

Free up mon disk space

Check the following items for disk use:

1. journalctl

- a. Run `journalctl --disk-usage` to check disk use
- b. If needed, reduce journald's disk use via `journalctl --vacuum-size=<integer>M`

2. docker unused images

- a. Check for multiple images by running `docker image ls -a --digests` . If there are multiple images, then continue with this step. If not, skip to step 3 below.
- b. Check for any unused images by comparing outputs of
 - i. `docker ps -a --no-trunc --format 'table {{{.Image}}}\t{{{.ID}}}\t{{{.Status}}}\t{{{.Names}}}'`
 - ii. `docker image ls -a --digests`
 - iii. **Note** the digest of a docker image is shown after the @ in the IMAGE of `docker ps` output
- c. If there are images not being used by running containers, determine if any of those are still needed . There could be containers not currently running that are transient but still used.
- d. If there are images that are not being used by running containers and you have established that those images are no longer needed, then check for dangling images via `ls -lah /var/lib/docker/overlay2/` . Currently running images in overlay2 will have two directory entries with almost identical names expect that the second name will have `-init` appended. A directory that is listed once and does not end in `-init` is unused.
- e. Another method to find unused images is to check each running container for their name, image, and overlay2 paths via `docker inspect <container id from docker ps> | jq -r '.[] | .Name,.Image,.GraphDriver'` but again you need to ensure that any image not in current use is not transient and is actually no longer in use at all.
- f. If you are confident that there are overlay2 directories that are no longer needed, you can remove them and reclaim that disk space with `docker image prune`. See [👉 docker image pr](#)

une before running `docker image prune -a`.

3. docker logs

- a. By default, docker logs do not rotate and grow throughout the uptime of the container. Check `/etc/docker` for presence of `daemon.conf` file. If present, cat it and see if it contains `log-opts`, which control docker container log rotation.
- b. Check the size of container logs via `ls -lah /var/lib/docker/containers/*/*.log` Note that logs are lost on container exit.
- c. If there are large logs, check if it is safe to restart the container with the large log file(s). Note that the CONTAINER ID returned by `docker ps` is embedded in the path to the log file, ie `/var/lib/docker/containers/<CONTAINER ID>/<CONTAINER ID>-json.log`
- d. If safe to restart, then check if there is any reason to preserve the log file. Preserve if needed then restart the docker container via:
 - i. in one terminal, monitor `watch ceph -s`
 - ii. in another terminal, restart via `ceph orch daemon restart <ceph orch ps name>`

docker tips

To map running containers to the overlay dir they use, run

```
if [ -f rpt ]; then rm rpt; fi; echo "CONTAINER_ID NAME OVERLAY_DIR" >> rpt; for d in $(docker ps | grep -v CONTAINER | awk '{print $1}'); do name=$(docker inspect $d | jq -r '.[0] | .Name'); overlay=$(docker inspect $d | jq -r '.[0] | .GraphDriver.Data.MergedDir'); echo $d $name $overlay >> rpt; done; cat rpt | sed 's|/ceph|ceph|' | sed 's|/merged||' | column -t
```

To get a quick look at docker space use, run `docker system df`