Kubernetes APIServer Network Proxy

Mailing List: https://groups.google.com/forum/#!forum/kubernetes-sig-cloud-provider

Meetings:
Biweekly Wednesday 12:00 PM ET / 9:00 AM PT / 16:00 UTC (Alternating with SIG Cloud Provider Meeting)
Zoom link: https://zoom.us/j/551547753

Next:

- GA requirements draft: https://github.com/kubernetes/enhancements/pull/4048

Kubernetes APIServer Network Proxy Meeting

Oct 30, 2024
Participants:

- Imran (ipochi) - Microsoft
- Walter (cheftako) - Google

Agenda:

- (ipochi) - Lease controller system crashes if token auth is not used.
    - Abstraction around authentication would be useful
    - Would require new cli flags for server to present itself as a client to apiserver

Oct 2, 2024

Participants:

- Walter (cheftako) - Google
- Azimjon (azimjohn) - Google
- Joseph (jkh52) - n/a

Agenda

- Lint error
    - Could test-infra kubekins image revs. be the culprit?
- Robot upgrade error
    - gomod-dependencies failing on lint, and
- Protobuf optimization
- Need 1.31.x versions / tags
-

Jul 24, 2024

Participants:

Agenda:

Jul 24, 2024

Participants:

- Walter (wfender) - Google
- Willow (carreter) - Google
- Avritt (avrittrohwer) - Google
- Azimjon (azimjohn) - Google

Agenda:

- Avritt - Release timeline once dynamic server count feature is merged (https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/635).
    - One of the github maintainers needs to cut a new version, takes ~days.
    - Only cut minor versions e.g. release-0.31.
    - Do we need to release KNP lease controller with the lease counting feature in agent?
        - No, as long as we guard the behavior with flag enablement.
- Azimjon - CVEs
    - CVE-2024-24790
        - Willow will bump to 1.22.5 in her PR (#639)

Notes:

- (Jparrill) Sorry I cannot attend today because another conflicting meeting happens during this one.

Jul 10, 2024

Participants:

- Willow (carreter) - Google
- Imran (ipochi) - Microsoft
- Azimjon (azimjohn) - Google

Agenda:

- Carreter - e2e testing
    - PR up to add e2e testing (#639)
    - ☑ ~~Need to confirm no new dependencies are being pulled in + see if there's a way to downgrade to go 1.22.2~~
- Carreter - dynamic proxy server count updates
    - Background: [PUBLIC] KNP Dynamic Proxy Server Count Design Doc
    - PR up to enable getting server count by counting leases (#635)
    - Blocked by e2e testing PR
    - Make lease controller for KNP servers next
-

Jun 26, 2024

Participants:

- Avritt - Google
- Willow (carreter) - Google

- Jparrill - Red Hat
- Walter - Google

Agenda:

- Carreter - dynamic proxy server count updates
    - Draft PR almost ready for review: #635
        - client-go versioning: want to work with oldest supported kube-apiserver version. Need to double check version support matrix

12 jun 2024

Participants:

- Jparrill - Red Hat
- Imran - Microsoft
- Avritt - Google
- Willow - Google

Agenda:

- Jparrill - Once a flow is sent through Konnectivity server to an agent, it's there any way to set a Proxy in the agent side to get the outgoing traffic forwarded following that path?
    - Document an issue upstream to track and get perspective from others.
- Carreter - Dynamic proxy server count (GitHub, design doc)
    - Cool improvement
    - Provide feedback on the doc. Raise github issue.
- http-connect: incorrect error propagation issue: #630
    - Client receives generic error message not the actual message.
    - Blind sided by actual error in the server-agent loop.

29 may 2024

Participants:

- Walter - Google
- Jparrill - Red Hat

17 abr 2024

Participants:

- Joseph - Google
- Walter - Google
- Jparrill - RedHat
- Avritt Rohwer - Google
- Azimjon - Google
- Mayank Kumar - Microsoft
- Wei Ling - Apple

Agenda:

- Joseph: team update
    - Avritt and Azim will take some of my responsibilities

- Joseph: agent drain PRs need review/feedback
- Avritt: will host an intern who will implement issue #<273>

Notes:
Avritt: on #273, the intern will have limited time (X weeks), how can we set them up for success contributing to apiserver-network-proxy in OSS?

- If changing kube-apiserver at all, will need a KEP
- Otherwise if change in contained in kubernetes-sigs/apiserver-network-proxy, likely only need a design doc with options
    - If making a protocol change between server and agent, detailed design doc needed. If no protocol change a one-pager likely will be sufficient
- Implementation options considered so far:
    - Have server write short-lived lease objects. Servers query for leases to obtain a count of servers.
        - Failure modes: what if leases cannot be written?
            - Would be really bad if lease writing / reading failures would not degrade gracefully. Would cause webhooks hosted in the cluster to not work.
            - Potential mitigation: --minimum_server_count that the servers use if lease writing / reading fails
    - Introduce --server_count_file that has a path server periodically reads from. Cluster admins are responsible for updating this file whenever server count changes.

3 abr 2024

Participants:

- Walter - Google
- Jparrill - Red Hat
- Wei Ling - Apple
- Mayank Kumar - Microsoft

Agenda:

-

Mar 20, 2024

Participants:

- Walter - Google
- Imran - Microsoft
- Joseph - Google
- Chris - Apple
- Jparrill - Red Hat

Agenda

- https://github.com/kubernetes-sigs/apiserver-network-proxy/issues/586
    - XfrSize - Why a constant?

    - https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/277
    - https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/269
    - Mea culpa. When I identified the length 1 channel as a per issue I ran a perf test and bumped till I stopped seeing an issue. Then figured if it wasn't enough we could add customization logic to allow the numbers to be bumped. However there was also an idea that we cannot arbitrarily bump these channel sizes. At some point if you need to increase the size it's a sign that something else is wrong. Why isn't the dequeue process able to keep up?

Notes

- Joseph: looks like there are two server uses (channels) and one agent, assuming we parameterize, should there be two server flags and one agent flag?
    - Walter: agent should have two as well, see: https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/310

Mar 6, 2024

Participants:

- Richard
- Jparrill - Red Hat
- Imran - Microsoft
- Joseph - Google

Agenda

- Joseph: time to create v0.30.* soon
    - https://github.com/kubernetes-sigs/apiserver-network-proxy/issues/574
- Joseph: agent drain feature: #566
- Imran - link to server readiness, is it only default that implements it
- Imran - I gave a talk on apiserver-network-proxy in Bangalore Meetup
    - A few silent users of apiserver-network-proxy

Feb 21, 2024

Participants:

- Walter - Google
- Joseph - Google
- Jparrill - Red Hat

Agenda

- Tactical
    - https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/567
    - https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/568

Feb 7, 2024

Participants:

- elmiko - Red Hat
- Walter - Google
- Joseph - Google

Agenda

- groom open PRs
- it would be nice for someone to take over TLS watcher: https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/501
-

Jan 24, 2024

Participants:

- ipochi - Microsoft
- elmiko - Red Hat
- Walter - Google
- Joseph - Google
- Jparrill - Red Hat

Agenda

- Joseph(jkh52) - a few tactical items:
  - #559 (disable broken github action)
  - #558 (would like to enable vendoring; any concerns?)

Jan 10, 2024

Participants

- Imran(ipochi) - Microsoft
- Joseph(jkh52) - Google

Agenda

- [ipochi] Blogpost: I'd like to write about the use case itself i.e Why or the need for apiserver-network-proxy.
  - some of it can be part of the README.
  - Talk about http-connect setup.
  - Get a draft out and share feedback in the next meeting.
- [Joseph] Looking to keep release PRs moving along
  - master: https://github.com/kubernetes/kubernetes/pull/122557
  - release-1.28: https://github.com/kubernetes/kubernetes/pull/120884

Dec 13, 2023

Participants

- elmiko - Red Hat
- Imran(ipochi) - Microsoft
- Walter - Google

Agenda

- [elmiko] i need to reduce my engagement here, i am working to get a few red hatters involved who are using konnectivity as part of their project.
- bunch of CVEs coming through, we are getting requests to upgrade go lang version and related
  - [walter] my old wip pr about alternate connection options might be worth resurrecting if we can get some help

    - [walter] also would be nice to get the tunnel re-use upgrades in place to reduce the need to extra handshakes
- [walter] maybe we should advertise the issues blocking GA on the readme page, perhaps help solicit more help
    - [imran] think we could also get a blog post going to reach out with this information. perhaps an announcement that could be broadcast internally/externally.
        - [elmiko] might be nice to include previous talks from kubecon as links
        - [ipochi] - think about topics that are blogpost worthy for the next meeting.
- next meeting (dec 27) is cancelled

Nov 15, 2023

Participants

- elmiko - Red Hat
- Walter - Google

Agenda

- https://github.com/kubernetes-sigs/apiserver-network-proxy/issues/538 fixed
    - https://github.com/kubernetes-sigs/apiserver-network-proxy/blob/master/Makefile#L30 still open
- Tim Allclair has a PR out to improve testing
    - looking for reviews
    - https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/536
- Walter - changes i think would be good
    - current assumption relies on a load balancer, but some users have indicated they would like alternative topologies. working on a WIP to allow multiple IP addresses so that other endpoints can be attempted.
    - current connection multiplexes agent and server traffic between proxy connection and server. between kapi server and proxy server an actual network connection is made between the two, this can result in many open connections (potentially thousands). for those using grpc, we should add the multiplexing behavior back in, this could result in performance gains (and also potentially avoid having too many file handles open) for grpc users.
    - we don't have a failure zone aware behavior between proxy server and proxy agent. in cases where the topology does not have one agent per kubelet, the fail over in agents is not the same as clusters with a one-to-one deployment. the lookup for agents could be improved by adding zonal awareness.

Nov 1, 2023

Participants

- elmiko - Red Hat
- Joseph, Walter - Google

Agenda

- reviewing open PRs, stuck on presubmit checks

    - blocking PRs on multi-arch
    - this might be the offending line, https://github.com/kubernetes-sigs/apiserver-network-proxy/blob/master/Makefile#L30
    - opened https://github.com/kubernetes-sigs/apiserver-network-proxy/issues/538
- [jhk] created a new release 0.28.1 to address CVE
    - we missed the k/k freeze date

Oct 18, 2023

Participants

- elmiko - Red Hat
- Walter - Google
- Imran - Microsoft

Agenda

- [elmiko] sync with jkh about slides for kubecon
    - will sync on slack
- reviewing backward compatibility concerns from PR 525
    - nuance from discussion, future path might see us having the server continue to accept connections in the same manner, but have the agent require setting agentID when it starts.
- [imran] agent identifier and multiple agents connecting
    - should we have provisions on the server to prevent an agent from connecting to itself
        - [walter] another detail here, could require authentication during connection process, then decisions could be made about how much trust to enable. this might make a lot of sense in multi-tenancy type situations.
        - [imran] will think about this more and if i come up with something will make a proposal to the project.
        - [walter] think it would be interesting to have some sort of failure domain routing.
        - [elmiko] is this like a hairpin prevention?
            - trying to solve for scenarios where agents live in different network namespaces, think control plane and customer.
            - possible bad scenario, with a sufficiently nefarious attack, they might be able to authenticate as an agent and then intercept webhook traffic from the control plane (for example).
- [imran] putting together a post mortem about the sync forever flag and related problems
    - [walter] would be cool to see this if can share
    - what types of problems?
        - expected delays when connecting
        - saw that not all agents were getting connected
        - during update scenarios saw that not all agents connected

    - [walter] this flag was designed to help inform agents when new servers come online, then the agents could fix their connectivity by learning about the servers.

Oct 4, 2023

Participants

- elmiko - Red Hat
- Joseph, Walter - Google

Agenda

- https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/525
    - there is a concern about backward compatibility with the change
    - related WIP: https://github.com/kubernetes-sigs/apiserver-network-proxy/compare/master...jkh52:apiserver-network-proxy:strategy-tests
    - Joseph's solution seems good for today, we might want to signal some sort of deprecation in the future.
        - changes to APIServer would require a KEP
        - other parts of the proxy might not **need** a KEP
    - temp IDs might be another approach, server creates the ID when it can't parse
        - this might lead to a memory leak if there is a load balancer in the middle
        - depends on who is owning the notion of identifying, the server or the agent
        - this is complicated
- Joseph updating templates

Sep 20, 2023

Participants:

- elmiko - Red Hat
- iaguis, ipochi - Microsoft

Agenda

- [ipochi]: question on 0.1.4 PR
    - Why are the steps mentioned in RELEASE.md not idempotent ?
        - ./hack/pin-dependency.sh sigs.k8s.io/apiserver-network-proxy/konnectivity-client v0.1.4
            - this should be idempotent, might need to be reviewed
            - [ipochi] ran multiple times and saw changes on subsequent runs
            - AI: open an issue with details about the failures
        - ./hack/update-codegen.sh
            - needs investigation about idempotency, should be idempotent
        - ./hack/update-vendor.sh

            - needs investigation about idempotency, this may not be idempotent as there could be changes in the upstream dependency version between runs.

- [ipochi] enhancement: a way to tell konnectivity-server for certain endpoints to not use the proxy:
    - We talked about it in the past in form of a KEP to EgressSelectorConfiguration
    - Question: What about another way to provide a similar list of endpoints to konnectivity-server and ask it to not proxy ? Much easier/less time to get it in konnectivity-server than k/k?
        - not quite sure how the “callback” would happen to inform apiserver about the change in method.
        - [tallclair] might need some investigation but it could be possible to build some sort of response to re-do the dial
        - [ipochi] if there is some way to handle this in konnectivity server then that will be a lighter process than updating k/k
        - [jkh] could be an argument against this proposal when contrast with the work that cheftako is doing.
        - AI: ipochi to create a list of use cases that might help inform about the proposed workflow
        - [jkh] think we should involve apimachinery in what we are proposing/designing to get input on the security implications
        - adding this as an alternative to an existing kep would be good
        - the code around this activity is informative about the complexity of the problem/solution
        - AI: ipochi reach out to apimachinery, if convenient
        - [tallclair] is this similar to what we are discussing?
            - https://github.com/kubernetes/kubernetes/issues/113926
        - tracking issue, https://github.com/kubernetes/kubernetes/issues/119002
        - AI: tallclair to review previous issues for similarity
- Action Items
    - ipochi, open an issue with details about the script failures
    - ipochi, create a list of use cases that might help inform about the proposed workflow for the EgressSelectorConfiguration
    - ipochi, reach out to apimachinery if their meeting times are convenient, if not sync with elmiko about bringing the issue up at their meeting
    - tallclair, review 113926 and 119002 for similarities

Sep 6, 2023

Participants:

- jkh52, ipochi, elmiko, cheftako,

Agenda

- [elmiko] branch naming

    - do we want to use “release-1.99” or “release-0.99” ?
        - No objection to using ‘release-0.99’
    - https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/518
- GA requirements draft: https://github.com/kubernetes/enhancements/pull/4048
    - tallclair has a PR up for version skew test coverage: https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/520
    - This is a progress towards the GA requirements.
    - Finer grain control is an api-machinery issue and not to be clubbed with the GA requirement.
- recent tags
    - 0.1.4 is proposed
    - want to move forward with new tagging scheme
    - would like to backport necessary changes into the 0.1
    - what to do about backporting? which version scheme to use?

Aug 23, 2023

Participants:

- elmiko, Joel Speed, jkh52, ipochi, cheftako

Agenda

- branching strategy
    - https://github.com/kubernetes-sigs/apiserver-network-proxy/issues/510
    - we are in agreement about moving forward with the plan
- recent nil deref: https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/514
    - need reviews here
- [cheftako] 0.2 (?0.28) branch to correspond to 1.28 and allow GRPC upgrade on konnectivity-client/go.mod (See https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/508)
    - we should follow the newly accepted branching guidelines from issue #510
    - open question, how will we address the community guidelines about having processes in place for backporting fixes, and similar. this requires approvals from release team members and we should figure out our strategy.
    - action item: if we can come to agreement here, we should codify our guidelines into the readme for the project. we should also codify our branching process to help inform.
        - lookup cherry-pick process for kubernetes
        - need to define our process for bring cherry-picks back to k/k
        - elmiko going to propose a PR
- connection management extensible https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/493
    - still need to do some work around failed connection states
    - tests need investigation
    - keeping the api small currently in expectation of future needs
    - looking for more reviews
- videos being uploaded
    - see the sig cloud provider playlist

Notes

- How does a cherry pick work?
    - konnectivity-client is vendored, but proxy-server and proxy-agent are configured via container images.
    - example k/k PR: https://github.com/kubernetes/kubernetes/pull/120029

Aug 9, 2023

Participants:

- ipochi, jkh52, JoelSpeed, cheftako, benluddy

Agenda

- Agent readiness: https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/485
    - Good to go. Thumbs up from Joseph and Walter.
- Branching strategy: https://github.com/kubernetes-sigs/apiserver-network-proxy/issues/510
    - cheftako: related liggitt PR https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/508
    - benluddy: publishing-bot: https://github.com/kubernetes/publishing-bot
- Kubecon
    - Joel is open to co-present but unfamiliar with Konnectivity
    - previous: https://www.youtube.com/watch?v=y0DBopR17-s and https://www.youtube.com/watch?v=0yltsB3Cbr4
- Imran - add as reviewer PR. Next steps? https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/507 https://github.com/kubernetes/org/issues/4368

Jul 26, 2023

Participants:

- elmiko, ipochi, jkh52, cheftako, benluddy

Agenda

- last few PRs blocked on a test issue
    - lint error coming up in CI
    - need a maintainer to followup, Joseph willing to help out
    - Imran has updated PR
    - sounds like we are hitting a known issue with golangci-lint
- can we add Imran to reviewers/maintainers list?
    - please open a PR
- Imran, added smart readiness check behind a flag
    - need to be careful about names here, readiness, health/liveness
        - readiness is generally ok to vary on
        - health/liveness we need to be careful about

- status updates on https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/493
    - Walter: will get back to this soon
    - there is some complexity around how to detect connections based on the way the agents connect to the load balancer, they won't know all the connections until the agents report back. this is part of the connection management package, and can be changed (eg in cases where a LB isn't used). 493 is implementing these changes to allow some variance.
    - most of heavy lifting is done
    - wrestling with some edge cases
- jkh52, would like to continue holding off on the next tag until we can get Imran's changes in.
    - should we switch to creating release branches that coincide with kubernetes releases?
    - part of path to GA is version skew strategy, so having release based branches will simplify this process.
    - jkh52 to make an issue describing the changes we are proposing to make

Jul 12, 2023

Participants:

- elmiko

Agenda

- cancelled due to lack of quorum

Jun 28, 2023

Participants:

- elmiko
- jkh52
- cheftako
- Benluddy
- deads2k

Agenda

- reviewing topics from last session
- Ability to control at a finer level in the EgressSelectorConfig
    - should be moved to k/k repo
    - won't have an impact on GA reqs, but we should keep track
- https://github.com/kubernetes/enhancements/pull/4089
    - merged
- delete stale uds file https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/476
    - would like to finalize decision
    - followup PR from jkh52, https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/498
        - prefer this as a solution

        - removed hold for merge
- GA requirements draft: https://github.com/kubernetes/enhancements/pull/4048
    - still need to get KEP into the new template format
- [jkh52] have some questions that have arisen when looking at new template about our branching strategy
    - connectivity client library in apiserver network proxy linked from k/k, causing some issues when contributors are updating the go.mod related to the kube version it is being compiled into. connectivity server and agent are sensitive to this as well.
    - current branching strategy is a little disjointed, might be nice in the future to have a more consistent strategy with respect to k/k releases.
    - how tightly do we want to couple?
    - [deads] like having binaries as separate projects
    - when should we create the new branches?
        - each release
        - CVEs
        - supported server version changes
    - [deads] if specific providers have maintenance needs around changing code in various versions, it should be completely acceptable to fork for those providers.
    - contributors can get confused about which branch they should be proposing changes to, there is some guidance in README but it isn't always followed
    - [cheftako] might follow a pattern similar to client-go where minor version tracks with k8s
- [cheftako] usage has two tunnel segments, conn server->conn agent pair is gRPC and creates HTTP connection for each pair(multiplexed). for api server -> conn server creates new connection for each. think we should consider having the same type of connection architecture with the apiserver-> conn server as we do for conn server -> conn agent.
    - [deads] this could be a significant change for folks, think we should consider having this in place as we go GA. should it be a GA requirement?
        - [cheftako] there is some complicated work that needs to be done to get this in place, it might not be able to be scheduled before GA. sceptical that this will get done in the next 6 months.
        - [jkh52] might need to confirm that proxy server can handle multiplexing
        - backward compatibility will add extra constraints on this change
        - [cheftako] would need a custom egress option for this
        - [deads] would like to see it feature gated as well

Jun 14, 2023

Participants: jkh52, iaguis, imran, benluddy, elmiko, cheftako

- Ability to control at a finer level in the EgressSelectorConfig
    - https://github.com/kubernetes-sigs/apiserver-network-proxy/issues/496

    - Right now it's an all or nothing approach for specifying if webhook and apiservice traffic goes through konnectivity or direct
- https://github.com/kubernetes/enhancements/pull/4089 cosmetic pass of KEP
- agent readiness check https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/485
- delete stale uds file https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/476
    - if no one cares/objects (next time), we should go forward with this
    - we could use a little more clarity on the history here, but it is proving difficult to find
- cheftako and elmiko to connect about getting recordings uploaded to youtube.

May 31, 2023

- Introductions
- GA requirements draft: https://github.com/kubernetes/enhancements/pull/4048
    - working towards updating for new template
    - looking to gather stakeholders
    - is overload protection needed for GA?
        - would be nice to have, not sure if necessary
- agent readiness check
    - standard kube readiness check, “can agent do its job?”
    - multiple behaviors using override flag to get second
    - One approach to the problem: PR link
- https://github.com/kubernetes-sigs/apiserver-network-proxy/pull/493 - How to support other connection modes
    - refactored code to allow for more connection methods
    - looking for reviews

May 17, 2023

- Introductions
- Need to look at GA’ing the original KEP (https://github.com/kubernetes/enhancements/tree/master/keps/sig-api-machinery/1281-network-proxy)
    - JKH taking initial steps with deads2k as advisor