🖵 **docker** / **distribution**

# private registry delete manifest results in broken state #1755

🛇 **Closed**   **yofreke** opened this issue on 31 May 2016 · 16 comments

---

**yofreke** commented on 31 May 2016

Saw some similar issues, but didnt find anything for this particular problem.

docker registry version: `registry:2`  Digest: `sha256:bf9b4a7b53a2f54c7b4d839103ca5be05b6a770ee0ba9c43e9ef23d602414f44`

I am using a private registry backed by S3, using basic auth.

**Steps:**

1. list tags
   do: `curl https://<regsitry>/v2/manager/tags/list`
   observe: `{"name":"manager","tags":[...,"1.3.0","1.3.1","latest"]}`

I want to delete tag `1.3.1`

2. get tag digest
   do: `curl https://<registry>/v2/manager/manifests/1.3.1 -v`
   observe: `< Docker-Content-Digest: sha256:4cd37f5633ac95dc32ceca3359a109f2c5a20517f2ded9eace4e12dccf58835a`

3. delete tag
   do: `curl -X DELETE https://<registry>/v2/manager/manifests/sha256:4cd37f5633ac95dc32ceca3359a109f2c5a20517f2ded9eace4e12dccf58835a -v`
   observe: `< HTTP/1.1 202 Accepted`

4. ensure tag is gone
   do: `curl https://<regsitry>/v2/manager/tags/list`
   observe: `{"name":"manager","tags":[...,"1.3.0","1.3.1","latest"]}`
   server logs:

   ```
   INFO[0907] response completed                          go.version=go1.6.2 http.request.host=
   <registry> http.request.id=63c217c1-b5ee-4d67-9a19-8fd79f210d87 http.request.method=GET
   http.request.remoteaddr=<my ip>:53486 http.request.uri=/v2/manager/tags/list
   http.request.useragent=curl/7.43.0 http.response.contenttype=application/json; charset=utf-8
   http.response.duration=52.464518ms http.response.status=200 http.response.written=200
   instance.id=6484f785-0dfd-4300-afde-13e8ad9da085 version=v2.4.1
   67.161.4.204 - - [31/May/2016:00:24:31 +0000] "GET /v2/manager/tags/list HTTP/1.1" 200 200 ""
   "curl/7.43.0"
   ```

5. investigate why tag is still there
   do: question life and the meaning of happiness, `curl https://<registry>/v2/manager/manifests/1.3.1 -v`
   observe: `{"errors":[{"code":"MANIFEST_UNKNOWN","message":"manifest unknown","detail": {"Name":"manager","Revision":"sha256:4cd37f5633ac95dc32ceca3359a109f2c5a20517f2ded9eace4e12dccf58835a"}}]}`
   server logs:

   ```
   ERRO[0940] response completed with error               auth.user.name=<username>
   err.code=manifest unknown err.detail=unknown manifest name=manager
   revision=sha256:4cd37f5633ac95dc32ceca3359a109f2c5a20517f2ded9eace4e12dccf58835a
   err.message=manifest unknown go.version=go1.6.2 http.request.host=<registry>
   http.request.id=2a33440c-2d08-49d0-9ee0-61a9e6cf4996 http.request.method=GET
   http.request.remoteaddr=<my ip>:53492 http.request.uri=/v2/manager/manifests/1.3.1
   http.request.useragent=curl/7.43.0 http.response.contenttype=application/json; charset=utf-8
   http.response.duration=30.85648ms http.response.status=404 http.response.written=183
   instance.id=6484f785-0dfd-4300-afde-13e8ad9da085 vars.name=manager vars.reference=1.3.1
   version=v2.4.1
   67.161.4.204 - - [31/May/2016:00:25:04 +0000] "GET /v2/manager/manifests/1.3.1 HTTP/1.1" 404 183
   "" "curl/7.43.0"
   ```

**Assignees**

No one assigned

**Labels**

None yet

**Projects**

None yet

**Milestone**

No milestone

**Notifications**

13 participants

Questions:
How do I *actually* remove this tag?

---

**RichardScothern** commented on 1 Jun 2016                                          `Contributor`

hi **@yofreke** this is not the expected behavior. When a manifest is removed, any tags which reference it are also removed.

If you have a minimal reproduction case could paste the file layout of your registry?

---

**yofreke** commented on 1 Jun 2016

The registry we are using has >100gb worth of data. I can do some investigating if there are specifics you are looking for?

---

**oxygen0211** commented on 1 Jun 2016

+1

Seeing the exact same problem with registry:2.4.1 (ID b0bfd772479f4e19f5546daf8c2a0c9673e699006de30d1f5b1623ee6d8d27f8) with a fairly big amount of images (~550GB).

Probably will have to ignore this error for now in my cleanup script so that our registry won't run out of HDD

---

**RichardScothern** commented on 2 Jun 2016                                          `Contributor`

**@yofreke** : i'm curious to see the repository layout, specifically the tags and manifests.

Otherwise, if you can post the registry logs with the most verbose setting that would help too.

---

**Grief** commented on 6 Jun 2016

I am observing the same issue, and I believe that the same steps lead me to it. But I cannot find any clue except that directories for tags still contain `link` files with `sha256` which were removed. I'd bet that `curl -X DELETE https://<registry>/v2/manager/manifests/sha256:4cd37f5633ac95dc32ceca3359a109f2c5a20517f2ded9eace4e12dccf58835a -v` didn't delete tags, but deleted manifest only.

---

**RichardScothern** commented on 7 Jun 2016                                          `Contributor`

The delete command removes the link file of the manifest and then the tags which are referencing that manifest. If you inspect the storage backend for the repository in question and can still see tags contain the manifest then there is a bug.

If you do see this, please post your registry version, logs from the delete operation and if possible the repository keys in the back-end. Without this information, debugging will be very difficult.

---

**kuijp** commented on 13 Jul 2016

+1

Having the same issue with same commands. Registry version 2.4.1

---

**1e3oss** commented on 13 Jul 2016

**FYI**
This command

```
curl -v https://<registry>/v2/<repo>/manifests/<tag>
```

does not provide the right tag digest for v2 deletion. You have to use a additional header as mentionned in the offical docs: https://docs.docker.com/registry/spec/api/#/deleting-an-image

```
curl -v https://<registry>/v2/<repo>/manifests/<tag> -H 'Accept:
application/vnd.docker.distribution.manifest.v2+json'
```

Proceeding this way fixed the issue for me.

👍 12

---

**jessegit** commented on 4 Jan 2017

This was happening to me, too. Initially tried to delete with registry:2.4.1 and some of the tags were stuck. Upgrade to registry:2.5.1 seems to have fixed it for me.

---

**stevvooe** commented on 11 Jan 2017            Contributor

Sounds like this is fixed in 2.5.1. Closing.

If anyone is still having issues, please let me know.

👍 1

---

**stevvooe** closed this on 11 Jan 2017

---

**jostyee** commented on 6 May 2017 • edited ▾

We're experiencing this bug in Registry v2.6.1.

```
curl -X GET "localhost:5000/v2/thirdparty/gitlab/tags/list"
{"name":"thirdparty/gitlab","tags":["8.17","9.0.5","9.1.1","9.1.2"]}
```

returns 4 tags, when fetching manifests of tag 8.17 as #1755 (comment) said, it failed:

```
curl -v "localhost:5000/v2/thirdparty/gitlab/manifests/8.17" -H 'Accept:
application/vnd.docker.distribution.manifest.v2+json'
* Hostname was NOT found in DNS cache
*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 5000 (#0)
> GET /v2/thirdparty/gitlab/manifests/8.17 HTTP/1.1
> User-Agent: curl/7.35.0
> Host: localhost:5000
> Accept: application/vnd.docker.distribution.manifest.v2+json
>
< HTTP/1.1 404 Not Found
< Content-Type: application/json; charset=utf-8
< Docker-Distribution-Api-Version: registry/2.0
< Date: Sat, 06 May 2017 06:49:36 GMT
< Content-Length: 193
<
{"errors":[{"code":"MANIFEST_UNKNOWN","message":"manifest unknown","detail":
{"Name":"thirdparty/gitlab","Revision":"sha256:1f2291aa0094f379a331772b3e34dfd68d26594055388d74877d

* Connection #0 to host localhost left intact
```

docker logs of registry as below:

```
time="2017-05-06T06:54:53Z" level=error msg="response completed with error" err.code="manifest
unknown" err.detail="unknown manifest name=thirdparty/gitlab
revision=sha256:1f2291aa0094f379a331772b3e34dfd68d26594055388d74877d5d05b2de72c0"
err.message="manifest unknown" go.version=go1.7.3 http.request.host="localhost:5000"
http.request.id=dff260f1-b89a-450a-bad5-7446fd7f7283 http.request.method=GET
http.request.remoteaddr="172.18.0.1:50522"
http.request.uri="/v2/thirdparty/gitlab/manifests/8.17" http.request.useragent="curl/7.35.0"
```

```
http.response.contenttype="application/json; charset=utf-8" http.response.duration=45.967892ms
http.response.status=404 http.response.written=193 instance.id=c02efd08-a6e7-4ac0-86ce-
75427ca2d6bb vars.name="thirdparty/gitlab" vars.reference=8.17 version=v2.6.1
172.18.0.1 - - [06/May/2017:06:54:53 +0000] "GET /v2/thirdparty/gitlab/manifests/8.17 HTTP/1.1"
404 193 "" "curl/7.35.0"
```

How can I delete this tag now?

👍 **4**      💙 **1**

---

**RainerW** commented on 8 Sep 2017 • edited ▾

Got the same problem in v.2.6.2. Running the garbage collector will also not free diskspace ... not good!
Removing the Manifest Blob folder will also free disk, but this requires FS access which means the script
knows how the registry is storing data!

---

**Kanshiroron** commented on 3 Oct 2017 • edited ▾

@stevvooe
+1 with v2.6.2

---

**stevvooe** commented on 4 Oct 2017                                              `Contributor`

@hinshun Could you please take a peak at this issue?

👍 **1**

---

**hinshun** commented on 5 Oct 2017                                               `Contributor`

Hi all, thanks for reporting the issue. Unfortunately I was unable to reproduce the issue.

I made a minimal `config.yml` where I deployed registry with:
```
docker run --rm -it -v $(pwd)/config.yml:/etc/docker/registry/config.yml -p 5000:5000 --name
registry registry:2.6.2
```

Configuration:

```
version: 0.1
log:
  fields:
    service: registry
storage:
  delete:
    enabled: true
  s3:
    region: <stripped>
    bucket: <stripped>
    accesskey: <stripped>
    secretkey: <stripped>
http:
  addr: :5000
```

My steps:

```
$ docker pull alpine:3.5
$ docker pull alpine:3.6
$ docker tag alpine 3.5 localhost:5000/some/repo:1
$ docker tag alpine 3.6 localhost:5000/some/repo:2
$ docker push localhost:5000/some/repo:1
$ docker push localhost:5000/some/repo:2

$ curl localhost:5000/v2/some/repo/tags/list
{"name":"some/repo","tags":["1","2"]}
$ curl -v "localhost:5000/v2/some/repo/manifests/1" -H 'Accept:
application/vnd.docker.distribution.manifest.v2+json'
*   Trying ::1...
* TCP_NODELAY set
* Connected to localhost (::1) port 5000 (#0)
> GET /v2/some/repo/manifests/1 HTTP/1.1
> Host: localhost:5000
```

```
> User-Agent: curl/7.54.1
> Accept: application/vnd.docker.distribution.manifest.v2+json
>
< HTTP/1.1 200 OK
< Content-Length: 528
< Content-Type: application/vnd.docker.distribution.manifest.v2+json
< Docker-Content-Digest: sha256:b12d242b6f881c97dbed5205d130b3b17307595c90b27cee79b1b3ea34cd9e2b
< Docker-Distribution-Api-Version: registry/2.0
< Etag: "sha256:b12d242b6f881c97dbed5205d130b3b17307595c90b27cee79b1b3ea34cd9e2b"
< Date: Wed, 04 Oct 2017 21:22:20 GMT
<
{
    "schemaVersion": 2,
    "mediaType": "application/vnd.docker.distribution.manifest.v2+json",
    "config": {
        "mediaType": "application/vnd.docker.container.image.v1+json",
        "size": 1511,
        "digest": "sha256:37c7be7a096b71f4e12e957c5ac83c3a7b01d8ae9f43a68c319e298d61b608fe"
    },
    "layers": [
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 1970271,
            "digest": "sha256:ab14e39f58e6d8ba465d2bb577a82a750ec0bcd2342b380920f9e7f307be3c4f"
        }
    ]
* Connection #0 to host localhost left intact
}

$ curl -X DELETE -v
localhost:5000/v2/some/repo/manifests/sha256:b12d242b6f881c97dbed5205d130b3b17307595c90b27cee79b1b3

*   Trying ::1...
* TCP_NODELAY set
* Connected to localhost (::1) port 5000 (#0)
> DELETE
/v2/some/repo/manifests/sha256:b12d242b6f881c97dbed5205d130b3b17307595c90b27cee79b1b3ea34cd9e2b
HTTP/1.1
> Host: localhost:5000
> User-Agent: curl/7.54.1
> Accept: */*
>
< HTTP/1.1 202 Accepted
< Docker-Distribution-Api-Version: registry/2.0
< Date: Wed, 04 Oct 2017 21:23:15 GMT
< Content-Length: 0
< Content-Type: text/plain; charset=utf-8
<
* Connection #0 to host localhost left intact

$ curl localhost:5000/v2/some/repo/tags/list
{"name":"some/repo","tags":["2"]}
```

If you are still experiencing problems, please open a new issue and provide your configuration and steps to reproduce the issue. It's likely there are differences between my minimal setup and specific configurations that triggering the bug.

👍 2

**Davidrjx** commented on 23 Jan • edited ▾

i also use registry 2.6.2 and following **@hinshun** 's solution, it works for me.
however, i am not clear whether an image is fully deleted from disk and what the cleanup operation
contains via `curl -X DELETE https_server:port/v2/<name>/manifests/digest` .
so i think both manifests and blobs are erased ,that indicates an image from local private registry is deleted
as `docker image rm` , right?
still please some experts provide some ideas,thanks