

Harbor Operator DEV 04/07

07 April 2020 / 21:00 / ZOOM: <https://VMware.zoom.com.cn/j/4950184181>

ATTENDEES

Steven Zou@VMware

HuiJun Zhu@yunmanman

Canfeng Wang@Netease

Xiaoyang zhu@QingCloud (absent)

Weiwei He@VMware

AGENDA

Last Meeting Follow-up

<NONE>

Topics:

- CRD
- Reconcile flow
- Community operators issues

NOTES

- Harbor-operator is open sourced:
<https://github.com/goharbor/harbor-operator>
- Harbor cluster operator Github repo is created:
<https://github.com/goharbor/harbor-cluster-operator> (docs/scaffolded code)
- Storage operator: minIO (s3 compatible)
- **CONFIRMED:**
 - <https://github.com/zalando/postgres-operator>

- **CONFIRMED:**
 - <https://github.com/spotahome/redis-operator>
- Use the project to track project dev
 - All tasks should be linked with issues
- Code reviewers - need two
- Create `pkg` package for components and utilities
- Release 1.0
 - Goal: deploy full-stack Harbor
 - Provision
 - Update (scale)
 - Upgrade (lower v -> higher V / migration)
 - Delete
 - CI/CD pipeline (gorelease for release)
 - Documents
 - Rough timeline:
 - **Mid of June**

ACTION ITEMS

- **Communications with Harbor core operator:**
 - Storage: registry configuration file template
 - Database/Redis: secrets
 - New adding configurations: postponed to V1.0 (no GA date is planned)
 - Split tasks into two stages (1st version do not consider newly added configurations in the spec)
- Redis sentinel mode supported in Harbor
 - Sentinel PR is delayed to V2.1
 - Impacts to harbor cluster operator dev - [Ziming Zhang](#) supports
 - Patch registry (enable Sentinel)
 - Patch harbor (enable Sentinel)

- **Block cluster operator GA**
- **Work assignments:**
 - Reconcile flow design - Canfeng
 - **docs/reconcile_flow.nd - canfeng**
 - Sharing reconcile flow sharing
 - Need to refactor
 - Explicitly integration points (Redis, DB, storage with Harbor)
 - External services may be 0-3
 - Add notes to the diagram
 - Indicate when to do update
 - Indicate when to do a new deployment
 - Indicate we'll use Redis client to ping the Redis server to make sure Redis is available
 - Service name conventions
 - **Leave extending space for doing upgrading in future**
 - Found CR version is upgraded, then guide reconciles flow to upgrader interface (e.g.:
*redis.Upgrade(*oldSpec, *newSpec)*
(relase 0.5+)).
 - Aggregate stats of downstream CRs(db,redis et.) to Harbor status - Summarized by canfeng wang
 - E.g: add `Upgrading` status if we starting to update the downstream CRs
 - Add notes to the reconcile flow diagram
 - Complete CRD spec design

- docs/cr_HarborCluster_spec.md - co-work Xiaoyang &huijun
 - PR:
 - <https://github.com/goharbor/harbor-cluster-operator/pull/1>
 - Comments left
 - Suspend adding new missing configuration items e.g.: log, proxy, read_only.
 - add kind for Redis/database
 - For external, we only need a secret reference.
 - Document the secret content format to users.
 - **Merge date:** Within this week (04/10)
 - Next step: v1_types.go
 - Spec struct
 - Default admission controller
 - webhook
 - OpenAPI definition
 - //kuberbuild:xxxx
 - component.SetDefaults(spec *HarborClusterSpec)
 - Status spec struct
 - **Target date:** before next sync meeting (04/14) (PR raised)
- **Community operators issues**
 - K8s client for Redis operator is too old (1.15)
 - Dynamic(unstructured)/**typed client**
 - Raise an issue to the Redis operator community - No response till now.

- The only option for the current case: **Dynamic**
- No status spec
 - Use Redis client to connect to the sentinel node to check if it is ready
 - **[EXTENDED]** Create health observers to update the stats of components for a harbor cluster
- MinIO: need extra service creation (headless services so far)
 - Do investigation about the approach to check the availability of the minIO service
- Harbor core CR definition - Xiaoyang
 - Focus on storage, database, Redis
 - Make sure compatible with harbor operator spec
 - Cluster spec is a superset of harbor operator spec
- Based on current selections
 - SQL - huijun
 - REDIS - huijun

```
spec:
  redis:
    server:
      replicas: 3
    resources:
      cpu: 2000m
      memory: 2048Mi
      storage: 5Gi
    sentinel:
      replicas: 3
  db:
    storage: 1Gi
```

replicas: 2 version: "12"

- MINIO - xiao yang
 - PV re-mount of minIO
 - minIO service issue(headless service)
- Scaffold code - steven z (DONE)
 - CR Component manager interface - huijun zhu
 - DB implementation as an example
 - Redis implementation
 - Cluster spec struct/status spec struct - xiaoyang
 - minIO implementation based on the same manager interface

NEXT WEEK'S AGENDA

- Create a release project 0.5 - Steven Z (ongoing)
 - Create issues for tracking tasks
- Use PRs for review - ongoing