Arvados - Feature #13493
Federated record retrieval
05/16/2018 03:15 PM - Peter Amstutz

Status: Resolved
Priority: Normal
Assigned To: Peter Amstutz
Category: 
Target version: 2018-10-03 Sprint
Start date: 06/28/2018
Due date: 
% Done: 100%
Estimated time: 0.00 hour

Description
From Federation implementation roadmap

When retrieving a workflow record for arvados-cwl-runner, API server notices workflow UUID is remote, and fetches it with salted token (instead of looking in local DB)

Implementation notes

- This is a feature of the new (Go) API server (#13497)
- Implement as http handler middleware ("proxy to remote cluster if applicable, otherwise fall through to next handler")

Configuration

- Add configs (similar to Rails API's remote_hosts and remote_hosts_via_dns) to cluster config object in source/sdk/go/arvados/config.go

Subtasks:

<table>
<thead>
<tr>
<th>Task #</th>
<th>Description</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>13660</td>
<td>Review 13493-federation-proxy</td>
<td>Resolved</td>
</tr>
<tr>
<td>14045</td>
<td>Document federation features / design</td>
<td>Resolved</td>
</tr>
<tr>
<td>14054</td>
<td>Review 13493-document-federation</td>
<td>Resolved</td>
</tr>
</tbody>
</table>

Related issues:

| Related to Arvados Epics - Story #9053: Port API server to Go | New | 08/01/2020 | 02/01/2021 |
| Related to Arvados - Feature #13993: [API] Fetch remote-hosted collection by ... | Resolved | 09/07/2018 |
| Blocked by Arvados - Story #13497: [API] Initial "arvados-controller" server ... | Resolved | 06/15/2018 |

Associated revisions

Revision b3b3d214 - 07/20/2018 03:11 PM - Tom Clegg
Merge branch '13493-federation-proxy'
refs #13493

Arvados-DCO-1.1-Signed-off-by: Tom Clegg <tclegg@veritasgenetics.com>

Revision ab92b051 - 09/20/2018 06:13 PM - Peter Amstutz
Merge branch '13493-document-federation' refs #13493

Arvados-DCO-1.1-Signed-off-by: Peter Amstutz <pamstutz@veritasgenetics.com>

History

#1 - 05/16/2018 03:15 PM - Peter Amstutz
- Status changed from New to In Progress

#2 - 05/16/2018 03:20 PM - Peter Amstutz
- Description updated

#3 - 05/16/2018 03:45 PM - Peter Amstutz
- Description updated
Doesn't cover all possible permutations of {issuer of client auth, uuid of object being retrieved, cluster receiving proxyable request}. In the context of using arvados-cwl-runner on cluster A, loading a workflow from cluster B:

- Works if your token was issued by cluster A (i.e., your user account is also on cluster A)
- Works if your token was issued by cluster B, and has not been restricted to only work on cluster A (afaik there is no way to get a UI to offer this pair of host/token env vars)
- Doesn't work if your token was issued by cluster B and has been restricted to only work on cluster A (e.g., by logging in to Workbench B, following a link to Workbench A, and using "current token" in the nav menu)
- Doesn't work if your token was issued by cluster C (to make this work, API server needs to start issuing v2 tokens)

- Doesn't support object types other than workflow -- the idea is to use federation to host workflows on cluster A and use them on clusters B..E, even though we haven't yet dealt with all the complications/implications of proxying other kinds of requests

The test suite is a bit counterintuitive: unlike most tests, "zzzzz" is the remote cluster, and "zhome" is the local cluster whose proxy behavior is being tested, so that the remote cluster can be hooked up to the Rails server provided by run-tests.sh, which is always "zzzzz".

### #14 - 07/02/2018 09:25 PM - Peter Amstutz

- Missing support for looking up remote clusters by DNS.
- SaltToken() returns ErrObsoleteToken for "plain" tokens, so it depends on the API server giving out "v2:" tokens (or the client converting them to 'v2' tokens) but I don't think the API server does that yet.
- Decides if a token is salted or not based on only the token length, which seems likely to cause problems in the future. Could we tweak the v2 token format so there is a way to unambiguously indicate a token is salted?
- Going a bit further, did we ever investigate using something like JSON Web Tokens (JWT) for Arvados tokens? This would provide a standard structured format for token payload and metadata.
- It appears that if the token was provided in the query string as ?api_token=... it is not filtered out, and will be passed along in addition to the salted token in the Authorization header.
Not sure whether/when we'll need this, but that's right, it's not included here.

- SaltToken() returns ErrObsoleteToken for "plain" tokens, so it depends on the API server giving out "v2/" tokens (or the client converting them to 'v2' tokens) but I don't think the API server does that yet.

That's right, that's where the "cluster C" restriction above comes from.

- Decides if a token is salted or not based on only the token length, which seems likely to cause problems in the future. Could we tweak the v2 token format so there is a way to unambiguously indicate a token is salted?

We could, although length==40 seems unambiguous. What kind of problems do you have in mind?

- Going a bit further, did we ever investigate using something like JSON Web Tokens (JWT) for Arvados tokens? This would provide a standard structured format for token payload and metadata.

We could investigate that. But this is probably not the best place to open a discussion. Wiki?

- It appears that if the token was provided in the query string as ?api_token=... it is not filtered out, and will be passed along in addition to the salted token in the Authorization header.

Indeed. Will fix.

#16 - 07/03/2018 03:17 PM - Tom Clegg
- Target version changed from 2018-07-03 Sprint to 2018-07-18 Sprint

#17 - 07/03/2018 03:27 PM - Tom Clegg
- Story points changed from 2.0 to 1.0

#18 - 07/03/2018 06:18 PM - Tom Clegg
13493-federation-proxy @ 64a357242cc7ba9fb877a9b3d622160318098c5chthttps://ci.curoverse.com/job/developer-run-tests/782/

- removes ?api_token=...
- cleanup tests, add "mocked remote" for testing request details like the query string

#19 - 07/05/2018 01:53 PM - Peter Amstutz
Tom Clegg wrote:

- Missing support for looking up remote clusters by DNS.

Not sure whether/when we'll need this, but yes, it's missing.

The API server supports it already, so why wouldn't we do it now?

- SaltToken() returns ErrObsoleteToken for "plain" tokens, so it depends on the API server giving out "v2/" tokens (or the client converting them to 'v2' tokens) but I don't think the API server does that yet.

That's right, that's where the "cluster C" restriction above comes from.

That doesn't address my comment, if you get a "classic" token from cluster A, you still can't use it for federation unless some as-yet-unwritten client code converts it to "v2/" format first. So it won't actually work?

- Decides if a token is salted or not based on only the token length, which seems likely to cause problems in the future. Could we tweak the v2 token format so there is a way to unambiguously indicate a token is salted?

We could, although length==40 seems unambiguous. What kind of problems do you have in mind?

Because we don't specify the plain token is required to have a certain length that != 40, it would be legal to have an unsalted token that is 40 characters long. If we're going to rely on a particular behavior we should document it.
Going a bit further, did we ever investigate using something like JSON Web Tokens (JWT) for Arvados tokens? This would provide a standard structured format for token payload and metadata.

We could investigate that.

I'll do a little research and see if it makes sense for us.

#20 - 07/05/2018 01:55 PM - Peter Amstutz
Also: please add a documentation page under "Architecture" that describes the current state of federation features.

#21 - 07/05/2018 02:40 PM - Peter Amstutz
Also there should be a documentation page describing how to install & configure arvados-controller (even if it is marked EXPERIMENTAL).

#22 - 07/05/2018 02:59 PM - Peter Amstutz
Comments about testing:

- Looks like we test GET and PATCH, what about POST, DELETE, or weird methods like OPTIONS?
- It looks like TestRemoteError tests for connection error, how about a test where the remote returns 404?
- Maybe errors talking to clusters be returned as "Bad Gateway" not "Internal Server Error"?

#23 - 07/05/2018 03:22 PM - Tom Clegg
Peter Amstutz wrote:

- Missing support for looking up remote clusters by DNS.

The API server supports it already, so why wouldn't we do it now?

The API server supports accepting and validating tokens issued by *.arvadosapi.com. Here, a wildcard would support proxying requests to *.arvadosapi.com. Both are "looking up by DNS" but they are different features.

I guess the reason not to do it now would be that we have other things to do now. It might be better to think about this in the context of #13619.

- SaltToken() returns ErrObsoleteToken for "plain" tokens, so it depends on the API server giving out "v2" tokens (or the client converting them to 'v2' tokens) but I don't think the API server does that yet.

That's right, that's where the "cluster C" restriction above comes from.

That doesn't address my comment, if you get a "classic" token from cluster A, you still can't use it for federation unless some as-yet-unwritten client code converts it to "v2" format first. So it won't actually work?

Aha. I thought "it" in your comment referred to SaltToken(), I didn't know you were thinking of the "token issued by A" scenario.

Indeed, for that to work, we need to either fix the FIXME in saltAuthToken(), or start issuing v2 tokens (and make sure Workbench recognizes/salts them properly).

```go
if err == auth.ErrObsoleteToken {
    // FIXME: If the token exists in our own database,
    // salt it for the remote. Otherwise, assume it was
    // issued by the remote, and pass it through
    // unmodified.
    token = creds.Tokens[0]
}
```

If we're going to rely on a particular behavior we should document it.

Agreed. Added API token format

#24 - 07/05/2018 09:29 PM - Tom Clegg
Peter Amstutz wrote:

Also there should be a documentation page describing how to install & configure arvados-controller (even if it is marked EXPERIMENTAL).
Yes, this is happening on #13497. Waiting for ops feedback about Installing controller service.

- Looks like we test GET and PATCH, what about POST, DELETE, or weird methods like OPTIONS?

Will add.

- It looks like TestRemoteError tests for connection error, how about a test where the remote returns 404?

TestNoAccess tests this (get object with no read access = 404).

- Maybe errors talking to clusters be returned as "Bad Gateway" not "Internal Server Error"?

Sounds good.

#25 - 07/16/2018 01:58 PM - Tom Clegg

13493-federation-proxy @ 66fc7401e777ac7ceff2b02cfc459b1368f429https://ci.curoverse.com/job/developer-run-tests/809/

- upgrade locally-issued v1 tokens to v2 so they can be salted for proxying (it's still desirable to move to issuing v2 tokens, but I don't want that
  and the associated Workbench updates to block this -- also, even after we do that, the transparent upgrade will allow old tokens to work as well
  as new tokens)
- test OPTIONS method (and updatedapiserver to return an empty body instead of "-"

#26 - 07/17/2018 07:40 PM - Peter Amstutz

s.remoteServer.Server.Handler.ServeHTTP(rec, req) // direct to remote -- can't proxy a create req because no uid

You could look for workflow.owner_uuid and use the the cluster id from that. (Not going to block the branch on this, but we should think about it.)

    err = db.QueryRowContext(req.Context(),`SELECT uuid FROM api_client_authorizations WHERE api_token=$1
AND (expires_at IS NULL OR expires_at > current_timestamp) LIMIT 1`, aca.APIToken).Scan(&aca.UUID)

This line is really long

```ruby
def empty
  render text: "-"
+  render text: ""
end
```

This is awesome.

So far this looks good. I'm going to do some manual testing / playing around with arvbox.

#27 - 07/18/2018 03:20 PM - Tom Clegg

- Target version changed from 2018-07-18 Sprint to 2018-08-01 Sprint

#28 - 07/20/2018 01:41 PM - Peter Amstutz

I'm happy to say after a bunch or arvbox hacking (adding support for arvados-controller) I was able to do a federated query between two arvbox
instances! Both the "arv" command line and workbench were able to retrieve a record from a federated instance.

LGTM!

#29 - 08/01/2018 03:24 PM - Tom Clegg

- Target version changed from 2018-08-01 Sprint to 2018-08-15 Sprint

#30 - 08/02/2018 04:36 AM - Tom Clegg

Peter Amstutz wrote:

    Also: please add a documentation page under "Architecture" that describes the current state of federation features.

The architecture section seems like a good place to explain how federation works under the hood, but would it be more appropriate to put the "feature
is available" info in a usage context?

- "Running registered workflows at the command line" (https://doc.arvados.org/user/tutorials/writing-cwl-workflow.html)
- "GET workflow API" (https://doc.arvados.org/api/methods/workflows.html)
I suppose these also have a dependency on docs for setting up the relevant parts of the cluster config and Rails API config.

#31 - 08/08/2018 03:13 PM - Peter Amstutz
- Related to Feature #13993: [API] Fetch remote-hosted collection by UUID added

#32 - 08/15/2018 03:16 PM - Tom Clegg
- Assigned To deleted (Tom Clegg)
- Target version changed from 2018-08-15 Sprint to 2018-09-05 Sprint

#33 - 08/15/2018 04:26 PM - Tom Clegg
- Assigned To set to Peter Amstutz

#34 - 09/05/2018 02:49 PM - Peter Amstutz
- Target version changed from 2018-09-05 Sprint to 2018-09-19 Sprint

#35 - 09/18/2018 08:41 PM - Peter Amstutz
13493-document-federation @ 5d8eb6ead9e2ca32f424a8379485a902cf09f
First pass at documenting the mechanics of federation: cluster ids, salted tokens, proxying record & keep block requests.

#36 - 09/18/2018 09:28 PM - Tom Clegg
There's a lot of information here about how things have been implemented under the hood, which is fine if we can keep it up to date.

We should probably mention that most of the federation story isn't actually implemented/supported yet -- just some specific parts, and with certain simplifying assumptions -- and that the behaviors described here should not be interpreted as a stable API.

This seems like a good place to put some (badly needed) sensible rules for choosing a cluster ID.

- For throw-away testing purposes, use "z****"
- For experimental/local clusters that will never be referenced by the outside world, use "x****"
- For long-lived clusters, contact us to get an ID that won't conflict with anyone else in the future

#37 - 09/19/2018 02:48 PM - Peter Amstutz
Tom Clegg wrote:

There's a lot of information here about how things have been implemented under the hood, which is fine if we can keep it up to date.

We should probably mention that most of the federation story isn't actually implemented/supported yet -- just some specific parts, and with certain simplifying assumptions -- and that the behaviors described here should not be interpreted as a stable API.

This seems like a good place to put some (badly needed) sensible rules for choosing a cluster ID.

- For throw-away testing purposes, use "z****"
- For experimental/local clusters that will never be referenced by the outside world, use "x****"
- For long-lived clusters, contact us to get an ID that won't conflict with anyone else in the future

13493-document-federation @ 898cd195f8b5916ed9c3a83dec212e858bfa610f updated taking these comments into account, also adds missing diagram

#38 - 09/19/2018 03:22 PM - Tom Morris
- Target version changed from 2018-09-19 Sprint to 2018-10-03 Sprint

#39 - 09/20/2018 04:24 PM - Tom Clegg
LGTM, thanks

#40 - 09/20/2018 06:18 PM - Peter Amstutz
- Status changed from In Progress to Resolved

#41 - 11/13/2018 08:49 PM - Tom Morris
- Release set to 14