
  - [Pavol] Had to disable single namespace mode for conversion webhook
    - It *should* still work but its limited by the conversion webhook definition
- [all] Issues to discuss at sig (always last)

2024/08/01, 9 AM PT/6 PM CET

Attendees (company):

- Israel Blancas Alvarez (Red Hat)
- Jacob Aronoff (Lightstep/ServiceNow)
- Benedikt Bongartz (Red Hat)
- Aunsh Chaudhari (Splunk)

Agenda:

- [israel] working on a bug where multiple instances can't be seen in the openshift dashboard
- [israel] must gather
  - Working on a version that's more generic, not OpenShift specific
- [jacob] InstrumentationRules
  - Link
  - [jacob] I will reach out in slack and propose that he writes an RFC style proposal that we can keep in the repo to get more feedback on
  - Overall the design makes sense, i want the overall goal and edge cases written down and covered as well as rollout plan
- [aunsh] RedHat OpenShift certification for Operator
  - Trying to understand if we're going to have a certification for the otel operator?
  - Bene will take this on and figure it out
- [all] Issues to discuss at sig (always last)
  - 0.0.0.0 for everything
    - We should do this, Jaeger is doing it now too
    - Bene will get to it next week probably
    - Jacob gave a suggestion
  - Jacob will refactor rbac parsing into the new format
    - Israel wil review the issues and add support for the missing components
  - Jacob will look into Role vs ClusterRole story

2024/07/18, 9 AM PT/6 PM CET

Attendees (company):

- Benedikt Bongartz (Red Hat)
- Pavol Loffay (Red Hat)
- Jacob Aronoff (Lightstep/servicenow)
- Mikołaj Świątek
- Israel Blancas Alvarez (Red Hat)
- Curtis Robert (Splunk)
- Tyler Helmuth (Honeycomb)
- Ravi Hari (Intuit)

Agenda:

- Bene: https://github.com/open-telemetry/opentelemetry-operator/issues/3126
    - Does it make sense to hide this behavior behind a feature gate on the operator that we enable once we disable the component.UseLocalHostAsDefaultHost feature gate on the collector?
    - How to deal with components for which we don't have a parser - then users might get inconsistent behavior.
    - The feature gate on the operator is not needed the operator can explicitly set the endpoint which should be respected by collector even -component.UseLocalHostAsDefaultHost is enabled.
- Bene: Please have a look on **Opinionated OpenTelemetry Operator Sampling CR**
    - Should we merge the Processor and Sampler CR?
    - How is the communication between the collector components secured?
        - The Operator generates TLS certificates?
        - Cert-Manager is required and generates certificates?
        - The CR owner can provide references to certificates.
    - Should we provide an option to enable OTLP metric delivery?
        - Maybe also something to generate RED metrics before dropping data points?
        - Is it something we may want to extend afterwards and ignore it for now?
- [all] Issues to discuss at sig (always last)

2024/07/04, 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)

Agenda:

- Bene: Please have a look on **Opinionated OpenTelemetry Operator Sampling CR**

2024/06/20, 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Sumo Logic)
- Pavol Loffay (Red Hat)
- Joshua Voravong (Splunk)
- Benedikt Bongartz (Red Hat)
- Israel Blancas (Red Hat)
- Curtis Robert (Splunk)
- Tyler Helmuth (Honeycomb)
- Jacob Aronoff (Lightstep)

Agenda:

- [Mikołaj] Kube-rbac-proxy upgrades, how do we go about them? The version currently in our manifests has some CVEs filed against it.
    - There to protect the metrics endpoint of the operator from malicious access?
    - Do we need this? Let's open an issue…
    - Mikolaj Swiatek will open the issue and a PR to upgrade, and maybe even to get dependabot to do it automatically.

- Israel Blancas Alvarez : creation of OpenShift meta monitoring dashboard
  - We tried different approaches to avoid adding this programmatically but didn't work properly. We would like to continue with this approach. Any objections?
  - [Mikołaj] this does feel like hacking around OLM and it would be ideal to just put that in the OLM bundle. Is there something we could do there?
  - [Pavol] CM needs to be created in a *different* namespace which makes the owner references not possible.
    - [Mikołaj] this is similar to Prometheus' service installation thing
- [Mikołaj] Am I the only one annoyed by the leader election lease taking 30 seconds to re-acquire when restarting the operator? Can we set it to release immediately?
  - https://github.com/open-telemetry/opentelemetry-operator/blob/dac47749e45fc34d7e83888f3ffedc1b0fb8ea28/main.go#L238
  - Consensus: yes this is probably worth doing :)
  - Mikołaj will open an issue for this
  - https://github.com/open-telemetry/opentelemetry-operator/issues/3058
- [Jacob] golang version bump?
  - Going to follow the collector chart
  - Would be ideal to have some real guidance here
  - [Mikołaj] we should upgrade to 1.22 to unblock us (kubernetes wants us to do this) and then merge the policy... Maybe we should support both?
  - [Pavol] Upgrade, see how it works...
  - [Jacob] Maybe we should just align to whatever Kubernetes is on given we do not treat the operator as a library
  - [Mikołaj] We need more data here to make an effective decision for the policy
- [Josh] Operator chart distribution subchart support. Use case, in OpenShift the operator service account needs a Security Context Constraint associated with the account.
  - https://github.com/signalfx/splunk-otel-collector-chart/blob/1f0617f5ff3c1ac51ca9f9d11885cad990a59fd1/examples/distribution-openshift/rendered_manifests/securityContextConstraints.yaml
  - [Pavol] We have this in the bundle for the OLM installation
  - Tl;dr: if you do this via the openshift subscription you don't need this
  - [Bene] if we made an openshift-specific helm chart distribution then we could just have this logic there
- [Mikołaj] How do we want to maintain auto-instrumentation going forward?
  - We don't have per-language expertise about the instrumentation sdks.
    - How should we deal with issues like this one?
    - [Tyler] we are responsible for the *configuration* of auto-instrumentation correctly inside a pod/container/sidecar/etc. Most of the time we believe we are doing this correctly but on occasion it will break
    - [Pavol] Java and dotnet is simple enough because they are agent based
      - Node and python we are building it ourselves. Are we missing things when we build? We don't get a ton of feedback about this...
      - [Tyler] People do *use* this and it works, it is just fragile and particular.
        - Right now our tests only check if things start, but we don't check the out-end of the pipe
  - We don't have tests for it beyond checking if a simple application container starts.

    - [Josh] We have been adding functional tests using “golden results test logic” per language.
      - https://github.com/signalfx/splunk-otel-collector-chart/blob/main/functional_tests/testdata/expected_kind_values/expected_java_traces.yaml
  - Whose responsibility should it be to upgrade instrumentation libraries?
  - To what extent should the operator test successful instrumentation?
    - See the http semantic convention breaking change as an example of a situation which technically isn’t the operator’s problem, but would come back to bite us if we had ignored it.
  - **Summary**
    - [Pavol] We should generate real smoke tests that actually proves that the instrumented application created real telemetry https://github.com/open-telemetry/opentelemetry-operator/issues/552
      - [Mikołaj] What do we assert on? That telemetry arrived *or* the shape of the data?
    - [Jacob] This should be a milestone that we break down
      - [Mikołaj] Responsibility challenge, who is in charge of these things on upgrades?
- [all] Issues to discuss at sig (always last)

2024/06/06, 9 AM PT/5 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Tyler Helmuth (Honeycomb)
- Ravi Hari (Intuit)
- Kristina Pathak (Lightstep)
- Israel Blancas (Red Hat)
- Mikołaj Świątek (Sumo Logic)

Agenda:

- [Jacob/Tyler] How should we handle CRDs in helm? Long term planning...
  - Right now we have the operator chart and kube-stack chart
  - Decisions to be made
    - I am proposing that the kube-stack embed the CRDs to ***install*** only and then users are responsible for manual upgrades. Is that acceptable in the short term?
      - Do we want a separate isolated chart?
      - [Tyler] The kube-prom-stack chart serves as a good example and we should be close to their patterns. This means doing a nested CRD chart as a subchart
      - [Mikołaj] this is fine for now. 😀
      - [Tyler] OKS will never support v1alpha1 so we can make it super super simple there. Would be a bummer to couple another chart to things...
      - [Mikołaj] A CRDs only chart is conceptually simple (aside from maintenance / updates). There are users that were broken by the templating change and it would be great to give them a nicer solution.
    - Do we eventually want to merge the operator chart into the kube-stack chart? This is what prometheus does?
      - [Mikołaj] Prometheus and otel are not 1:1 so we can’t base *everything* off of them. Not *every* user is going to want to use this chart

  - [Tyler] this is not something we need a decision on today. Starting with a nested CRD means that we limit the support/problem space for this. We could move to a standalone chart pretty simply. Regardless of the decision to move the operator chart, it limits the amount of things we would need to potentially deprecate in the future. It could be challenging to create a new chart and expand our blast radius.
- How long do we want to keep v1alpha1 and conversion webhooks around?
  - It’s deprecated now, but when do we remove it?
- Should the operator go back to installing only after v1alpha1 is gone?
  - [jacob] IMO no... it’s hard to go from installing via helm to not installing via helm. I think the main issue for users today is the conversion webhook
- Ordering of things
  - Embed CRDs in OKS
    - Theoretically a user could use this chart to only install CRDs and the operator...
    - It’s a bit confusing but it is possible!
      - We can write up an example and docs for this!
  - [evaluate decision for isolated CRD chart]
    - Easy to not break CRDs
    - One way door decision because then we need to publish
    - Maybe we should just tell users to manage CRDs entirely separately
  - **DOCS!!!**
    - As part of our docs story we should be documenting what an ideal “production” setup looks like.
  - **NOTE! If helm let you define an order of installation we would be out of this world.**

- [Mikołaj] Target Allocator CRD mechanics (issue)
  - [Jacob] What if it's the TA that contains the entirety of a prometheus configuration and does everything in isolation. And then when we see a collector with the prometheus receiver, we check for a target allocator selector at which point the connection is made? This is sort of related to our distributed configuration discussion, but I’m thinking this would simplify the config song and dance that we have to go through now. This change would also probably make it simpler for non-operator users to use the TA. This would make it more difficult for the current world where we just grab the config from the collector and load it into the TA though which is definitely a major operator benefit. All that to say, I am conflicted.
  - [Jacob] Another option: what if it's the collector that sets a target allocator selector. This would then be added to a configmap. The target allocator is then polling for configmaps and then when it detects a create/update to a configmap it reloads? From there, the TA would select another label (maybe app.kubernetes.io/instance or app.kubernetes.io/name) to determine the collectors that are using this map.
- [Ravi Hari] Load balancing Issue with Target allocator on Otel Collector.
  - How to ensure the pods are distributed evenly across collectors.
- [all] Issues to discuss at sig (always last)

2024/05/23, 9 AM PT/5 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Josh Voravong (Splunk)
- Kristina Pathak (Lightstep)
- Jacob Aronoff (Lightstep)
- Yuri Oliveira (Red Hat)
- Benedikt Bongartz (Red Hat)
- Mikołaj Świątek (Sumo Logic)

Agenda:

- [Josh] Operator Chart Default Instrumentation
  - [operator] Reject invalid/unknown properties using additionalProperties #1065
  - A default instrumentation configuration could be nice, but we need to be careful for deployment
    - Installing the CRD and an instance of a CRD is not fun in Helm
    - Helm Hook solutions are not great, they have pain points
  - AdmissionWebook Issue
    - Fail open, likely what certmanager and prometheus do
  - Options
    - Install CRDs first as part of their own chart/subchart. May not be the best still for downstream consumers.
    - Need to decide how long to support v1 webhook and how we are going to convert
- [Team] How long do we support the old version? Need to decide how long to support v1 webhook and how we are going to convert.
  - We should find convention and what the maintenance burden is for multiple versions
  - Biggest burden is in helm where we need to install in the CRDs via the templating
    - Theoretically getting rid of this would allow us to use the work for a separate CRD chart
    - Positives: Guaranteed upgrades
    - Negatives: You can’t use the chart as a dependency if you want to create your own CRDs in your primary chart
  - **JACOB WILL LOOK INTO CRD CHART**
- [Jacob] Seeking reviews on this (multiple collector versions)
- [all] Issues to discuss at sig
  - Instrumentation support select
    - [Jacob] Maybe this should be called a filter not a selector?
      - Unclear why the cluster admin needs multiple instrumentation resources for the same language? What’s different in each of their java sections?
      - This makes sense for skywalking which is java only and by default will select *all* containers not just the ones with the annotation. This is different from our resource which requires an annotation to know the language.
    - [Pavol] Two concerns
      - On the issue, the use case has no upvotes
        - Unclear if we need this
      - Feels like adding unnecessary complexity
        - With the label selector you’re not saying which language should be injected
    - [Mikołaj] This label is exclusive and forgoes the annotation
      - You could repro this by adding a label to the things you want injected

            - The user wants to have the same annotation on all workloads and then let the operator determine which instrumentation to use
        - [Yuri] You still need the annotation on the pod in order for this to work

2024/05/09, 9 AM PT/5 PM CET

Attendees (company):

- Israel Blancas (Red Hat)
- Jacob Aronoff (Lightstep)
- Kristina Pathak (Lightstep)
- Tristan Sloughter (MyDecisiveAI)
- Tyler Helmuth (Honeycomb)
- Ravi Hari (Intuit)
- Josh Voravong (Splunk)
- Mikołaj Świątek (Sumo Logic)

Agenda:

- [Jacob] release 0.99.0 and 0.100.0
    - Jacob will reach out to vineeth to confirm he can work on the release
    - Link to the 0.99.0 helm release it will be done after this
- [all] Issues to discuss at sig
    - [Israel] Allow setting default configs for all instances
        - [Jacob] Maybe we could generate a new CRD called “default config”
        - [Mikołaj] Why not just call this “Config”? Why wait until we get distributed config and not just build that
            - We’d eventually have to deprecate this in favor of actual distributed config
            - We should probably write this up as “here’s what we’re solving and here’s what we won’t solve”
        - [Israel] Initially don’t create a CR, just have the user specify a config file and merge from that. The configmap *must* be in the same namespace as the operator
            - New CR feels like too much work for this...
            - A flag can be deprecated pretty easily
        - [Mikołaj] If we consider this as a feature for cluster admins, that would be okay
            - We shouldn’t commit right now to complicated config merging
            - The cost is pretty low here
- [Mikołaj] In the Helm Chart, how to handle the caBundle for the conversion webhook if no cert-manager is available?
    -

2024/04/24, 9 AM PT/5 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Matt Hagenbuch (Lightstep)
- Israel Blancas (Red Hat)
- Benedikt Bongartz (Red Hat)
- Jacob Aronoff (Lightstep)
- Tristan Sloughter (MyDecisiveAI)
- Kristina Pathak (Lightstep/ServiceNow)
- Mikołaj Świątek (Sumo Logic)
- Prashant Shahi (SigNoz)

Agenda:

- [Pavol] change default image from core to k8s https://github.com/open-telemetry/opentelemetry-operator/issues/2835
    - Proposal:
        - Change the image in operator 1.0.0
        - Clean operator configuration so no breaking changes are introduced after 1.0 (e.g. use config map instead of flags)
        - Build first class support for collector k8s components in the operator (e.g. simple RBAC management, configuration check)
    - Notes
        - Require users to set the image? It puts more complexity to end users.
        - Use the upgrade procedure for setting/changing the image
        - Wait for the helm chart to rollout the change for requiring the default image
- [Matt] Add support for versioned collector config https://github.com/open-telemetry/opentelemetry-operator/issues/2871
    - Supporting externally defined ConfigMap would be nice but would break some features of the operator (those which rely on reading the config such as container port, targetallocator, liveness probes, likely others)
    - I propose new feature/configuration for the operator to keep N old versions of the collector configmap for rollback purposes
    - Notes:
        - Two separate issues - maybe external config in configmap is ok
        - For versioned configmaps, this should look the same to the user given our current annotation + hash mechanism for updating the deployments
        - Rolling back to previous config is out of scope for operator
        - Make it configurable: set number of versions to keep
        - Chainsaw test
- [Jacob] more typed config here
- [Bene] Opinionated Tailbased Sampling CRD as guinea pig on top of v1beta1 or as part of v1alpha1?
- [Mikołaj] Should the TargetAllocator CRD start at alpha?
- [Ravi] Support for Native Histograms?

2024/04/11, 9 AM PT/5 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Ruben Vargas (Red Hat)
- Mikołaj Świątek (Sumo Logic)
- Benedikt Bongartz (Red Hat)
- Szilard Parrag (Axoflow)
- Kristina Pathak (Lightstep/ServiceNow)
- Tristan Sloughter (MyDecisiveAI)
- Jacob Aronoff (Lightstep/ServiceNow)
- Israel Blancas (Red Hat)
- Tyler Helmuth (Honeycomb)
- Ravi Hari (Intuit)

Agenda:

- [Mikołaj] go and nginx instrumentations are disabled by default, but enabled in our release bundle. We should decide one way or the other. See: https://github.com/open-telemetry/opentelemetry-operator/pull/2777#event-12311525608.
    - [pavol] enable by default everywhere, we still need people to manually do the thing to make it usable
        - Jacob agrees
        - Mikołaj agrees
- [Ruben] Expose metrics on how collector is used https://github.com/open-telemetry/opentelemetry-operator/issues/2829
    - Allow cluster admins to scrape operator metrics to understand more about the collectors it handles
    - Two things:
        - Featuregate alpha
        - Having a list of metrics that users would want here
            - Which components people are running
            - Collector mode
        - Clear list of metrics and why
    - NOTE! Open an issue to discuss metric naming :)
- [Mikołaj] Do we bite the bullet and declare the operator to be Beta? https://github.com/open-telemetry/opentelemetry.io/pull/4148
    - What does beta mean
    - We may technically already be there?
    - How much do users care about our own flags?
    - decision
        - Operator is beta, **we should update our readme** with what that means
        - Maybe update contributing and release instructions too?
- [Tristan] AWS Load Balancers
    - Operator fork to support AWS load balancer ingress
    - Openshift and nginx are supported
    - Would AWS support be accepted?
    - Should be able to just set an ingress class
    - We have an errant variable name
- [Pavol] 0.98.0 release https://github.com/open-telemetry/opentelemetry-operator/pull/2840
    - [jacob] lots of PRs made during the meeting :P
    - Wants to merge this and then the revert back so that we have lots of time
    - https://github.com/open-telemetry/opentelemetry-operator/pull/2836
        - Switch to k8s distro
        - Maybe only v1beta1 will use the new k8s distro?
            - This is maybe *more* complicated
        - [Pavol] Changing for everything would probably be simpler
        - [Tyler] DONT MERGE THE PR DOCKER IMAGE :(TIGRAN IS ON IT !!!
            - Same thing happened as when we moved to docker
        - **Jacob to add subtext and release notes should contain an info block!**
- [Pavol] v1beta1 released with 0.99.0 ?
    - [Mikołaj] We should also produce a What's Changed doc for v1alpha1 -> v1beta1 upgrade
        - We should have a summary of changes
        - How to handle breaking changes

        - Mikołaj will handle this and put it in a markdown file in the repo
        - Docs folder?
        - Pavol to update api docs to use v1beta1?
        - Many selectors are changing
    - [Bene] Seems like ingress consts are not used, we should remove it from v1beta1 as long as we can. 🙂o
        - Pavol agrees, let’s make this nice :)
        - Route config within ingress is :(
    - ***All approvers and maintainers: run the upgrade process in dev clusters to ensure a smooth release!!***
        - **Pavol will inspect webhooks to ensure they work correctly by merging version based webhooks**
- [Ravi] https://github.com/open-telemetry/opentelemetry-operator/issues/2685
    - Seeing an issue with this now
- [Jacob] housecleaning for discuss at sig label
    - Yes
- [Mikołaj] We should audit existing e2e tests and convert them to envtest to make them easier to debug
    - Mikołaj can open an issue

2024/03/28, 9 AM PT/5 PM CET

Attendees (company):

- Mikołaj Świątek (Sumo Logic)
- Benedikt Bongartz (Red Hat)
- Tristan Sloughter (MyDecisiveAI)

Agenda:

- [Mikołaj] Go version policy, let’s make a decision: https://github.com/open-telemetry/opentelemetry-operator/issues/2757
    - Align to the collector repo policy! ✅
    - We should add docs to repo, Bene is on it!
- [Mikołaj] Kubernetes version support policy
    - EOL doesn’t cost anything rn
    - We used to be pinned to kind, but that’s not the case anymore
    - Mikolaj for all eternity commits to bumping kubernetes version so we support HEAD - 6 versions
- [Tyler] When should we include the beta CRDs in the helm chart?
    - Have we released the reconciliation version yet?
        - 0.97.0
    - Starting with 0.97.0 we will need to release the CRDs with the helm chart
    - HIJACK! Mikolaj
        - Right now we cannot upgrade to 0.97.0 to do the following:
            - Add descriptions in to the CRDs
            - We have no API docs
            - https://github.com/open-telemetry/opentelemetry-operator/pull/2563
- [Mikołaj] Windows support, what would be the necessary conditions for us to support it? https://github.com/open-telemetry/opentelemetry-operator/issues/642
    - [Jacob, Mikolaj] I’m not opposed

    - It’s impossible to test this right now?
    - Can’t use windows nodes in kind... grrrrrr
    - Very challenging to test.
    - Precondition for seriously looking at this would be running tests similar to how openshift works
        - Kind: Support Windows "nodes" (not hosts, which are supported) #410
            - https://github.com/kubernetes-sigs/kind/issues/410
    - [Jacob] Seeking help for this issue, would love for someone to assist with this. Testing is challenging and a requirement.
- [Bene] Create milestone for docs (deployment patterns and so on)
    - Do we want to participate in google summer of docs? I will get some time to manage this.
    - https://developers.google.com/season-of-docs
    - [Jacob] Hooray bene, thank u for calling this out
    - [Bene] README is growing endlessly and needs cleanup
    - [Bene] We could create an issue and milestone for documentation
        - Seeking input
        - Planning to apply for the google summer of docs
        - RH has a doc writer who is willing to assist us as well
    - Is otel already applying?
        - Individual groups can submit
- [Jacob] local workflow
    - Will wait :/
- [Tyler] I’m proposing a new k8s distro of the collector, please review the components: https://github.com/open-telemetry/opentelemetry-collector-releases/pull/507
    - **NVM! The helm chart sets it to contrib, but WE set it to core, so NO breaking changes :D <3 <3 <3**
        - **ACCEPTABLE!**
        - **Yes, we should do this with v1beta, we need to delay publishing the CRDs**
    - ~~[Jacob] I think we should change the default image with the v1beta1 release. This makes it a part of a larger breaking change~~
        - ~~We can also easily add in warnings to verify that no one is attempting to run the k8s image with components NOT in the k8s image~~
        - ~~We could also convert the image *back* to contrib if we want~~
            - ~~To what extent do we want to do this?~~
            - ~~We could use a featureflag to handle this for 2 versions?~~
            - ~~[Mikolaj] this is when we convert from v1alpha1 to v1beta1~~
                - ~~Only convert upon the upgrade~~
                - ~~Add a warning for people using AUTOMATIC contrib as part of the conversion~~
        - ~~How do we handle custom registries~~
            - ~~We don’t!~~
            - ~~I don’t think we should solve the general problem for people using the default images~~
        - ~~[Tyler] Helm will also need to do this change~~
            - ~~Most people are using contrib w/o thought~~
            - ~~Some type of helm featuregate and a six month long process because they can’t do major release upgrades. etc.~~

2024/03/22, 9 AM PT/5 PM CET

Attendees (company):

- Pavol Loffay (RedHat)
- Mikołaj Świątek (Sumo Logic)
- Benedikt Bongatz (Red Hat)

Agenda:

- Validate collector configuration
    - Should we implement enhancement validation based on what is in contrib or should we wait for requests?
    - Should the operator switch from being contrib based to the k8s distro?
        - We should define a policy on what components of the opentelemetry collector we want to support. (e.g. everything that is in beta)
- Documentation for the most popular use-cases
    - Target Allocator, how to collect prometheus metrics
        - Can we rename (in documentation) to e.g. prometheus bridge?
            - Maybe even the CRD?
    - Collecting logs
    - Traces from instrumentation
        - Tailsampling
    - Missing docs for RBAC of the components
        - For all components in the kubernetes distro
    - Reference architectures for deployments
- Agree on a long term goal of the operator
    - Operator should do RBAC check and report to the user
        - Not the creation
        - But publish a warning
        - Show the warning in the webhook
    - Provide opinionated CRD that helps with the most common use-cases
- Switch to kubernetes collector distribution
    - It will be a breaking change
    - Do it in v1beta1?

2024/03/14, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (ServiceNow/Lightstep)
- Mikołaj Świątek (Sumo Logic)
- Szilard Parrag (Axoflow)
- Tyler Helmuth (Honeycomb)
- Pavol Loffay (Red Hat), will join 2nd half
- Bryan Aguilar (AWS)
- Benedikt Bongartz (Red Hat)
- Israel Blancas (Red Hat)

Agenda:

- [Jacob] Status section in the otel website (link)
    - Seeking someone to write about this
    - Do we want to declare stability for the operator? Or just for the CRDs?
        - Operator is probably at Beta phase
        - CRDs are self describing
        - Bridge is alpha
        - TA is beta
        - Instrumentation is mixed. Depends on the language
- [Mikołaj] Should the TargetAllocator CRD be published without reconciliation enabled?
    - This is confusing... We could publish it but it will do nothing
    - Maybe this was published by accident in the reconciliation PR
    - Pavol will give thoughts later!
    - It’s not included in the release bundle, so that should be fine.
- [Mikołaj] Go version upgrade policy: https://github.com/open-telemetry/opentelemetry-operator/issues/2757
    - Do we give people a go support policy for people use the operator as a library?
        - Immediate reaction no
    - Collector policy is that they are always N-1
    - Less about features in go, more about dependencies
    - Effectively fast moving deps in prometheus who don’t really care (maybe rightfully so...)
    - Redhat has an operator distro which is also N-1
    - Bene needs to add thoughts here too
- [Mikołaj] Minor question about Makefile targets - do we need to have generate, fmt and vet as a dependency of running tests and building the operator? https://github.com/open-telemetry/opentelemetry-operator/blob/9710eedce46d35ca71a9fa6bd85213859fe6917d/Makefile#L104
    - There should be a distinction between make targets that check things and targets that change things
    - Maybe locally this is good, but in CI we can’t build the manager until we generate things and we can’t generate until we build controller tools
    - Maybe make a separate target for local dev but test shouldn’t need these things
    - This didn’t need to be ‘caught’ in the first place because this binary didn’t require building in the first place
- [Pavol] Distributed collector config (link)
    - [Jacob] My worry is about what fields can be overridden, how do we do the diffing for things like autoscaler, pdb, replicas, etc. when those can have defaults set
    - [Mikolaj] At the very least this is the wrong terminology. This isn’t a set of collectors doing things. Maybe this is too low level as well. What we want is not to merge configurations. Managing configuration merges or masking is the wrong way to think about this
        - This should be composition of things and prevent merging/clashing
        - This is difficult to manage
        - Difficult to tell the end result
        - If we were to build a configuration from pieces, we should try to guarantee that we’re not producing conflicts. We should make things identifiable and sharable.
        - We shouldn’t have situations where we define things in one place and override them in another

  - [Bryan] This seems like a thing that the collector is solving or attempting to solve with remote config sources
    - We want to do something similar to the prometheus operator which allows a set of users in a kube cluster to define what
    - It would be great if users could allow users to define pipelines
  - [Mikolaj]
    - It would be great to solve a situation where diff receivers require different modes
    - If you want a kube event receiver, that has to be a singleton
  - [Jacob] Had to drop. Sorry!
    - Overall i think we need some user stories and problems prior to us making solutions
  - [Mikolaj]
    - *Syncing with Pavol, who just joined*
  - [Pavol]
    - Idea: operator would be doing merging and validation
      - Checking for collisions
  - [Mikolaj] Collector CRs may not be the way to go
    - Partial collector configs
    - How to approach it little-by-little?
      - Full pipelines as a base?
  - [Pavol] Partial configs have their owners
    - CR for component types (receivers, exporters, …)
  - [Mikolaj] Component CRs
    - May be too much for just MVP
    - Single pipeline + it's components' definitions
    - Real use cases will come once they tried it
    - Collector spec new field:
      - Pipelines:
        - PipelineFoo
        - PipelineBar
    - Higher level CRDs
      - Leverages usability
      - Goal: usable OOTB experience without deep understanding of the Collector
  - [Mikolaj] Faulty subconfig can cause crash in the collector
    - Encapsulating correct source / destination definitions, allowlisted processors?
      - Crash free property might be required
- [Mikołaj] Should the TargetAllocator CRD be published without reconciliation enabled?
  - [Pavol] Did not want to publish it into bundle / artifacts
  - Not in the bundle
- [Bene] Follow the go version guideline of the collector
  - https://github.com/open-telemetry/opentelemetry-operator/issues/2757
  - [Mikołaj] Concerned about the prometheus deps
  - [Mikołaj] CRD version mismatch could happen
  - [Pavol] Compatibility matrix for Prometheus version, like for cert-manager?

2024/02/29, 9 AM PT/6 PM CET

Attendees (company):

- Szilard Parrag (Axoflow)
- Peter Wilcsinszky (Axoflow)
- Jacob Aronoff (ServiceNow/Lightstep)
- Pavol Loffay (Red Hat)
- Israel Blancas (Red Hat)
- Hans Kristian Flaatten (Norwegian Government, NAV)
- Mikołaj Świątek (Sumo Logic)
- Shivanshu (Adyen)
- Benedikt Bongartz (Red Hat)
- Dipankar Das
- Praful Khanduri

Agenda:

- [Mikołaj] Changes to the release process
  - Should we have a bot create a release issue automatically with a checklist, like the collector does?
- [Mikołaj] Can we release 0.95.0?
- [Peter] Approaches on handling secrets in the config (e.g. exporter credentials)?
  - What's the right way to handle secrets and credentials for exporters?
  - Bene recommends to mount the secret as environment variable for us via config replacement in the collector
    - https://opentelemetry.io/docs/collector/configuration/#other-information
- [Jacob] discuss stability of changes
  - Link
- [Pavol] v1beta2 collector CR
  - Reconciliation with conversion webhook is getting large
- [Hans] operator config docs + operational best practices (#2682)
  - Digging through the operator config there are some environment variables read directly that are not flags, should those be fixed?
  -
- [Dipankar] jaeger operator question
  - So is there some docs /design / sequence diagram for how jaeger operator v1 looks like. Actually I was trying to understand what all changes are required for jaeger v2. Aka is there a the overall docs for this
  - Also what I'm seeing is *3 controllers [namespace, elasticsearch, jaeger]*. How it is connected and also seen webhooks which injects jaeger as a sidecar.
  - Need for a sidecar deployment option for jaeger v2?
    - As most of the things are now being done just by replacing the config.yml
    - I can miss somethings as I am exploring both simultaneously
  - What all things I should keep in mind for jaeger v2 operator
  - *Context*: I am beginner to kubernetes operators
  - Also where to ask future questions on jaeger v2 operator specifically?

2024/02/15, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (ServiceNow/Lightstep)
- Pavol Loffay (Red Hat)
- Charles-Edouard Brétéché (Nirmata, kyverno/kyverno-json/chainsaw)
- Kristof Gyuracz (Axoflow)

- Szilard Parrag (Axoflow)
- Tyler Helmuth (Honeycomb)
- Peter Wilcsinszky (Axoflow)
- Kristina Pathak (ServiceNow/Lightstep)
- Mikołaj Świątek (Sumo Logic)

Agenda:

- [Pavol Loffay] RBAC handling for TA and processors
  - https://github.com/open-telemetry/opentelemetry-operator/issues/424 RBAC for TA
  - https://github.com/open-telemetry/opentelemetry-operator/issues/2588 RBAC for processors (k8sattribute…)
  - We should improve docs for these features.
  - Document which RBAC is required for different operator functionality
    - TA
    - K8sattribute processor
- [Charles-Edouard Brétéché] Chainsaw
  - https://github.com/open-telemetry/opentelemetry-operator/pull/2607
  - Overall positive things :)
- [Kristof Gyuracz]: Telemetry-controller demo (follow up from last meeting)
- [Jacob Aronoff]: Discuss signature cleanup
- [Pavo lLoffay]: v1alpha2 progress.
- [Pavol Loffay]: When to release 0.94.0?

2024/02/01, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (ServiceNow/Lightstep)
- Peter Wilcsinszky (Axoflow, maintainer of Logging Operator)
- Benedikt Bongartz (Red Hat)
- Israel Blancas (Red Hat)
- Pavol Loffay (Red Hat)
- Kristina Pathak (ServiceNow/Lightstep)
- Tyler Helmuth (Honeycomb)'
- Mikołaj Świątek (Sumo Logic)

Agenda:

- Peter Wilcsinszky: Multi-tenant controller intro (~20m)
- Bene: https://github.com/open-telemetry/opentelemetry-operator/pull/2532

2024/01/18, 9 AM PT/6 PM CET

Attendees (company):

- Israel Blancas (Red Hat)
- Jacob Aronoff (ServiceNow/Lightstep)
- Ben Evans (Red Hat)
- Benedikt Bongartz (Red Hat)
- Tyler Helmuth (Honeycomb)
- Kristina Pathak (ServiceNow/Lightstep)
- Elliott Baron (Red Hat)
- Josh Voravong (Splunk)

- Pavol Loffay (Red Hat)
- Bryan Aguilar (AWS)

Agenda:

- [jpkroehling] If you are a code owner, an approver (core, contrib, operator) or maintainer (core, contrib, operator) and will be at KubeCon (or Observability Day), ping me!
- Israel Blancas After the changes I made to automatically create the RBAC resources for some processors, this happened (thanks Mikołaj for fixing it). Now, when you enable the "--create-rbac-permissions" flag, the operator has not permissions anymore over ClusterRole and ClusterRoleBinding:
    - Should we add them back or, at least, document that users need to add them manually?
    - Can we make it so that we check that the operator has permissions before we add this?
        - Yes, Israel to open issue
        - We should make users aware this is happening, and users who don't want this can disable this in a very easy way as well.
- Benjamin Evans (Ben) Question regarding use of "auxiliary" agent instead of Java auto-instrumentation.
    - Redhat wildfly and quarkus which don't require the java auto-inst instrumentation
    - What does exist is a metadata agent
    - One is able to spin up a custom image for instrumentation
    - How flexible is the configuration for that agent?
- Bene:
    - Please add new fields not only to v1alphav1
    - https://github.com/open-telemetry/opentelemetry-operator/pull/2541
    - Can I get someeyes on https://github.com/open-telemetry/opentelemetry-operator/pull/2532
    - Wip (should be polished in a few hours)
- Tyler: discuss how to handle the dotnet auto-instrumentation upgrade: https://github.com/open-telemetry/opentelemetry-operator/pull/2538
    - There are breaking changes in dotnet and they're not supporting backwards compatibility
        - We should raise an issue that their changelog needs to be more descriptive about these breaking changes
    - Release notes are very unclear this is a breaking change
    - Add a warning for the switch in the webhook?
        - Maybe we can give people docs in a warning message
    - Should we feature gate this?
    - Can the collector do this?
        - Schemaprocessor not ready yet
        - Can't do translation for reaggregation for metrics
        - Some attributes split in to two, not handled currently
    - Mikolaj:
        - This is not ONLY for dotnet, every instrumentation will have this issue
        - Lets document a warning for this upgrade
        - If we know where the break happens, we could add a shared instrumentation flag for using the new schema
        - Add an opt-in per language setting
            - We support them for N releases

      - Committees have not implemented a mandate around the semconv end dates for when they have to make this change
    - This is a big problem for end-users
      - Has this been brought up in the GC? Agents/Instrumentation libs should make it clear what they support.
    - We should do a major version bump for the published instrumentation libraries to make it clear that each of these are breaking.
    - Let's hold off on bumping dotnet
      - Let's also wait on bumping the default for java
      - DO NOT INCREASE THE VERSIONS.TXT VERSIONS
    - We'll have to do this per language, and we should add something to release notes
    - Tyler will create an issue for us :)
      - Jacob will join for the next maintainers meeting
- jacob.aronoff@lightstep.com go over target allocator architecture
  - TL;DR the way we do servicemonitor config gen is cumbersome and makes it difficult to do things like secret and configmap mounting

2024/01/04, 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Sumo Logic)
- Israel Blancas (Red Hat)
- Rajib Mitra (Cisco)
- Doug Odegaard (ServiceNow/Lightstep)
- Jacob Aronoff (ServiceNow/Lightstep)
- Bryan Aguilar (AWS)

Agenda:

- Release 0.91
  - Anything else we want to get in?
- Frustration around security context for the TA
  - Right now pod security context is in AS security context
  - What do we do now?
    - Check if this can be done as a non-breaking change
    - If it is breaking, we can do in v2

2023/12/21, 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Mikołaj Świątek (Sumo Logic)
- Benedikt Bongartz (Red Hat)
- Israel Blancas (Red Hat)

Agenda:

- Pavol Loffay : https://github.com/open-telemetry/opentelemetry-operator/pull/2444
  - kustomize removes empty objects e.g. batch:.
  - Kubectl edit removes empty objects as well.
  - Workaround: batch: {}
- Israel Blancas Alvarez : feature flags vs CLI arguments

    - Context: https://github.com/open-telemetry/opentelemetry-operator/pull/2394#discussion_r1431298130
    - Should we revisit the feature flags we currently have and move some of them to be CLI arguments?
    - We'd like these to be command line flags to begin with, and eventually move them to a configuration file
    - Migrating existing instrumentation flags requires some input from owners
    - For new functionality which we want users to be able to control, we should use command line flags, and not feature flags
- Mikolaj Swiatek Where should we check if manifests need to be generated? Currently we don't have a consistent policy.
    - Feature flag checks in the builder?
    - CR field checks in manifest functions, return nil if not generated?
    - Yes, do the above and document in the builder file
- Mikolaj Swiatek TargetAllocator CRD final decisions on the design

2023/12/07, 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)
- Mikołaj Świątek (Sumo Logic)
- Jacob Aronoff (Lightstep)
- Israel Blancas (Red Hat)
- Tyler Helmuth (Honeycomb)

Agenda:

- Pavol Loffay : scoping v1alpha2 https://github.com/open-telemetry/opentelemetry-operator/milestone/4
    - We do not want to duplicate the reconcile logic for v1alphav2
    - Should the instrumentation CR stay on the previous version?
        - Pro:
            - Consistency
        - Con:
            - We avoid the chance of doing breaking changes
    - Triage milestone async
    - The milestone should contain only changes that require breaking changes in the collector v2 type
- Israel Blancas : v1alpha2 reconciliation. How do we want to do it?
    - My understanding was that what was proposed by the SIG was:
        - Duplicate the reconciliation logic from v1alpha1
        - Protect the new reconciliation logic behind a feature flag
    - Some of us think duplicate the full logic can be a maintainability problem. Ideas?
        - Israel Blancas : I propose to enable the conversion webhook using the same feature flag than from the reconciliation
- Mikolaj Swiatek New TargetAllocator CRD: https://github.com/open-telemetry/opentelemetry-operator/issues/2422
    - How should the TargetAllocator be related to the OpenTelemetryCollector?

        - They are independent and have references to each other
        - Can one of the CRs update the other?
            - Maybe via an annotation?
            - Kong operator works like this
            -
    - Do we continue to support the subresource?
    - Rollout plan
- Mikolaj Swiatek This Target Allocator change cleans up the logic and allows us to support more prometheus-operator features. It also requires a new RBAC permission and would surely completely break a lot of users. What can we do about this?
    - Just do the breaking change and put a huge warning in the changelog?
    - Maintain two code paths at runtime while warning users about the permission?
    - Add a flag to control this?
    - Could we do this check in the webhook? Check the RBAC against the TA image?

2023/11/23, 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)
- Mikołaj Świątek (Sumo Logic)

Agenda:

- Thanksgiving?
- Pavol Loffay first time contributor experience
    - https://github.com/open-telemetry/opentelemetry-operator/pull/1794
    - get rid of DATE from the Makefile
    - Remove $USER
    - run e2e tests easily, no changes
- [Mikołaj] Using a single Go binary for all binaries
    - easier to work in a repo that has a single go.mod
    - move the operator main file to ./cmd
    - try and revert in case of issues
- [Mikołaj] Should we wait for v2 CRD to do https://github.com/open-telemetry/opentelemetry-operator/pull/2369?
    - do the change in v1alpha2/v2

2023/10/26, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Israel Blancas (Red Hat)
- Mikołaj Świątek (Sumo Logic)
- Benedikt Bongartz (Red Hat)
- Bryan Aguilar (AWS)
- Doug Odegaard (ServiceNow/Lightstep)
- Tyler Helmuth (Honeycomb)

Agenda:

- ~~[Morgan] Kubecon: we have a meeting / discussion space called the OpenTelemetry Observatory that anyone in the community can sign up for: https://docs.google.com/forms/d/1MaUy6i5hASesz7QxAvrdBMQICdJMxGOXKvsi6bWmNPQ/viewform?edit_requested=true~~
- ~~For example, someone could host a feedback session or walkthrough of our Android or iOS status~~
- [Jacob] Is this type of PR something we want to accept? (link)
  - This generates some data, sends to collector -> kafka -> collector -> prometheus
  - Then data is read from prometheus
  - Where does this belong?
    - Maybe the collector?
    - Not sure this benefits the operator project
  - **Yes, operator doesn't want this. Should go to contrib or openshift**
    - Collector does have kube e2e tests so it has some precedence there!
    - This definitely stresses the test framework.
    - If we're going to write more things like this, we should probably have our own test framework that can be a bit more advanced, it's not very maintainable.
      - Maybe have an issue for this to discuss for later Mikolaj Swiatek
- [Jacob] How can we embed pod specs more effectively?
  - I was thinking we could do a shared pod spec config with the paired down fields we want configurable and just inline them in each objects config
    - Use some inlining of a struct?
    - We could provide some type of "override fields" that can override specific fields in the generated CRD spec
    - [mikolaj] not comfortable with walking around the typing, feels unsafe and fraught with concern
    - [bene] this feels like something to be used with experiments
    - [mikolaj] i feel like we should just bite the bullet and embed everything we can to give to users, its far from ideal, but there's not a great way to avoid it
    - jacob.aronoff@lightstep.com will take this as an issue to embed the spec
  - I also was thinking it may be time to move the TA spec to its own CRD
    - Create a new CRD specifically for running the target allocator, none of the logic changes, just cleans up the ability for a user to configure it
    - We could do something smart with references as well
    - [mikolaj] lets avoid the problem that the prometheus CR has where its too large
    - Redirect any request to add more fields in to the CR itself
    - jacob.aronoff@lightstep.com will make an issue and draw up some designs!
- Israel Blancas Alvarez : please review the bumping of the CRD version
  - Seeking reviews
  - Bump CRD version to v1alpha2
  - This doesnt remove or change anything, just copies it directly
  - Also has a webhook that allows for backwards compatibility
  - [bene] people will be able to immediately apply v1alpha2, we should probably hide this behind a featuregate so that people can't create it directly just yet
    - This would let us remove the fields in subsequent PRs with deprecating
  - [bryan] maybe we do this in a feature branch and then merge it one go?
    - [bene] there may be _A LOT_ of conflicts given all the past discussion

        - [bene] first, create the v2 structs, dont let anyone create them
            - Then add the webhook and enable as a feature to migrate it
            - This would let us also split out the TA and embed changes
        - [mikolaj] example: someone creates a v1alpha1, autoconverts to v1alpha2
            - Any change to the v1alpha2 will break existing users
            - To merge as is, you would maybe put the webhook behind the featureflag
        - [jacob] heres how you can schedule it
            - Just create the CRD types, no manifests or anything
            - Have the operator reconcile it behind a feature flag
            - We stack all the above changes as part of that feature flag
                - Embedding spec inline
                - Ta spec with its own CRD
                - Etc.
            - Some period of testing?
                - Put reconciliation flag in beta
                - Webhook behind a featureflag
                - Some period of testing
                - Put webhook flag in beta
                - Some stable state
                - Stable feature flags for all
            - Remove old types?
        - [mikolaj] most of the current PR is boilerplate
            - Lots of generated type defs
            - Concerned about how we do this to prevent a long dual state
        - Israel Blancas Alvarez will write up a new plan in the issue :)
    - [Mikołaj] I'd like to change how scrape targets are assigned to collectors by the target allocator, looking for feedback about unintended consequences (link)
        - Targets shouldn't move between collectors unless 100% necessary
        - Some of the labels we work with are mutable
        - What if we did allocation _only_ on the address, not the hash?
    - [Ilya M] Instrumentation - Musl libc based application containers. Looking for a feedback on Issue and PR draft link (+ slack link)
        - We do this for .NET, we have to be careful to not go too crazy here
        - It does create an issue where we would need everyone to specify an extra annotation
        - [mikolaj] we would have one image with multiple architectures
        - [tyler] we would need to add an annotation for MUSL
        - [mikolaj] there's not a way to auto-detect this to make it easier for users
            - Maybe we could look at the image manifest? But that's not very simple
            - As more languages have this problem, it only is exacerbated
            - The only solution found was entrypoint injection

2023/10/12, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Benedikt Bongartz (Red Hat)
- Bryan Aguilar (AWS)
- Tyler Helmuth (Honeycomb)
- Pavol Loffay (Red Hat)

- Mikołaj Świątek (Sumo Logic)

Agenda:

- [Jacob] Webhook refactor PR [link]
  - Going to move the webhooks back to v1alpha
- Pavol Loffay when to cut 0.87.0
  - Fix the flaky test first
  - Fix the CVE before the release
    - Wait for the collector patch release
    - Golang CVE-2023-44487
      - https://github.com/open-telemetry/opentelemetry-operator/security/dependabot
      - https://github.com/golang/go/issues/63417
      - https://github.com/grpc/grpc-go/releases/tag/v1.58.3
      - https://github.com/grpc/grpc-go/commit/5efd7bd73e11fea58d1c7f1c110902e78a286299
      - https://github.com/grpc/grpc-go/commit/c40c9ba315aaaa58eb4dc74ba9d8bbc5058f1dfa

2023/09/28, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)
- Tyler Helmuth (Honeycomb)
- Bryan Aguilar (AWS)
- Mikołaj Świątek (Sumo Logic)

Agenda:

- [Bene] Downward compatibility of the collector configuration
  - https://github.com/open-telemetry/opentelemetry-operator/issues/1707
  - Bump to new API version (e.g. v1beta2)
  - Prometheus versioning https://github.com/prometheus-operator/prometheus-operator#project-status
  - https://kubernetes.io/docs/tasks/extend-kubernetes/custom-resources/custom-resource-definition-versioning/#webhook-conversion
  - Use config struct from collector core for some validation
- [Bene] Init container to validate collector config - Did we not want to validate in the webhook?
  - https://github.com/open-telemetry/opentelemetry-operator/pull/2166

2023/09/14, 9 AM PT/6 PM CET

Attendees (company):

- Israel Blancas (Red Hat)
- Jacob Aronoff (Lightstep)
- Bryan Aguilar (AWS)
- Benedikt Bongartz (Red Hat)
- Pavol Loffay (Red Hat)
- Mikołaj Świątek (Sumo Logic)

Agenda:

- [Israel] Host the example applications for the E2e instrumentation tests in the repository
    - Discussion: https://github.com/open-telemetry/opentelemetry-operator/pull/2082
    - The proposed approach in the PR is to build and publish the images when they are modified.
    - Should the images be built everytime we run the CI or build and published when modified?
    - [Mikołaj] For testing instrumentation injection alone, the image may as well be empty. Are we planning to develop proper E2E tests checking if the instrumentation actually sends data?
        - [Israel] There was a comment about that this week
    - Hosted apps will still receive dependabot and CVEs
        - Enable dependabot schedule for every 3 months
        - Group dependabot PRs
        - Mark images as experimental/dev
        - Make it low maintenance for GHA (e.g. single GHA)
- [Jacob] go over minor re-designs in Bridge design
- [Bryan] Failed port mapping config validations can reject collector configurations
    - https://github.com/open-telemetry/opentelemetry-operator/issues/2105
    - https://github.com/open-telemetry/opentelemetry-operator/blob/7d503f6d89a6cc9823988bb7d33439d216af99ce/internal/manifests/collector/adapters/config_to_ports.go#L133
    - Enhancement - Run a job as validating webhook using collector validate command. Allows the operator to fail fast for bad configs.
    - Decision:
        - We should make a best effort for parsing and validation
        - We can run a job for validation? Or maybe we should do an init container?
        - We should also define our framework for requirements/rejections
            - Do the best effort
            - Don't fail the deployment unless for a _known_ configuration
            - We should be able to _explicitly_ fail for operator-related functionality
- [Mikołaj] Wanted some more eyes on this: https://github.com/open-telemetry/opentelemetry-operator/pull/1794
- [Jacob] Call for reviews on https://github.com/open-telemetry/opentelemetry-operator/pull/1901

2023/08/31, 9 AM PT/6 PM CET

Attendees (company):

- Israel Blancas (Red Hat)
- Jacob Aronoff (Lightstep)
- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)
- Bryan Aguilar (AWS)

Agenda:

- [Tyler] https://github.com/open-telemetry/opentelemetry-operator/pull/1901
    - Will check in on this in a week or two as people come back from vacation
- [Israel] Move the "business applications" used for the E2E tests to the repository
    - Current problem: one the images we use cannot be run as non-root https://github.com/cisco-open/appdynamics-k8s-webhook-instrumentor/pull/6

    - Some of the images are big like the to test the Java instrumentation so, maybe reducing the size can speed up the CI
    - Notes
        - Israel will make a folder per instrumentation e2e tests
        - Jacob will make an issue to investigate conditional test runs
- [jacob] reconcile changes
    - We should move things from autodetect to featuregate? (hpa, routes, etc.)
    - Create ticket to migrate openshift route changes via tasks to use manifest building

2023/08/17, 9 AM PT/6 PM CET

Attendees (company):

- Mikołaj Świątek (Sumo Logic)
- Jacob Aronoff (Lightstep)
- Israel Blancas (Red Hat)
- Tyler Helmuth (Honeycomb)
- Kristina Pathak (Lightstep)

Agenda:

- [Mikołaj] Configuration hot reload for Target Allocator. Do we want it?
    - Yes, remove via a feature flag. Deprecate now, remove in 2 versions
- [Mikołaj] Upgrading k8s libraries to 0.28.0 causes CRD changes: https://github.com/open-telemetry/opentelemetry-operator/pull/2031
    - Approved and merged
- [Mikołaj] Do we have a policy for CRD api version changes?
    - ... I don't think so
    - Maybe wait on this
    - When do we increase the minor alpha version?
    - When does the CRD become v1?
    - Helm makes doing these changes difficult
    - How would this work?
        - Not sure.............................
        - Helm can install but cannot upgrade/delete them
        - You can bump a version and then provide a migration
- [Jacob] distributed collector config https://github.com/open-telemetry/opentelemetry-operator/issues/1906
    - We have an open issue for config NOT as a string! Would be good to do this first
    - Tyler is going to hand off this issue, doesn't have time currently
        - SEEKING CONTRIBUTOR!
- [Israel] Add metrics to the OpenTelemetry Collector status
    - Related, but not exactly this
    - We could make an extension for getting status from the collector once this is done at which point we can implement the logic similar to the health check extension where we look if its present and if so get the status from the collector

2023/08/03, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Tyler Helmuth (Honeycomb)

- Benedikt Bongartz (Red Hat)
- Pavol Loffay (Red Hat)
- Israel Blancas (Red Hat)
- Kit Sutliff
- Kristina Pathak (Lightstep)
- Bryan Aguilar (AWS)

Agenda:

- [Tyler] Interesting issue with initContainers failing: https://github.com/open-telemetry/opentelemetry-operator/issues/1955
- [pavol]: Ingress/Route remove path e.g. /otel-grpc
    - Going to change the logic to let people do either
    - Need to make sure we're not going to break existing users
- [pavol]: Prow.ci https://github.com/open-telemetry/community/issues/1557 and revert https://github.com/open-telemetry/community/issues/1591
    - Slack https://cloud-native.slack.com/archives/C056QUS3RPV/p1689171900286689
        - https://github.com/open-telemetry/opentelemetry-operator/pull/1889
        - https://github.com/open-telemetry/opentelemetry-operator/pull/1910
    - Add prow.ci back but disable openshift-bot comments and make the prow.ci jobs optional. Document prow.ci in the contributing.

2023/07/20, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Israel Blancas (Red Hat)
- Mikołaj Świątek (Sumo Logic)
- Benedikt Bongartz (Red Hat)
- Kristina Pathak (Lightstep)
- Tyler Helmuth (Honeycomb)
- Bryan Aguilar (AWS)
- Kit Sutliff

Agenda:

- https://github.com/open-telemetry/opentelemetry-operator/issues/1849
    - Can we be a bit lazy and just make new annotations?
        - This would then be specified via an annotation for runtime
        - We'll need to probably cross-compile our image
        - Who wants to do this?
            - Tyler can maybe get to this in a month
            - Mikołaj too
            - Add a help wanted, describe the solution

2023/07/06, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Mikołaj Świątek (Sumo Logic)
- Benedikt Bongartz (Red Hat)
- Bryan Aguilar (AWS)

- Tyler Helmuth (Honeycomb)
- Pavol Loffay (Red Hat)

Agenda:

- [Mikołaj] Do we want to consider distributed otel configuration as within the scope of the operator?
    - Similar to Service/Pod monitors from Prometheus
    - Use case: https://github.com/kube-logging/logging-operator
    - User wants to distribute OTEL collector config to service teams
    - Mikolaj Swiatek will create an issue to house the discussion about this.
- [Bene] What's your thoughts on splitting the otel collector and allowing adding some kind of modules via e.g. configmap?
    - First the support needs to be added in the collector.
    - https://github.com/open-telemetry/opentelemetry-collector/issues/7961
    - https://github.com/open-telemetry/opentelemetry-operator/issues/1817
        - (Focus on comment - not original proposal)
    - Vendors get requests for running additional components that are not included in their distributions.

2023/06/22, 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)
- Jacob Aronoff (Lightstep)
- Mikołaj Świątek (Sumo Logic)
- Kristina Pathak (Lightstep)
- Kit Sutliff
- Bryan Aguilar (AWS)

Agenda:

- [Tyler] May need additional dotnet images: https://github.com/open-telemetry/opentelemetry-operator/issues/1849
    - Glibc vs musl
    - Dotnet instrumentation image per architecture
    - Can the operator figure out what architecture image to use?
    - Ask dotnet community to make the dotnet instrumentation detect glibc/musl automatically.
    - We can bundle all dependencies into the same auto-instrumentation image and introduce config to select new dependencies
- [Tyler] Something is wrong with our CI (specifically starting kind clusters)
    - Figure out why clusters actually don't start
    - Create a ticket with links to failed jobs - Tyler
        - https://github.com/open-telemetry/opentelemetry-operator/issues/1863
- [Pavol] Update PRs before merging
    - Pavol - create a ticket to disable the requirement to have updated the PR to head/main
    - Issue https://github.com/open-telemetry/community/issues/1558
- [Pavol] Openshift prow.ci - https://github.com/open-telemetry/community/issues/1557
- [Mikołaj] Updating the OLM bundle when working locally (https://github.com/open-telemetry/opentelemetry-operator/pull/1794)

    - Publish bundle https://github.com/open-telemetry/opentelemetry-operator/issues/1823

2023/06/08, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Mikołaj Świątek (Sumo Logic)
- Israel Blancas (Red Hat)
- Tyler Helmuth (Honeycomb)
- Kit Sutliff (OpenTelemetry - Contributor)
- Bryan Aguilar (AWS)
- Terra Mauthe (Grafana)
- Benedikt Bongartz (Red Hat)
- Pavol Loffay (Red Hat)

Agenda:

- [Jacob] Retraction of bad versions (issue)
- [Mikołaj] Injecting env variables into Prometheus config (issue)
- [Mikołaj] Requesting review for https://github.com/open-telemetry/opentelemetry-operator/pull/1794
- [Bene] distribution image builder service https://github.com/open-telemetry/opentelemetry-operator/issues/1817
    - [Tyler]
        - Overall likes this idea
        - Not a lot of people spend the time to use the builder
        - Maybe we can create a kubernetes distribution in the OC-releases?
            - Pavol likes this :)
        - If the operator can help with this, great :)
        - It would be great to add a bunch of labels/metadata to the docker image to read them and we could do something with that
        - Tyler will propose a Kubernetes distribution in the release repository
            - Nothing stopping US from releasing our own distro :) We're an agreeable bunch are we not :)
    - [Mikołaj]
        - This is going to be ... hard
        - Overall likes the idea
        - Is there a simpler way to start?
        - Kube distro yes, maybe also a node agent?
        - Contrib is a pretty big mem hog OOTB
        - For Bryan
            - We already parse the target allocator and prometheus configuration
            - For the validation, we're running against the problem that we need to run an actual image here
            - What if we could do something with the docker image metadata
            - Maybe there were a standard way of including this in the docker image metadata and read that statically to know if its present
            - (Jacob thought: maybe this would be good with improving the collector's BOM!)

- [Jacob]
    - Is there any prior art in Kubernetes for this?
        - Are there other projects that also do this?
        - I know that ArgoCD does something with docker creds that may be good to look at?
            - Drone CI?
            - OSBuild? Builds Fedora images
    - Could we make a github action that helps with this?
        - We could also use this ourselves for the Kubernetes node and gateway collectors?
    - How would creds work for pulling from a private git repo?
        - Store creds in secret, use them over http
        - Operator has access to those secrets and can fill them in
        - Pretty good failure states for debugging (just bubble up the issues)
- [Bryan]
    - Really cool new and novel idea!
        - Definitely keep encouraging this :)
    - This is a big ask – all the implementation would be pretty large
    - Is it a good idea to mix these two things together?
        - BOM is becoming a _super_ hot topic
        - Are we able to deliver this at build time
    - How would this work for multi-cluster setups?
        - What do you do if you need to go through some amount of validation / verification
        - If we continue going this route, we should clear up the use cases
    - Is it the responsibility of the operator to build these images?
    - Is it more an issue of the collector that it's difficult to generate these
        - Do we need a SIG to swarm on collector releases?
    - Managers would probably want really clear validation steps on this
    - What if we had a group of distributions
        - [Tyler] There are no hard component requirements in the operator right now
        - [Tyler/Bryan] Prometheus is probably a big requirement right now...
    - It could be a great thing to make composable github actions for the OCB and then donate that to the releaser
        - The action would then contain the metadata
- [Pavol]
    - Likes the kubernetes distro
    - Operator should be able to check the capabilities of the distribution
        - (Jacob has thoughts here!)
        - https://github.com/open-telemetry/opentelemetry-collector/pull/7835
        - Could the validation webhook run the image and validate and give back an error to a kubectl apply?
    - Operator cannot verify that certain capabilities are in the image
    - Operator could be built with default image, and could potentially run the container on its own file system
- [Bene]

        - With the metadata that comes with the container image
        - Could we attach the components directly via OCB?
            - While building it could I add the metadata right there?
        - Maybe we could add metadata to the go-releaser action

2023/05/25, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Benedikt Bongartz (Red Hat)
- Josh MacDonald (Lightstep)
- Israel Blancas (Red Hat)
- Pavol Loffay (Red Hat)
- Mikołaj Świątek (Sumo Logic)
- Bryan Aguilar (AWS)

Agenda:

- ~~[Tyler] https://github.com/open-telemetry/opentelemetry-operator/issues/1765~~
    - It was istio's fault
- [Israel] Support collecting metrics from instances: https://github.com/open-telemetry/opentelemetry-operator/issues/1768
    - Some discussion about why prometheus
    - What do we do when we eventually go OTLP?
    - Maybe we can provide some convenience for configuring the collector's own telemetry?
    - We absolutely need to set the service telemetry at _some_ point
    - Follow-up task:
        - What alerts do people set up for the collector?
        - https://github.com/open-telemetry/opentelemetry-collector/blob/main/docs/monitoring.md
        - https://opentelemetry.io/docs/collector/scaling/
            - Doesn't cover specific pipeline components monitoring information
    - Could we add in automatic smoke testing in the operator to confirm that the collector is working as expected
        - jacob.aronoff@servicenow.com will make an issue for this
        - Tempo operator runs a smoke test like this as well to confirm that metrics arrived
        - Runs test and updates status on CRD
    - Maybe operator could also create configmap that defines otel-environment variables
        - Pavol will make an issue for this - https://github.com/open-telemetry/opentelemetry-operator/issues/1770
        - Maybe this is an added instrumentation feature
        - Add a new field in to Instrumentation for collector name
- [Mikołaj] Is it valid for target allocator to do nothing? As in, no Prometheus CR and no scrape configs from receiver. See: https://github.com/open-telemetry/opentelemetry-operator/pull/1767
    - Summary: prom receiver / target allocator, FF will auto-add the TA config. For a collector that is configured with the prometheus receiver, it can have scrape configs _and_ target allocator section. Collector needs _at least one_, the TA can have neither and that is maybe valid?

        - In a situation where the TA has no scrape configs and prom CR is disabled, should it error?
        - Maybe run and do nothing?
    - [bene] if it fails, does it get restarted? We could also reject at CRD admission?
        - Maybe a better status field to inform someone that the TA and collector are “idle”
        - Maybe also drop it...
    - [Mikołaj] using the prometheus parser is a PAIN! If we just parse the things as dumb maps it becomes easier to figure it out.
- [Jacob] https://github.com/open-telemetry/opentelemetry-operator/pull/1653

2023/05/11, 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)
- Tyler Helmuth (Honeycomb)
- Jacob Aronoff (Lightstep)
- Josh Voravong (Splunk)
- Bryan Aguilar (AWS)
- Israel Blancas (Red Hat)
- Avadhut Pisal (Logicmonitor)
- Anusha (Apple)
- Alolita Sharma (Apple)
- Juan Vargas-Murillo (Splunk)
- Mikołaj Świątek (Sumo Logic)
- Kristina Pathak (Lightstep)

Agenda:

- [Jacob/Anu] This PR is solving a difficult issue, looking for thoughts!
- [Tyler Helmuth] Final reviews of Go Instrumentation Support: https://github.com/open-telemetry/opentelemetry-operator/pull/1555
- [Bryan Aguilar] open-telemetry/opentelemetry-operator [proposal] Use rust based cp tool rather than busybox in auto instrumentation images.
    - +1 Josh with Go based
-

2023/04/27, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Henrik Rexed (Dynatrace)
- Andy Keller (observIQ)
- Mikołaj Świątek (Sumo Logic)
- Tyler Helmuth (Honeycomb)
- Avadhut Pisal (LogicMonitor)
- Vineeth (Timescale)
- Israel Blancas (Red Hat)
- Raj Nishtala (sumo logic)
- Pavol Loffay (Red Hat)

Agenda:

- [Tyler] I didn't attend Kubecon, any cool news?
- [Jacob] This issue about auto-creating the cluster roles needed (link)
    - Someone installed operator w/ helm chart, but didn't make a clusterrole/rolebinding
    - Collector made by the operator doesn't have a default clusterrole created
    - I don't think we should be doing RBAC for people
    - Prometheus operator does create a cluster role and a binding for prometheus?
        - Not the operator, but the helm chart (kube-prometheus-stack) does.
            - https://github.com/open-telemetry/opentelemetry-helm-charts/issues/762#issuecomment-1523842859
            - The prometheus operator only recommends RBAC and then the chart uses it
        - This should be part of our kube-otel-stack :)
    - Should the operator chart have RBAC?
        - Operator chart has RBAC for the operator
- [Mikołaj] Anybody planning to resubmit https://github.com/open-telemetry/opentelemetry-operator/pull/1557?
    - Mikołaj is going to take this on, Jacob has abdicated the responsibility!
- [Jacob] Discuss embedding a pod template (issue link)
    - [Jacob] I believe that Bene may have brought up a similar issue some time ago and after some discussions with users at Kubecon I think it may be worth discussing the configuration of this.
    - [Tyler] If we allowed the template spec, would it mess up with the way things get auto-generated potentially
    - [Jacob] This isn't helm...
    - [Tyler] It's tedious to add all the fields/capability right now, which is painful
    - [Henrik] Doesn't see the value of embedding a template in to the spec
        - It would mean upgrades become more difficult too
    - [Andy] We could put those all top level, and then copy them over when needed
        - Maybe a top level field that we could merge in
        - Should this be behind a top level key?
    - [Tyler] Embedding it within is the easiest to keep up to date, but the actual configuration would look very funny
        - Collector is very flexible... what would the operator deny?
    - Consensus:
        - We don't want to embed the pod spec, there are fields in there we don't want to allow (extra containers, figuring out ports that we disallow/override, etc.)
        - It may make sense to pull common fields for the pod template in to a separate struct for better code organization, but inline it in yaml to prevent a diff
- [Vineeth] It's annoying that config in the spec is just a string that we do some hacks with
    - [Tyler] We can't validate config until run time
    - [Andy] Good example of this is when someone doesn't include a processor in the build
    - [Henrik] What about defining configuration as a separate thing that the collector pulls in via an annotation?
    - [Mikołaj] This makes sense for something like Prometheus or Fluent, but it's more difficult with the collector because it's a much larger scope

    - [Tyler] We shouldn't restrict _what_ people run, it would be pretty hard to do
    - [Mikołaj] The benefit of restricting is that we can more effectively separate operators and users. Bene was talking about more opinionated CRDs
    - [Henrik] Fluent lets you define pipelines as a top-level CRD
        - It can be difficult to understand how it gets built together
        - What if config is its own thing?
    - [Mikołaj] Maybe having different CRDs for different
    - [Jacob] bringing it back... can we run validation somehow?
        - [Tyler] every component needs to have a validation function?
        - [Tyler] We _may_ be able to pull in something from the collector to validate?
    - [Pavol] When you run the operator you need to provide the default image
        - Could the operator spin up a collector as part of the validation webhook CR which can create an instance of the container to run a check against?
        - Init container probably wouldn't do enough here, because that's post validation.
    - Consensus:
        - It would be great if we could do this, the how is difficult
        - What if we could generate the CRD from the existing config structs
            - Breaking changes would happen frequently!
        - Could we do something around validating the structure?
            - Maybe there's a middle ground where we could put in the "basic components" (receiver, processor, exporter) [Jacob makes issue]
            - If we separate config out to its own CRD, a bunch more options are available as far as what we're able to reference. [Jacob makes issue]
                - You could even reference multiple configs and then let the collector figure it out (already available)
                - A common use case is a customer using a base config and then overlaying

2023/04/13, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Pavol Loffay (Red Hat)
- Tyler Helmuth (Honeycomb)
- Nico Stewart (observIQ)
- Andy Keller (observIQ)
- Curtis Robert (Splunk)

Agenda:

- Otel go auto-inst (Tyler H)
    - Waiting on an image (hopefully released today)
        - Will update with image, retest everything
    - E2E testing
        - We're going to add go-inst behind a gate, default disabled
        - Operator manager will need to add this flag in
        - We may need to update the default bundle
        - Does the E2E test use the default bundle/manifest?
            - Yes

            - Pavol comment: e2e tests right now just tests the manifest files, doesn’t check the actual instrumentation - https://github.com/open-telemetry/opentelemetry-operator/issues/552
        - There’s an open ticket for better end to end testing
        - Default bundle will contain the gate _enabled_, cluster operators should be using the helm chart, NOT the default bundle.
    - We should merge this PR first which contains docs for the gates
        - Emit an event when someone tries to use the instrumentation when the flag is disabled
- Featuregating instrumentation languages (Tyler H)
    - Temp check?
        - Temp check green, cool with doing it
- Kubecon EU!
    - Bindplane OPAMP !
    - Tutorial for running Otel on Kubernetes
    - Will do a meetup during Telemetry day (Tuesday)
- Operator OpAMP
    - Jacob explained the current status
- Kube 1.27
    - Jacob will open and close an issue for this.

2023/03/30, 9 AM PT/6 PM CET

Attendees (company):

- Israel Blancas (Red Hat)
- Bryan Aguilar (AWS)
- Benedikt Bongartz (Red Hat)
- Pavol Loffay (Red Hat)
- Josh Voravong (Splunk)
- Tyler Helmuth (Honeycomb)
- Mikołaj Świątek (Sumo Logic)
- Kristina Pathak (Lightstep)
- Eric Hsueh (AWS)
- Curtis Robert (Splunk)
- Mike Dame (Google)

Agenda:

- Improve repository CVE detection tooling #1451 (Josh V)
    - https://github.com/aquasecurity/trivy
- BusyBox image for auto-instrumentation discussion #1600 (multiple people, Josh V)
    - Submit golang and rust PRs and then decide
- PR for adding Golang auto-instrumentation support https://github.com/open-telemetry/opentelemetry-operator/pull/1555 (Tyler H)
    - Review the PR
    - Put it behind feature gate
- Distributed configuration (Mikołaj)
    - Allow users to define parts of OTELcol config

2023/03/16, 9 AM PT/6 PM CET

Attendees (company):

- Pavol Loffay (Red Hat)
- Israel Blancas (Red Hat)
- Josh Voravong (Splunk)
- Mikołaj Świątek (Sumo Logic)
- Benedikt Bongartz (Red Hat)
-

Agenda:

- Mikolaj (sumo logic)
  - Optional product, mostly for auto-instrumentation and in the future replace prometheus stack with OTEL
  - Move more functionality from OTEL helm chart to OTEL CRDs
  - Telegraph sidecar that exposes JMX as p8s metrics - add it to operator
  - Filelog receiver used for k8s logs, perf is fine, filelog receiver is reliable
  - Maybe get somebody else to talk about auto-instrumentation in 2 weeks. Python and .net java, nodejs are supported
- Josh (splunk)
  - Currently shipping the OTEL helm charts
  - Used for auto-instrumentations (suppoted: java, .net, python, php). Use own distributions. Auto-instrumentation base images from otel use busybox, which have unnecessary dependencies.
  - Don’t use the collector part, collector is deployed via helm
  - Trying to move to filelog receiver
- Missing docs for collecting Prometheus metrics
  - Blog post or quickstart
  - TA directory should have a link to user guide
- P8s exporter with persistence
  - https://github.com/open-telemetry/opentelemetry-collector/tree/main/exporter/exporterhelper#persistent-queue
  - https://github.com/open-telemetry/opentelemetry-collector-contrib/tree/main/extension/storage/filestorage
  - On k8s statefulset has a PV

2023/03/02, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Benedikt Bongartz (Red Hat)
- Tyler Helmuth (Honeycomb)
- Alex Beach (New Relic)
- Vineeth (Timescale)
- Kristina Pathak (Lightstep)
- Mike Dame (Google)
- Eric Hsueh (AWS)
- Israel Blancas (Red Hat)

Agenda:

- [Tyler Helmuth] Go Auto Instrumentation
    - Checking in on if ^^ is possible
    - How do we communicate that go auto-inst is experimental?
        - Should we have an alpha flag system?
- [Alex Beach] Questions about operator use of https://github.com/open-telemetry/opentelemetry-collector-contrib/pull/16594
    - What's up with opamp and what's needed on the collector side?
        - Otel agent man working group (#otel-agentmanwg) is pushing through some requirements to make things work, above is the open PR and two linked issues
        - Changes to be made are potentially breaking
        - Until collector issues are fixed, opAmp extension can't be merged yet
    - OpAmp extension for the collector itself - why is this necessary for the opAmp bridge?
        - It's not 👍
        - Not everyone will be running collectors or wanting to use this management mechanism (opamp extension)
        - The opamp bridge would allow you to apply your CRD and not lose config because no CI/CD is reconciling the CRD/configuration
        - Future feature: connecting opamp extension to the bridge for healthchecks
            - Extension would talk to the bridge directly
- [Benedikt Bongartz] https://github.com/open-telemetry/opentelemetry-operator/pull/1540
-

2023/02/16, 9 AM PT/6 PM CET

Attendees (company):

- Benedikt Bongartz (Red hat)
- Jacob Aronoff (Lightstep)
- Anthony Mirabella (AWS)
- Israel Blancas (Red Hat)
- Kristina Pathak (Lightstep)
- Matej Gera (Coralogix)
- Eric Hsueh (AWS)

Agenda:

- [Bene] Simplify Operator usage (some thoughts)
    - https://github.com/open-telemetry/opentelemetry-operator/issues/1477
        - This may be a bit difficult to do because of custom collector configurations
        - What if we could mount configmaps as configurations
            - Doing this makes running an update for a collector more difficult
        - Would a better helm chart be useful here?
            - I have one here that is pretty opinionated
            - This doesn't help people using kustomize
        - Parts of the configuration can be error prone
            - It is difficult to know if the thing you want will work before applying
        - What are some common use cases
            - Scraping metrics
            - A big config of getting traces/metrics/logs?
        - Can we do configuration building/validation?

  - Folks have made internal builders to build collectors from UI
  - Maybe something like the caddy installer?
    - https://caddyserver.com/download
  - A user could configure the collector via UI and when they want the operator it's another box with preferences
    - How do you verify that a component is part of the builder
      - Maybe you also build an image for someone using the builder
  - Maybe this is just a remote config?
    - A remote server operator could provide a convenience to translate from desired features to actual configuration
  - What if this were a github action for the collector builder repo?
    - Someone could add it to their own CI/CD to auto-create a builder and publish for you
    - Maybe it would simplify the billing of this too :)
- First steps?
  - Start with a better helm chart for customizing the operator <has issue>
  - Better docs for what features are available and how they should be configured <has issue>
  - See collector best practices PR: https://github.com/open-telemetry/opentelemetry.io/pull/2312
  - Create some examples for common use cases <needs issue>
    - Run tests on the examples to ensure they are always valid / updated with code
    - ***Maybe this should be on the docs site!!!***
      - How do we keep this up to date in both places?
      - Not a good solution rn, people are working on this (phil and severin)
  - Open the PR to auto-configure the target allocator section of the collector config
- Lots of complexity here – seems like these problems are spread throughout a lot of otel
  - Many different ways to get this wrong, it's too easy to do the wrong thing
  - How do we make it easier to do the right thing?

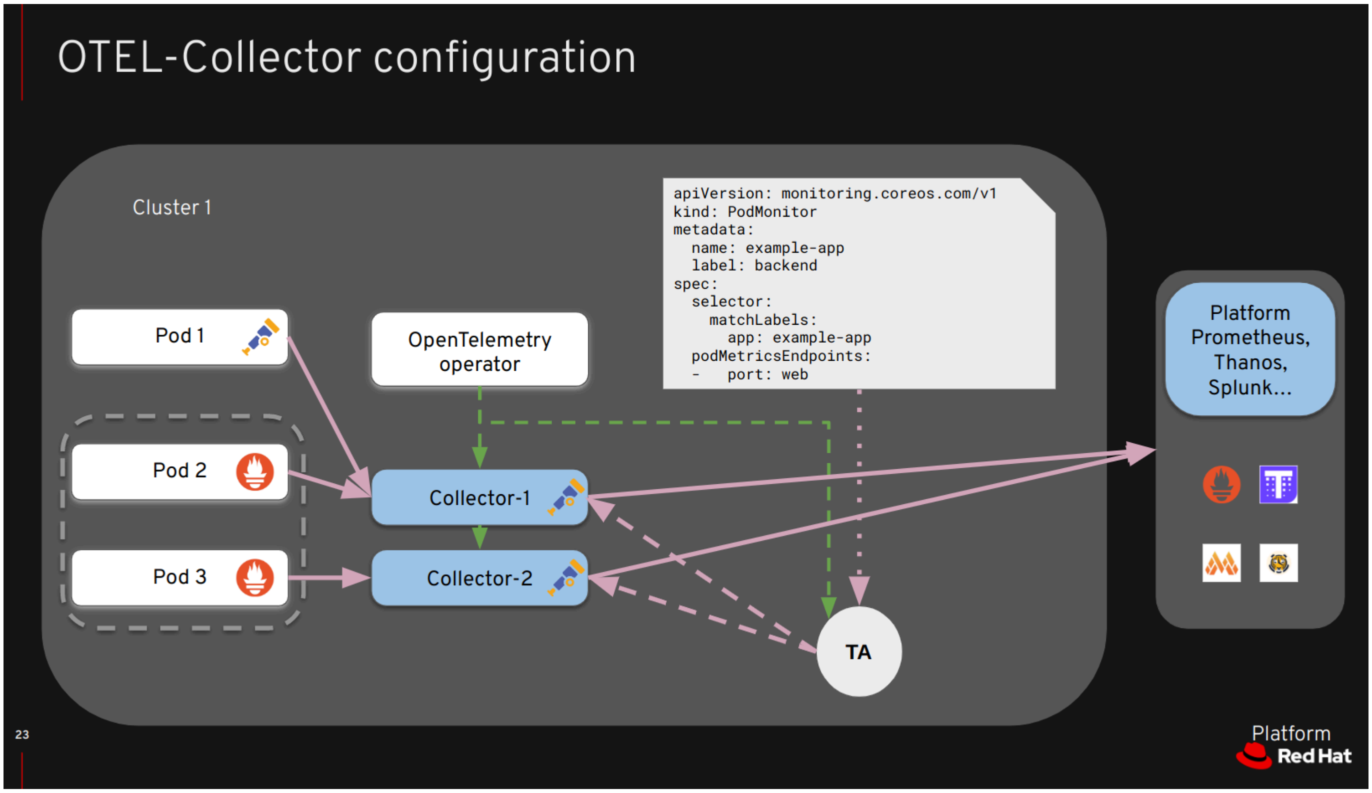
Notes:
2023/02/02, 9 AM PT/6 PM CET
Attendees (company):

- Pavol Loffay (Red Hat)
- Benedikt Bongartz (Red Hat)
- Israel Blancas (Red Hat)
- Kristina Pathak (Lightstep)
- Anthony Mirabella (AWS)
- Eric Hsueh (AWS)
- Jacob Aronoff (Lightstep)
- Mike Dame (Google)

Agenda:

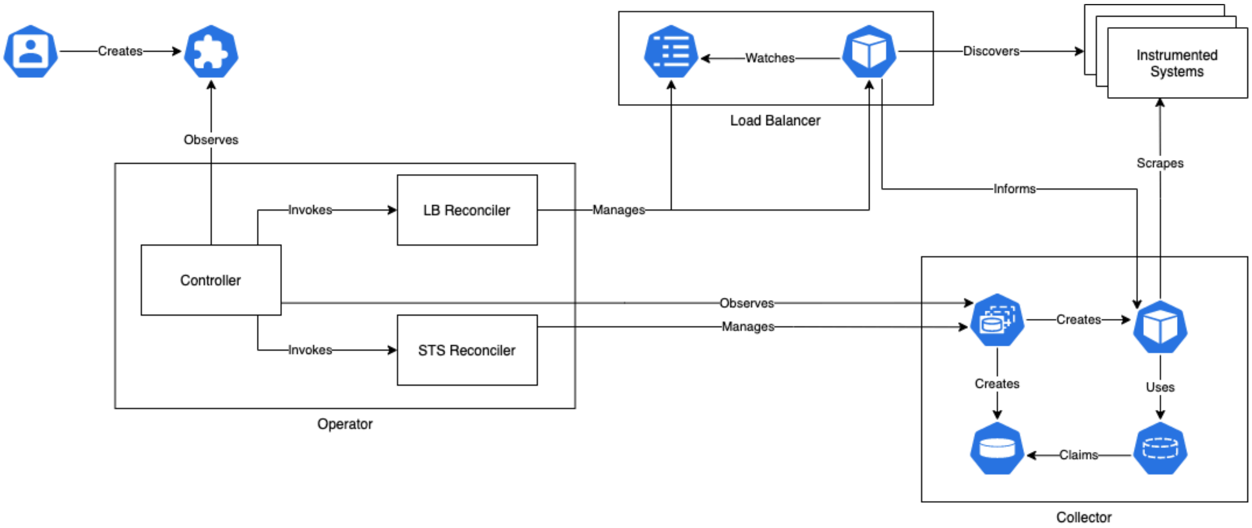
- [Pavol]: Release rotation/volunteers
- [Pavol]: Operator hub release
- [Bene] feedback on TA slides

Notes:

- rotate release responsibility - all maintainers and approvers will rotate release responsibility
- Automate the operator hub release PRs
    - release mangers are responsible for the PRs to be merged
    - This can probably be done using a [github script](#)
    - Example [here](#)
-


OTEL-Collector configuration

```
apiVersion: monitoring.coreos.com/v1
kind: PodMonitor
metadata:
  name: example-app
  label: backend
spec:
  selector:
    matchLabels:
      app: example-app
  podMetricsEndpoints:
  -   port: web
```

-


-

[2023/01/19, 9 AM PT/6 PM CET](#)

Attendees (company):

- Antoine Toulme (Splunk)
- Pavol Loffay (Red Hat)
- Anthony Mirabella (AWS)
- Benedikt Bongartz (Red Hat)
- Joshua Voravong (Splunk)
- Jacob Aronoff (Lightstep)
- Kristina Pathak (Lightstep)
- Ann Harter (Lightstep)
- Matej Gera (Coralogix)
- Mike Dame (Google)

Agenda:

- Target allocator CRD functionality / memory issues (jacob)
  - How should we fix this?
  - Can we move the functionality to the operator which will alleviate the load/complexity on everyone?
  - Options: 1. Fix p8s, 2.copy over p8s structs to the operator repo 3. Use yaml
  - Option 3. might be preferred
- Installing the operator and a collector CRD together
  - https://github.com/open-telemetry/opentelemetry-helm-charts/issues/69
    - Jacob will open the issue
    - Israel will move the check API script to readiness probe of the operator
      - The webhooks could be initially configured to allow the CR creation and then the readiness probe could change the config to the original/desired state
- ~~Feedback on some OpenTelemetry Operator + Kubernetes slides~~

2023/01/05, 9 AM PT/6 PM CET

Attendees (company):

Agenda:

- canceled - no agenda

2022/12/08, 9 AM PT/6 PM CET

Attendees (company):

- Jacob Aronoff (Lightstep)
- Benedikt Bongartz (Red Hat)
- Pavol Loffay (Red Hat)
- Anthony Mirabella (AWS)
- Kristina Pathak (Lightstep)
- Moh Osman (Lightstep)
- Gustavo Paiva (Lightstep)
- Michael Dong (Apple)
- Israel Blancas (Red Hat)

Agenda:

- Documentation (Jacob Aronoff)
  - no examples
  - no TA docs

    - website for quick starts, examples
    - readmes for detailed docs
    - community demo could be used asexample
- Target Allocator versioning (Kristina Pathak)
    - no gomod version for TA - useful for people building custom allocators
    - section for each sub-component e.g. TA
        - maybe use https://github.com/open-telemetry/opentelemetry-go-build-tools/tree/main/chloggen for changelog
- Remote Configuration (Jacob Aronoff)
    - OpAMP for the OpenTelemetry Operator