Container Support - Issue #7462

auth token used in registry requests are not validated properly when behind an SSL proxy

09/04/2020 08:35 PM - jsherril@redhat.com

Status:	CLOSED - CURRENTRELEASE	**Start date:**	
Priority:	High	**Due date:**	
Assignee:	dkliban@redhat.com	**Estimated time:**	0:00 hour
Category:			
Sprint/Milestone:	2.0.1		
Severity:	2. Medium	**Sprint Candidate:**	No
Platform Release:		**Tags:**	Katello
OS:		**Sprint:**	Sprint 81
Triaged:	No	**Quarter:**	
Groomed:	No		

Description

When the content app is running over HTTP deployed behind a reverse proxy running on HTTPS, the content app does not properly detect the url that the request is coming in as and fails to validate the provided auth token.

As an example, my CONTENT_ORIGIN is set to "https://hostname.example.com", a fetch to the api, would generate an auth token based on this URL:

https://hostname.example.com/pulp/container/default_organization-foo-foobar/manifests/latest

however, when the request is redirected to that URL, and proxied back to the content app, the content app tries to get the URL that is coming in and gets a url that looks like:

http://hostname.example.com/pulp/container/default_organization-foo-foobar/manifests/latest?validate_token=BLAH

It seems to think that the request is coming in over 'http' since the content app is running over http. In addition, it seems to be ignoring the port #, so if CONTENT_ORIGIN was configured to a non-standard port, comparison would also fail (although this is not a concern for us at this time).

I think it makes sense to just create and validate the token based off the request path, and ignore hostname, protocol, and port

Some relevant links showing why/where this is a problem:
https://docs.aiohttp.org/en/stable/web_advanced.html#deploying-behind-a-proxy
https://github.com/aio-libs/aiohttp/blob/72b5344af59ccdefc1e9a9e489c222a2b2592c7d/aiohttp/web_request.py#L384-L386
https://github.com/pulp/pulp_container/blob/789979a8706c527e4c9e8d04d084b262ea78208f/pulp_container/app/models.py#L442

Associated revisions

Revision b6d54914 - 09/08/2020 05:43 PM - dkliban@redhat.com

Use the path and query string of the URL to generate a 'validate_token'.

fixes: #7462 https://pulp.plan.io/issues/7462

Revision b6d54914 - 09/08/2020 05:43 PM - dkliban@redhat.com

Use the path and query string of the URL to generate a 'validate_token'.

fixes: #7462 https://pulp.plan.io/issues/7462

Revision e6b9e932 - 09/08/2020 06:21 PM - dkliban@redhat.com

Use the path and query string of the URL to generate a 'validate_token'.

fixes: #7462 https://pulp.plan.io/issues/7462

Revision e6b9e932 - 09/08/2020 06:21 PM - dkliban@redhat.com

Use the path and query string of the URL to generate a 'validate_token'.

fixes: #7462 https://pulp.plan.io/issues/7462

History

#1 - 09/04/2020 08:36 PM - jsherril@redhat.com

- Description updated

#2 - 09/04/2020 11:36 PM - dkliban@redhat.com

I was able to reproduce on my machine. I had to ensure that the 'CONTENT_ORIGIN' setting was set to https://localhost

#3 - 09/07/2020 03:47 PM - ekohl

A common way with reverse proxy setups is https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Forwarded-Proto. Looks like Django has docs on this and https://docs.djangoproject.com/en/2.2/ref/settings/#secure-proxy-ssl-header suggests to use this setting:

SECURE_PROXY_SSL_HEADER = ('HTTP_X_FORWARDED_PROTO', 'https')

#4 - 09/08/2020 03:29 PM - mdellweg

jsherril@redhat.com wrote:

> [...] I think it makes sense to just create and validate the token based off the request path, and ignore hostname, protocol, and port [...]

I agree. This would be compatible with whatever strange url rewrite rules a user might apply on his reverse proxy (well almost, but we need to assume something for the hash).

#5 - 09/08/2020 04:44 PM - pulpbot

- Status changed from NEW to POST

PR: https://github.com/pulp/pulp_container/pull/146

#6 - 09/08/2020 05:52 PM - dkliban@redhat.com

- Assignee set to dkliban@redhat.com

#7 - 09/08/2020 05:53 PM - dkliban@redhat.com

- Sprint set to Sprint 81

#8 - 09/08/2020 05:56 PM - dkliban@redhat.com

- Status changed from POST to MODIFIED

Applied in changeset b6d5491405f33df4da46d07143473b3005f680a7.

#9 - 09/08/2020 06:05 PM - pulpbot

PR: https://github.com/pulp/pulp_container/pull/147

#10 - 09/08/2020 07:49 PM - dkliban@redhat.com

- Sprint/Milestone set to 2.0.1

#11 - 09/08/2020 09:07 PM - pulpbot

- Status changed from MODIFIED to CLOSED - CURRENTRELEASE