

Arvados - Feature #9968

[Crunch2] [API] Support container request expiry (expires_at)

09/07/2016 03:32 PM - Tom Clegg

Status:	New	Start date:	09/07/2016
Priority:	Normal	Due date:	
Assigned To:		% Done:	0%
Category:	Crunch	Estimated time:	0.00 hour
Target version:	Arvados Future Sprints		

Description

See [Containers API](#)

Implementation has two parts:

Check for past expiry when creating/updating container requests

Filter CRs on expires_at is null or expires_at>now

This will stop a container when its requesting CR is updated to expires_at=now, for example.

Check expiry periodically

This can be implemented as a Rails script that runs as a cron job (or systemd timer).

A single postgres query should be able to return the containers whose priority has gone out of sync due to expired requests. Something like this:

```
select c.uuid from containers c
  join container_requests cr on (cr.container_uuid=c.uuid and cr.expires_at is null or cr.expires_at>now)
 where c.priority > 0 and c.state in ('Queued', 'Locked', 'Running')
 having cr.container_uuid is null or max(cr.priority)<c.priority
```

Run c.update_priority! on the selected containers.

History

#1 - 08/29/2017 01:34 PM - Tom Morris

- Target version set to Arvados Future Sprints

#2 - 04/25/2018 03:12 PM - Tom Morris

- Release deleted (11)