# Container Support - Issue #8905

## The pulp container fails to take the updated remote token.

06/17/2021 06:32 AM - sandeepc1988

| | | | |
|---|---|---|---|
| **Status:** | CLOSED - NOTABUG | **Start date:** | |
| **Priority:** | Normal | **Due date:** | |
| **Assignee:** | | **Estimated time:** | 0:00 hour |
| **Category:** | | | |
| **Sprint/Milestone:** | | | |
| **Severity:** | 3. High | **Sprint Candidate:** | No |
| **Platform Release:** | | **Tags:** | |
| **OS:** | | **Sprint:** | |
| **Triaged:** | No | **Quarter:** | |
| **Groomed:** | No | | |

### Description

- The remote API call is invoked to configure the AWS ECR container registry with username and password.
- AWS token expiration is 12 hrs.
- Once the token is expired, When the docker pull request is invoked on the pulp server, the remote returns a 403 forbidden error.
- Now a Remote API call is patched with a new valid AWS token, the docker pull request is expected to pull the image from the remote.
- But pulp container fails to take the updated token and fails to pull the image.

Observation:
The pulp container plugin is not handling 403 forbidden errors.

### History

#### #1 - 06/17/2021 08:03 AM - sandeepc1988

sandeepc1988 wrote:

> - The remote API call is invoked to configure the AWS ECR container registry with username and password.
> - AWS token expiration is 12 hrs.
> - Once the token is expired, When the docker pull request is invoked on the pulp server, the remote returns a 403 forbidden error.
> - Now a Remote API call is patched with a new valid AWS token, the docker pull request is expected to pull the image from the remote.
> - But pulp container fails to take the updated token and fails to pull the image.
>
> Observation:
> The pulp container plugin is not handling 403 forbidden errors.

Update:

The Basic auth credentials is cached here:
https://github.com/pulp/pulp_container/blob/af01b442daa7a47b2ffc4674eafbc7a0331e2fa0/pulp_container/app/downloaders.py#L56 with the expired token AWS registry always returns 403.

The new Token is not read from remote until 401 error occurs.
https://github.com/pulp/pulp_container/blob/af01b442daa7a47b2ffc4674eafbc7a0331e2fa0/pulp_container/app/downloaders.py#L67

Clearing the cache for error 403 would result in 401 responses on retry. This forces code to pick the new token from the remote. We Need a better way of handling basic auth credentials rotation/update.

#### #2 - 06/22/2021 02:05 PM - ipanova@redhat.com

sandeepc1988 wrote:

> sandeepc1988 wrote:
>
> > - The remote API call is invoked to configure the AWS ECR container registry with username and password.
> > - AWS token expiration is 12 hrs.
> > - Once the token is expired, When the docker pull request is invoked on the pulp server, the remote returns a 403 forbidden error.
> > - Now a Remote API call is patched with a new valid AWS token, the docker pull request is expected to pull the image from the remote.

- But pulp container fails to take the updated token and fails to pull the image.

Observation:
The pulp container plugin is not handling 403 forbidden errors.

Update:

The Basic auth credentials is cached here: https://github.com/pulp/pulp_container/blob/af01b442daa7a47b2ffc4674eafbc7a0331e2fa0/pulp_container/app/downloaders.py#L56 with the expired token AWS registry always returns 403.

Token auth has been implemented following these specs and the code relies on 401 auth response https://docs.docker.com/registry/spec/auth/token/. Whether the token is invalid because it has expired or has wrong format, or has insufficient scope a 401 is expected. www-authenticate header tells where to fetch a new token together with the scope details, if a 403 is returned that header does not contain any of the information and client would not know where to retrieve the token

```
$ curl -i -H "Authorization:Bearer $ACCESS_TOKEN"  'https://registry-1.docker.io/v2/_catalog' -L
HTTP/1.1 401 Unauthorized
content-type: application/json
docker-distribution-api-version: registry/2.0
www-authenticate: Bearer realm="https://auth.docker.io/token",service="registry.docker.io",scope="registry:catalog:*",error="insufficient_scope"
date: Tue, 22 Jun 2021 11:46:28 GMT
content-length: 145
strict-transport-security: max-age=31536000

{"errors":[{"code":"UNAUTHORIZED","message":"authentication required","detail":[{"Type":"registry","Class":"","Name":"catalog","Action":"*"}]}]}

$ curl -i -H "Authorization:Bearer $ACCESS_TOKEN"  'https://registry-1.docker.io/v2/library/busybox/manifests/latest' -L
HTTP/1.1 401 Unauthorized
content-type: application/json
docker-distribution-api-version: registry/2.0
www-authenticate: Bearer realm="https://auth.docker.io/token",service="registry.docker.io",scope="repository:library/busybox:pull",error="invalid_token"
date: Tue, 22 Jun 2021 11:50:24 GMT
content-length: 158
strict-transport-security: max-age=31536000

{"errors":[{"code":"UNAUTHORIZED","message":"authentication required","detail":[{"Type":"repository","Class":"","Name":"library/busybox","Action":"pull"}]}]}
```

The new Token is not read from remote until 401 error occurs.
https://github.com/pulp/pulp_container/blob/af01b442daa7a47b2ffc4674eafbc7a0331e2fa0/pulp_container/app/downloaders.py#L67

Clearing the cache for error 403 would result in 401 responses on retry. This forces code to pick the new token from the remote. We Need a better way of handling basic auth credentials  rotation/update.

**#3 - 07/19/2021 04:29 PM - ipanova@redhat.com**

*- Status changed from NEW to CLOSED - NOTABUG*

closing per explanation in the previous comment.