

## Arvados - Story #5824

### [Workbench] [Keep] collection browse/download server

04/24/2015 08:35 PM - Tom Clegg

<b>Status:</b>	Resolved	<b>Start date:</b>	05/21/2015
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assigned To:</b>	Tom Clegg	<b>% Done:</b>	100%
<b>Category:</b>	Keep	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	2015-11-11 sprint		

#### Description

Location

- [source:services/keep-web](#)
- deb/rpm package name: keep-web

Authentication mechanisms needed

- Accept Authorization: OAuth2 {token} header, and set CORS headers<sup>1</sup> (suitable for aware clients, like an in-browser app)
- HTTP basic authentication just like arv-git-httpd (probably easier to use with general purpose tools: curl, wget, .netrc)
- reader\_token in path (suitable for "get a link to share" downloads)
- Configured anonymous token (try this if other auths don't work)

Download URIs mirror Workbench's

- <http://dl.{domain}/collections/{uuid}/path/file.ext>
- [http://dl.{domain}/collections/{portable\\_data\\_hash}/path/file.ext](http://dl.{domain}/collections/{portable_data_hash}/path/file.ext)
- <http://dl.{domain}/collections/download/{uuid}/{token}/path/file.ext>
- [http://dl.{domain}/collections/download/{portable\\_data\\_hash}/{token}/path/file.ext](http://dl.{domain}/collections/download/{portable_data_hash}/{token}/path/file.ext)

MIME types

- Look up file extension with [mime.TypeByExtension\(\)](#)

Deployment

- Install docs & [source:docker/](#) need to run keep-web

Nice to have: directory listings

- Very basic formatting (bootstrap would be nice but can wait)
- OK if we show a directory listing only at the top level, similar to the way Workbench does it.
- OK not to support directory listings *at all* (at least for the first version), and just let Workbench do that part?
- Explicitly no index.html in place of directory listings.
  - This means "wget -r" can always be expected to work, but it also ensures this can't be used to host entire web sites along the lines of github pages. If we add this capability in the future we could add a -render-index flag. This would make more sense if it came with a bunch of other features anyway, like dynamically mapping vhosts to collections.

<sup>1</sup> Do not accept cookies with CORS requests!

#### Subtasks:

Task # 6070: Refactor arv-git-httpd authenticator into a package	<b>Resolved</b>
Task # 5851: Write integration tests	<b>Resolved</b>
Task # 6327: Set mime type header	<b>Resolved</b>
Task # 6952: Review 5824-go-sdk	<b>Resolved</b>
Task # 6131: Sort out browser and command line authentication mechanisms	<b>Resolved</b>
Task # 5852: Write install docs	<b>Resolved</b>
Task # 7555: Workbench integration tests	<b>Resolved</b>
Task # 7700: Add -anonymous-token flag	<b>Resolved</b>
Task # 7744: Add keep_web_download_url config	<b>Resolved</b>
Task # 7708: Add upgrade notes	<b>Resolved</b>

Task # 7580: Accept CORS requests	<b>Closed</b>
Task # 7754: Fix rails-style query string, disposition[]=attachment	<b>Resolved</b>
Task # 7756: Decode + to space in keep-web, or encode space to %20 in workbench	<b>Resolved</b>
Task # 7554: Review	<b>Resolved</b>
<b>Related issues:</b>	
Related to Arvados - Bug #5426: [Workbench] Large downloads through workbench...	<b>Resolved</b> 03/18/2015
Related to Arvados - Bug #5521: [Workbench] Slow download speed from browser	<b>Closed</b> 03/20/2015
Related to Arvados - Bug #7743: [Workbench] Hide file preview buttons when pr...	<b>New</b> 05/21/2015
Related to Arvados - Story #8784: [Workbench] Use keep-web to generate direct...	<b>Resolved</b> 03/23/2016
Blocked by Arvados - Feature #6280: [SDKs] Go SDK can return an io.Reader for...	<b>Resolved</b> 06/10/2015
Blocks Arvados - Feature #6312: [Workbench] Use download service instead of p...	<b>Resolved</b> 06/10/2015

### Associated revisions

#### Revision 98d6a7c2 - 08/12/2015 05:55 PM - Tom Clegg

Merge branch '5824-go-sdk' refs #5824

#### Revision 057ea788 - 11/04/2015 04:54 AM - Tom Clegg

Merge branch '5824-keep-web'

refs #5824

#### Revision 057ea788 - 11/04/2015 04:54 AM - Tom Clegg

Merge branch '5824-keep-web'

refs #5824

#### Revision 21006cfa - 11/04/2015 04:55 AM - Tom Clegg

Merge branch '5824-keep-web'

refs #5824

#### Revision b85cb1ba - 11/05/2015 04:37 PM - Tom Clegg

Merge branch '5824-keep-web' refs #5824

#### Revision 1e2ace54 - 11/11/2015 05:14 PM - Tom Clegg

Merge branch '5824-keep-web-workbench' refs #5824

#### Revision 723ab702 - 11/11/2015 05:14 PM - Tom Clegg

Merge branch '5824-keep-web-workbench' refs #5824

#### Revision 723ab702 - 11/11/2015 05:14 PM - Tom Clegg

Merge branch '5824-keep-web-workbench' refs #5824

#### Revision 209d83a0 - 11/12/2015 08:00 PM - Tom Clegg

Merge branch '5824-keep-web-workbench' closes #5824

#### Revision 615100bb - 11/25/2015 03:49 PM - Tom Clegg

5824: Move run\_test\_server diag messages from stdout to stderr.

refs #5824

### History

#### #1 - 04/29/2015 07:27 PM - Peter Amstutz

- Assigned To set to Peter Amstutz

#### #2 - 04/29/2015 07:30 PM - Ward Vandewege

- Subject changed from [Download] collection browse/download server (probably go) to [Workbench] collection browse/download server (probably go)

### #3 - 04/29/2015 07:31 PM - Peter Amstutz

- Assigned To changed from Peter Amstutz to Tom Clegg

### #4 - 05/19/2015 10:01 PM - Brett Smith

- Target version changed from 2015-05-20 sprint to Arvados Future Sprints

### #5 - 06/11/2015 06:00 PM - Tom Clegg

- Description updated

### #6 - 08/10/2015 10:08 PM - Radhika Chippada

Review comments for 5824-go-sdk branch at [be4e5f225cf1615504ee2c1feb0f200d6904716c](https://github.com/Arvados/arvados-go-sdk/commit/be4e5f225cf1615504ee2c1feb0f200d6904716c)

- `func (c ArvadosClient) Get(resource string, uuid string, parameters Dict, output interface{}) (err error)`

Can you call "resource" as type or resource\_type? It is not clear to me that "resource" here refers to the object type instead of the uuid.

- The comment below:

```
// There's no endpoint for that because GET /type/ is
// the List API. If there were an endpoint, the
// response would be 404: no object has uuid == "".
```

This is a bit confusing. Can you say something like "you must provide uuid of that resource type ... and it cannot be empty ..." ?

- In `arvadosclient_test.go`, can you add a test with missing uuid? Also, not sure if there is already a test with invalid "resource type"; if not, can you please add one?
- Does this have to be `"sdk/go/httpserver/log.go"` ? Can it be `"sdk/go/util/log.go"` (package util instead of httpserver)?
- `auth.go` => This is not a big deal, but wondering if "LoadCredentialsFromHTTPRequest" might be more intuitive than "NewCredentialsFromHTTPRequest"
- I did not run the tests in `arv-git-httpd` etc. Am sure you tested them.

### #7 - 08/11/2015 01:06 AM - Tom Clegg

Radhika Chippada wrote:

Can you call "resource" as type or resource\_type? It is not clear to me that "resource" here refers to the object type instead of the uuid.

I agree this is a nonintuitive use of the word "resource" -- I was just staying consistent with the existing code. But it seems that the language of REST agrees with intuition: a resource is an object, not a type. I've updated to resourceType throughout, and made a bunch of other comment fixes too (mostly less repetition of the same argument descriptions).

```
// There's no endpoint for that because GET /type/ is
// the List API. If there were an endpoint, the
// response would be 404: no object has uuid == "".
```

This is a bit confusing. Can you say something like "you must provide uuid of that resource type ... and it cannot be empty ..." ?

Agree. I've rephrased it:

```
// No object has uuid == "": there is no need to make
// an API call. Furthermore, the HTTP request for such
// an API call would be "GET /arvados/v1/type/", which
// is liable to be misinterpreted as the List API.
```

- In `arvadosclient_test.go`, can you add a test with missing uuid? Also, not sure if there is already a test with invalid "resource type"; if not, can you please add one?

Good call, added both.

- Does this have to be `"sdk/go/httpserver/log.go"` ? Can it be `"sdk/go/util/log.go"` (package util instead of httpserver)?

I agree it's not httpserver-specific, but I don't like "util" any better (the package name "util" is specifically called out on <https://blog.golang.org/package-names>). Should it be "quotedlog"? Or maybe it's OK to leave it in httpserver for now since those are the only things we have that need to log client-provided/unsafe data?

FWIW, I think it's easy to move it out later without breaking clients by putting `var Log func(...interface{}) = newpkg.Log` in its place.

- `auth.go =>` This is not a big deal, but wondering if `"LoadCredentialsFromHTTPRequest"` might be more intuitive than `"NewCredentialsFromHTTPRequest"`

Currently you have two options:

- `c=NewCredentialsFromHTTPRequest(r)` -- more convenient if HTTP request is the only/first place you're getting tokens
- `c=NewCredentials(); c.LoadCredentialsFromHTTPRequest(r)` -- more flexible (maybe you don't want to load any from `httprequest`)

Are you suggesting deleting `NewCredentialsFromHTTPRequest` shortcut? Or does this just mean I should add doc comments?

- I did not run the tests in `arv-git-httpd` etc. Am sure you tested them.

Yes, but if this is any indication, I missed several commits that should have been on this branch (now added, starting at [f9e23d0](#) which has a much more reasonable version of `httpserver.Log()`) ...and that I'm now hoping you review as well.

#### #8 - 08/11/2015 03:27 AM - Radhika Chippada

- Tom said:

I agree it's not `httpserver`-specific, but I don't like "util" any better (the package name "util" is specifically called out on <https://blog.golang.org/package-names>). Should it be "quotedlog"? Or maybe it's OK to leave it in `httpserver` for now since those are the only things we have that need to log client-provided/unsafe data?

It seems like this is a good method that we could be using more in future though? If I am not dragging it, I think `quotedlog` seems appropriate package name. I would also assume you do not want to put it in the `logger` package, right? Please ignore this whole pointer if you think it is not worth it at this time. Thanks.

- Tom said:

Currently you have two options:

```
c=NewCredentialsFromHTTPRequest(r) -- more convenient if HTTP request is the only/first place you're getting tokens
c=NewCredentials(); c.LoadCredentialsFromHTTPRequest(r) -- more flexible (maybe you don't want to load any from httprequest)
```

Actually, I was asking if we want to rename `NewCredentialsFromHTTPRequest`. The second usage `"c=NewCredentials(); c.LoadCredentialsFromHTTPRequest(r)"` actually is using `"LoadTokensFromHTTPRequest"`, not `"LoadCredentialsFromHTTPRequest"` and hence this name is still available.

- Are `UUIDMatch` and `PDHMatch` used anywhere? Also, can we add a couple tests in `arvadosclient_test.go` for these?
- Get method with empty `UUID` - I see that this was existing functionality and I do not intend to add additional work. But, can't help wondering, should we return 404 or Argument error in such a scenario?
- Wish "unicorns" were valid type in `TestInvalidResourceType` :)
- All tests passed with `run-tests.sh`. (However, I can no longer run `arv-git-httpd` tests using "go test", but I think this is probably acceptable? Also, one `keepstore` test fails when I use "go test", but as I said, all tests pass with `run-tests.sh` and so going forward I can switch to testing using `run-tests` script.)

#### #9 - 08/11/2015 09:28 PM - Tom Clegg

Radhika Chippada wrote:

It seems like this is a good method that we could be using more in future though? If I am not dragging it, I think `quotedlog` seems appropriate package name. I would also assume you do not want to put it in the `logger` package, right? Please ignore this whole pointer if you think it is not worth it at this time. Thanks.

I'd like to avoid getting stuck on it right now, but I do think the Go SDK bits could use some help in this area. "Logger" does an entirely different kind of logging, and probably deserves a more suggestive name itself.

Actually, I was asking if we want to rename `NewCredentialsFromHTTPRequest`. The second usage `"c=NewCredentials(); c.LoadCredentialsFromHTTPRequest(r)"` actually is using `"LoadTokensFromHTTPRequest"`, not `"LoadCredentialsFromHTTPRequest"` and hence this name is still available.

Ah, I see. I think `NewX` is better because it matches the Go convention ("`NewX`" returns a new X)... vs. `LoadTokensFrom*` which modifies its receiver

by loading tokens from \*.

We could also rename "auth" to "credentials" and just call the factory function "New()", so calling it would look like "creds := credentials.New()" instead of "creds := auth.NewCredentials()"

- Are UUIDMatch and PDHMatch used anywhere? Also, can we add a couple tests in arvadosclient\_test.go for these?

Hm, these are a bit mysterious, aren't they. They're used by the yet-unseen 5824-keepdl branch, but this seemed like a better source of truth for them. Added tests in [094d247](#) + [493e39c](#).

- Get method with empty UUID - I see that this was existing functionality and I do not intend to add additional work. But, can't help wondering, should we return 404 or Argument error in such a scenario?

Yes, returning 404 when there wasn't really an HTTP request seems like a recipe for confusion. I've added ErrInvalidArgument and used it here and in Discovery(), which had a similar problem in its "discovery returned 200, parsed OK, but key not found" case. [2566020](#)

- Wish "unicorns" were valid type in TestInvalidResourceType :)

Let's file a separate issue for that...

- All tests passed with run-tests.sh. (However, I can no longer run arv-git-httpd tests using "go test", but I think this is probably acceptable? Also, one keepstore test fails when I use "go test", but as I said, all tests pass with run-tests.sh and so going forward I can switch to testing using run-tests script.)

For me, arv-git-httpd tests pass, but keepstore pull\_worker\_test.go tests fail. But they fail on master too.

Is this a "depends on test order" issue? It looks like pull\_worker\_integration\_test.go starts (and doesn't stop) an API server, so maybe pull\_worker\_test only works when the entire suite is under run-tests.sh, or it happens to run after pull\_worker\_integration\_test.

#### #10 - 08/12/2015 05:44 PM - Radhika Chippada

Branch 5824-go-sdk lgtm.

#### #11 - 09/30/2015 03:25 PM - Tom Clegg

Currently 5824-keepdl @ [014975c](#) has

- working downloader (test suite includes large files; also works in manual testing against real sites)
- install doc page

Needs

- a new name? Currently "keepdl", maybe "keep-web"?
  - "arv-web" is already taken by something completely different.
  - we have "keepproxy" and "keepstore" and "arvados-\*" packages. presumably this will be "apt-get install keepsomething"?
- hide/remove/label the install doc page so people don't unwittingly beta-test it
- add support in workbench (redirect existing download routes to (and point "download"/"preview" buttons to) configured download service)
- make the install docs more decisive about "uuid--dl.domain.example" vs. "uuid.dl.domain.example" choice?
- add example of how to set up wildcard DNS?
- document "download-only" mode (-attachment-only-host) in install docs (workbench config + run script). This makes it possible to download files without wildcard DNS/SSL; you just can't preview them in the browser.

#### #12 - 09/30/2015 03:40 PM - Brett Smith

Tom Clegg wrote:

Needs

- hide/remove/label the install doc page so people don't unwittingly beta-test it

Can we just leave it out of the TOC? I think I would consider that sufficiently hidden, personally.

- add support in workbench (redirect existing download routes to (and point "download"/"preview" buttons to) configured download service)

This can be a separate story after this is reviewed and merged.

- make the install docs more decisive about "uuid--dl.domain.example" vs. "uuid.dl.domain.example" choice?

Yes please.

- add example of how to set up wildcard DNS?

I feel strongly that this really needs to be out of scope for our install guide. Arvados doesn't require a particular DNS server; most installers will have already decided which they're going to run; there are many possible options and we can't reasonably document them all. This really doesn't seem like a good use of our time.

#### #13 - 10/13/2015 06:30 PM - Tom Clegg

- Subject changed from [Workbench] collection browse/download server (probably go) to [Workbench] [Keep] collection browse/download server (probably go)
- Category set to Keep
- Status changed from New to In Progress
- Target version changed from Arvados Future Sprints to 2015-10-28 sprint

#### #14 - 10/14/2015 07:09 PM - Tom Clegg

- Subject changed from [Workbench] [Keep] collection browse/download server (probably go) to [Workbench] [Keep] collection browse/download server
- Description updated

#### #15 - 10/22/2015 04:17 PM - Radhika Chippada

Review comments for branch 5824-keep-web at [5f822d9281e5afea05b1c87e59740687c0ee6692](https://github.com/Arvados/arvados/branch/5f822d9281e5afea05b1c87e59740687c0ee6692):

Given the complexity and size of this branch, I am breaking the review of this branch into two steps. (1) First about the usage and the process itself, and (2) the other for keepclient/collectionreader\* and manifest/manifest\* updates.

#### Install guide:

- Title: "Install the download server". Should we say: "Install the keep-web server" instead? Some of the description says "download". We might want to rephrase / revisit to say something like keep-web downloader ...
- Do we want to add "-g" to the command to generate anonymous token? Without -g, a new token is generated each time the command is executed. "bundle exec ./get\_anonymous\_user\_token.rb -g"
- This section title says "Tell the API server about the keep-web service", but the description below it refers "workbench config". Do we want to correct the discrepancy?

#### keepproxy doc update:

- The update "suitable for lower-bandwidth clients located elsewhere on the internet": since lower-bandwidth is not a "requirement", I think saying "suitable for clients located on the internet" would be sufficient?
- This "... internet: a client sends a single copy of a data block" seems like it could use a bit more explanation. Can you break it into a separate sentence and also add a precursor saying, "keepproxy can be used to do get or put data into keep ... When putting a block in keep, client sends a single copy of ...."

#### doc.go:

- Download URLs first line says **The following "same origin" URL patterns are supported for public collections**. Is this true that this is only allowed for public urls? For ex: [http://dl.example.com/c=uuid\\_or\\_pdh/t=TOKEN/path/file.txt](http://dl.example.com/c=uuid_or_pdh/t=TOKEN/path/file.txt) is used in the test to access FooCollection ... with token
- **The first form minimizes the cost and effort of deploying a wildcard TLS certificate for ...** which one is first form (<http://dl.example.com...?>). I think clarifying this will help
- **keep-web ignores everything after the first "." or "-"** This is a bit confusing. Obviously, this needs to be sent in the URL to reach the host, correct (for ex: any load balancer etc)? I think this needs a bit explanation ...
- **"uuid\_or\_pdh" part can be either a collection UUID or a portable data hash with the "+" character replaced by "-"** .... In server\_test.go => Test200, third test spec is using "/c=" + arvados.test.FooPdh + "/t=" + arvados.test.ActiveToken + "/foo" without replacing "+" with "-". I also played with all tests by not doing this conversion and commenting out the logic in handler. All tests pass. So, it appears that a user can use pdh in url without converting + to - char. What is the reason behind this requirement? I ask because, it would be an extra step for all clients.
- How would I download a file from a subdir in a collection? Let's say, I have a collection with ./subdir1/file1 and ./subdir2/file2, what would the URL to download file1? Would it be possible to add an example to doc (and may be a test)?
- Since I do not know much about nginx configuration, I am hoping the section "example nginx configuration" is well reviewed by you :)

#### Args to start server

- I think it would be useful to have all the supported / expected input arguments listed in one place in “starting the server” section. Right now, they are spread over the doc and hard to get to the bottom of it easily. I found `-address`, `-attachment-only-host`, `-trust-all-content` to be the list by reading through various bits.
- is the example for “`-attachment-only-host` correct? The following is what I am seeing and does not work with `-address` also:

```
$ keep-web -attachment-only-host localhost:8123
2015/10/22 09:49:36 listen tcp :80: bind: permission denied
```

- Please include an example with all three (if one could use `-trust-all-content` and `-attachment-only-host`) together (which I think is acceptable)

#### auth.go:

- “`api_token`” for cookie name sounds too non-specific. Suggest “`arvados_api_token`” instead.
- The comment “`LoadTokensFromHttpRequestBody`” seems out of place and makes it feel like it is about “`loadTokenFromCookie`” since it is right above this func. May be adding a small comment about `loadTokenFromCookie` might help distinguish the `LoadTokensFromHttpRequestBody` comment from the func following
- `services/keep-web/.gitignore` has “`keepdl`” in it

#### main.go

- Why did you set `ARAVADOS_API_TOKEN = xxx` in `init`? If an admin runs `keep-web` without setting this, do you intend it to not exit? If it is for test purposes, suggest moving it into an `init` in one of the tests.

#### server\_test.go -> Test404

- I spent a very long time to understand why the following works / not works, going back and forth in the test and the doc. I think adding some comments explaining why a path should 404 etc, and adding some more paths to the test would help. Some details:
- Add another path `"/collections/" + arvadosTest.FooCollection + "/foo"` right after `"/collections/" + arvadosTest.FooCollection + "/"` (with filename)
- Add an explanation to `"/collections/" + arvadosTest.FooCollection + "/"` that `FooCollection` is not public and hence it is not accessible with “`collections`” ...
- Add another path here: `"/c=<foo-collection-uuid/t=<some-bad-token>/foo` — add comment that it will 404 because of bad token
- Add another path such as `"/c=<foo-collection-uuid/t=<good-token>/thepertcountyconspiracy`
- Add `"/collections/" + arvadosTest.HelloWorldCollection`, and explain that even though publicly accessible, 404 because no file is specified
  - you might also want to consider adding this to the doc that entire collection cannot be downloaded, only individual files. Am I correct?
- Seems like adding another test with a bad anonymous token and accessing a public collection fails would be helpful

#### server\_test.go -> Test200

- I think it would be nice to add one more spec with `host: arvadosTest.FooCollection + ".dl.example.com"` right after the first spec with `host: arvadosTest.FooCollection + "--dl.example.com"`. Even though something similar is added later on in the list, having these two together might help see it right away

#### handler.go

- `defer clientPool.Put(arv)` : Would this fail if `arv` is nil? (Tests panic'ed when I used nil, btw)
- Is `Expires` needed when setting the cookie? Since setting the expiry makes the cookie persistent and I am thinking we can probably do with session cookies? This may help with users that do not allow persistent cookies.

#### handler\_test.go

- Why call it `mustParseURL` instead of `parseURL`?
- I personally think all “`authz`” can be just “`auth`”. It just reads / flows easily when reading and equally descriptive ...
- It appears that we are not checking the resp content in case of tests using `testVhostRedirectTokenToCookie` (sorry, if I am wrong). I think it would be great to verify the body has the expected content at least in cases such as attachment only etc.
- I did not yet do as thorough a job as `server_test.go` for this, though it looked comprehensive. I may make another pass at it later.
- There are some golint complaints in `keep-web` dir and the new `fixtures.go`

#### keep-dl versus keep-web

- It appears that "dl" is used in urls and test uuids etc. It might help to clarify that we used "dl" to mean "download" in our default usages such as "\*.dl.example.com" etc
- Another thought. A naive user might wonder: "is keep-web a web interface to keep? If so, what is the keepproxy for ..." I think we need to clarify the distinction. As we discussed, may be we need a wiki page that lists and briefly describes all supported services etc with links to individual pages ...

#### #16 - 10/24/2015 10:51 PM - Radhika Chippada

Review comments for the collectionreader and manifest updates in branch 5824-keep-web

I was able to understand the flow of the code and debug and verify the tests. Just a few minor comments.

- ErrNoManifest message "Collection has no manifest" seems a bit misleading; how about "collection manifest not provided / available"?
- I think adding some new lines / extra white space help improve readability of the very complex code in CollectionFileReader, cfReader and Read func. You may also consider adding some more comments in Read
- In collectionreader\_test, this looks a bit complicated. Can it be as follows?

```
-      c.Check(err, check.Not(check.IsNil))
+      c.Check(err, check.NotNil)
```

- Would be nice to add some extra white space and comments in the new code manifest.go as well
- golint suggestions in both packages (and also need to fmt collectionreader\_test)

#### #17 - 10/26/2015 12:48 PM - Radhika Chippada

Review comment for branch 5824-keep-web-workbench at [0f535af4fbc7a9f6ca1e25675b31cc9656d88f3](https://github.com/keep-web/keep-web/pull/17)

- Workbench config versus what install guide says:
  - First of, install guide says "API config" instead of "workbench config"
  - Install guide says "keep-web" where as workbench uses "keep\_web\_url" as the parameter name
  - In "Configure DNS" section, the Install guide says that when the third form "dl.uuid\_prefix.your.domain" is used, only public data is served. And, the third form is listed in "Tell the API server ..." However, it seems the workbench is using the given token if one is available, which means non-public data also is server? Should the **if Rails.configuration.keep\_web\_url** block in collections\_controller.rb -> show\_file check the format of the configuration to determine whether or not to use token? Also, the example in "application.default.yml" does not list the third form as example.

```
keep-web: https://{uuid_or_pdh}--dl.uuid_prefix.your.domain
keep-web: https://{uuid_or_pdh}.dl.uuid_prefix.your.domain
keep-web: https://dl.uuid_prefix.your.domain
```

- Same question about why replace "+" with "-" in pdh in the urls.
- collections\_controller\_test.rb -> I think it would be nice if we also do Rails.configuration.anonymous\_user\_token in "test No redirect to keep\_web\_url if collection not found" to verify that it fails even when anonymous token is configured
- download\_test.rb
  - Can the test be improved to use "id\_type" to cover both the uuid and pdh cases as in collection\_controller\_test?
  - The test did not pass for me (I get 502 Bad Gateway error); but I only ran this one test.
- Several collection related workbench tests failed; I did not rerun yet.

#### #18 - 10/28/2015 07:09 PM - Tom Clegg

- Story points changed from 2.0 to 0.5

#### #19 - 10/28/2015 07:09 PM - Tom Clegg

- Target version changed from 2015-10-28 sprint to 2015-11-11 sprint

#### #20 - 10/29/2015 08:36 PM - Tom Clegg

Radhika Chippada wrote:

- Title: "Install the download server". Should we say: "Install the keep-web server" instead? Some of the description says "download". We might want to rephrase / revisit to say the something like keep-web downloader ...

Changed to "Install the keep-web server".



- Do we want to add "-g" to the command to generate anonymous token? Without -g, a new token is generated each time the command is executed. "bundle exec ./get\_anonymous\_user\_token.rb -g"

I don't think there's any harm in generating different anonymous tokens for each service that needs one? It's conceivable (although a bit of a stretch, I admit) it could be useful to change/disable them independently on the API side, see which is which from logs, etc. I think "-g" would only be really important for a cron job, update script, etc. that runs many times.

- This section title says "Tell the API server about the keep-web service", but the description below it refers "workbench config". Do we want to correct the discrepancy?

Fixed.

#### keepproxy doc update:

- The update "suitable for lower-bandwidth clients located elsewhere on the internet": since lower-bandwidth is not a "requirement", I think saying "suitable for clients located on the internet" would be sufficient?
- This "... internet: a client sends a single copy of a data block" seems like it could use a bit more explanation. Can you break it into a separate sentence and also add a precursor saying, "keepproxy can be used to do get or put data into keep ... When putting a block in keep, client sends a single copy of ...."

True. Removed "lower-bandwidth" from the first sentence, and moved the specific advantages/differences into bullet points.

#### doc.go:

- Download URLs first line says **The following "same origin" URL patterns are supported for public collections**. Is this true that this is only allowed for public urls? For ex: [http://dl.example.com/c=uuid\\_or\\_pdh/t=TOKEN/path/file.txt](http://dl.example.com/c=uuid_or_pdh/t=TOKEN/path/file.txt) is used in the test to access FooCollection ... with token

Fixed ("t=TOKEN" is accepted in a same-origin URL, so it's really "public collections + secret-sharing-link")

- **The first form minimizes the cost and effort of deploying a wildcard TLS certificate for ...** which one is first form (<http://dl.example.com...?>). I think clarifying this will help

Reworded this:

```

-// The first form minimizes the cost and effort of deploying a
-// wildcard TLS certificate for *.dl.example.com. The second form is
-// likely to be easier to configure, and more efficient to run, on an
-// upstream proxy.
+// The first form (with "--" instead of ".") avoids the cost and
+// effort of deploying a wildcard TLS certificate for
+// *.collections.example.com at sites that already have a wildcard
+// certificate for *.example.com. The second form is likely to be
+// easier to configure, and more efficient to run, on a downstream
+// proxy.

```

- **keep-web ignores everything after the first "." or "--"** This is a bit confusing. Obviously, this needs to be sent in the URL to reach the host, correct (for ex: any load balancer etc)? I think this needs a bit explanation ...

Reworded:

```

-// In all of the above forms, the "dl.example.com" part can be
-// anything at all: keep-web ignores everything after the first "." or
-// "--".
+// In all of the above forms, the "collections.example.com" part can
+// be anything at all: keep-web itself ignores everything after the
+// first "." or "--". (Of course, in order for clients to connect at
+// all, DNS and any relevant proxies must be configured accordingly.)

```

- **"uuid\_or\_pdh" part can be either a collection UUID or a portable data hash with the "+" character replaced by "-"** .... In server\_test.go => Test200, third test spec is using "/c=" + arvadostest.FooPdh + "/t=" + arvadostest.ActiveToken + "/foo" without replacing "+" with "-". I also played with all tests by not doing this conversion and commenting out the logic in handler. All tests pass. So, it appears that a user can use pdh in url without converting + to - char. What is the reason behind this requirement? I ask because, it would be an extra step for all clients.

Indeed, I seem to have left that as a puzzle. Added rationale, and mentioned when it's mandatory and when it's optional:

```
// In all of the above forms, the "uuid_or_pdh" part can be either a
// collection UUID or a portable data hash with the "+" character
-// replaced by "-".
+// optionally replaced by "-". (Replacing "+" with "-" is mandatory
+// when "uuid_or_pdh" appears in the domain name only because "+" is
+// not a valid character in a domain name.)
```

- How would I download a file from a subdir in a collection? Let's say, I have a collection with ./subdir1/file1 and ./subdir2/file2, what would the URL to download file1? Would it be possible to add an example to doc (and may be a test)?

Updated comments in doc.go to use "/foo/bar.txt" as an example, and added some tests using the fixture with a "/dir1/foo" file.

- Since I do not know much about nginx configuration, I am hoping the section "example nginx configuration" is well reviewed by you :)

wow -- such confidants

Related: Unfortunately the nginx.conf in the test suite (in the workbench integration test branch) does things somewhat differently than the recommended production config. The way we assign port numbers on the fly in run\_test\_server.py makes a chicken-and-egg problem: we start keep-web first so we can tell nginx.conf the right port number for "upstream keep-web"; then we start nginx and find out its port number; then it's too late to tell keep-web what port number it should expect clients to be putting in their Host headers. So the test nginx.conf munges the Host header.

### Args to start server

- I think it would be useful to have all the supported / expected input arguments listed in one place in "starting the server" section. Right now, they are spread over the doc and hard to get to the bottom of it easily. I found -address, -attachment-only-host, -trust-all-content to be the list by reading through various bits.

Added "Run "keep-web -help" to show all supported options." to godoc. (Less convenient, but I'm not super keen on manually synchronizing two copies of the full list of options.)

- is the example for "-attachment-only-host correct? The following is what I am seeing and does not work with -address also:  
[...]
- Please include an example with all three (if one could use -trust-all-content and -attachment-only-host) together (which I think is acceptable)

-attachment-only-host doesn't set the -address, so your test was trying to listen on the default -address=:80. Added an explicit -address arg to the examples to make this less mysterious.

Added an example (and a sentence about what happens) with all three:

```
+// When using trust-all-content mode, the only effect of the
+// -attachment-only-host option is to add a "Content-Disposition:
+// attachment" header.
+//
+// keep-web -address :9999 -attachment-only-host domain.example:9999 -trust-all-content
```

### auth.go:

- "api\_token" for cookie name sounds too non-specific. Suggest "arvados\_api\_token" instead.

Indeed, fixed. (I was mimicking ?api\_token=FOO, but mimicking ARVADOS\_API\_TOKEN=FOO does seem better.)

- The comment "LoadTokensFromHttpRequestBody" seems out of place and makes it feel like it is about "loadTokenFromCookie" since it is right above this func. May be adding a small comment about loadTokenFromCookie might help distinguish the LoadTokensFromHttpRequestBody comment from the func following

True. Moved the TODO comment to the bottom of the file.

- services/keep-web/.gitignore has "keepdl" in it

Fixed

### main.go

- Why did you set ARVADOS\_API\_TOKEN = xxx in init? If an admin runs keep-web without setting this, do you intend it to not exit? If it is for test purposes, suggest moving it into an init in one of the tests.

The admin / startup script shouldn't set ARVADOS\_API\_TOKEN: keep-web only uses client-provided tokens and the anonymous token(s) provided.

Hmm. Evidently I never actually added the -anonymous-token flag, so there's no way to specify one. Should fix that.

#### server\_test.go -> Test404

- I spent a very long time to understand why the following works / not works, going back and forth in the test and the doc. I think adding some comments explaining why a path should 404 etc, and adding some more paths to the test would help.

Added/improved comments about the various reasons for 404:

```
- // Routing errors
+ // Routing errors (always 404 regardless of what's stored in Keep)
+ "/",
+ "/foo",
+ "/download",
+ "/collections",
+ "/collections/",
+ // Implicit/generated index is not implemented yet;
+ // until then, return 404.
+ "/collections/" + arvadosetest.FooCollection,
+ "/collections/" + arvadosetest.FooCollection + "/",
+ "/collections/" + arvadosetest.FooBarDirCollection + "/dir1",
+ "/collections/" + arvadosetest.FooBarDirCollection + "/dir1/",
```

- Add another path here: "/c=<foo-collection-uuid/t=<some-bad-token>/foo" — add comment that it will 404 because of bad token
- Add another path such as "/c=<foo-collection-uuid/t=<good-token>/thepertthcountyconspiracy

Bad token is 403 -- added TestVhostRedirectQueryTokenToBogusCookie (with ?api\_token=bogus) and TestSingleOriginSecretLinkBadToken (with /t=bogus/)

- Add "/collections/" + arvadosetest.HelloWorldCollection, and explain that even though publicly accessible, 404 because no file is specified
  - you might also want to consider adding this to the doc that entire collection cannot be downloaded, only individual files. Am I correct?

Added:

```
+// Indexes
+//
+// Currently, keep-web does not generate HTML index listings, nor does
+// it serve a default file like "index.html" when a directory is
+// requested. These features are likely to be added in future
+// versions. Until then, keep-web responds with 404 if a directory
+// name (or any path ending with "/" ) is requested.
+//
```

- Seems like adding another test with a bad anonymous token and accessing a public collection fails would be helpful

Added.

#### server\_test.go -> Test200

- I think it would be nice to add one more spec with **host: arvadosetest.FooCollection + ".dl.example.com"** right after the first spec with **host: arvadosetest.FooCollection + "--dl.example.com"**. Even though something similar is added later on in the list, having these two together might help see it right away

Added.

#### handler.go

- defer clientPool.Put(arv) : Would this fail if arv is nil? (Tests panicked when I used nil, btw)

Yes, but there's an "if arv == nil { ... return }" right above it, so this is impossible... right?

- Is Expires needed when setting the cookie? Since setting the expiry makes the cookie persistent and I am thinking we can probably do with session cookies? This may help with users that do not allow persistent cookies.

I'm not sure what the right answer is here. Persistent cookies still work when you close & reopen your browser, which seems like a good feature.

OTOH, if browsers/plugins/users are in the habit of ignoring persistent cookies, this wouldn't work very well for them. Is that a thing? (Shouldn't a privacy-sensitive browser just automatically downgrade persistent cookies to session cookies, rather than ignoring them completely, so things like this continue to work?)

### handler\_test.go

- Why call it mustParseURL instead of parseURL?

Following regexp.MustCompile's lead: it panics instead of returning an error if anything goes wrong.

- I personally think all "authz" can be just "auth". It just reads / flows easily when reading and equally descriptive ...

Following web server tradition of using "authz" to disambiguate authentication/authorization... I don't feel strongly about it.

- It appears that we are not checking the resp content in case of tests using testVhostRedirectTokenToCookie (sorry, if I am wrong). I think it would be great to verify the body has the expected content at least in cases such as attachment only etc.

Ah yes, we were checking it only after redirect+success. Fixed to check in "success without redirect" and "error" cases as well (e.g., nice to check that we don't send the forbidden data along with an HTTP 401...)

- There are some golint complaints in keep-web dir and the new fixtures.go

Fixed a bunch of golint, govet, gofmt.

### keep-dl versus keep-web

- It appears that "dl" is used in urls and test uids etc. It might help to clarify that we used "dl" to mean "download" in our default usages such as "\*\*.dl.example.com" etc

Indeed. Fixed godoc, tests, and install doc to to say "collections".

- Another thought. A naive user might wonder: "is keep-web a web interface to keep? If so, what is the keepproxy for ..." I think we need to clarify the distinction. As we discussed, may be we need a wiki page that lists and briefly describes all supported services etc with links to individual pages ...

Well, I think keep-web *is* a web interface to Keep. I did try to disambiguate by updating the keepproxy description, but I'm sure there's still lots of room for improvement.

A "how all the pieces fit together, and why/when each is used" diagram ~~would~~ will be great.

- ErrNoManifest message "Collection has no manifest" seems a bit misleading; how about "collecion manifest not provided / available"?

I'm not sure that's any better. We don't really know whether one is "available", and "provided" depends on context... We just know that the object passed to this function has no manifest. Can we rely on the caller to wrap this with some appropriate context if needed?

- I think adding some new lines / extra white space help improve readability of the very complex code in CollectionFileReader, cfReader and Read func. You may also consider adding some more comments in Read

Added some comments and whitespace, and split the go func(){ ... }() out into a named method. (Helpful?)

- In collectionreader\_test, this looks a bit complicated. Can it be as follows?  
[...]

Ah yes, fixed.

- Would be nice to add some extra white space and comments in the new code manifest.go as well

Any particularly mysterious bits in mind?

- golint suggestions in both packages (and also need to fmt collectionreader\_test)

Bunch of golint/vet/fmt complaints fixed.

→ [6e76e33](#)

## #21 - 10/30/2015 06:25 PM - Tom Clegg

keep-web at [7bf5f28](#) with -anonymous-token flag.

keep-web-workbench at [ce2bf31](#), using -anonymous-token flag in run\_test\_server.py (but not making use of it in Workbench integration tests)

## #22 - 11/02/2015 02:15 PM - Radhika Chippada

Just a couple more comments after reviewing updates up to [7bf5f28](#)

- Question about "Replacing "+" with "-" is mandatory when "uuid\_or\_pdh" appears in the domain name only because "+" is not a valid character in a domain name" — So, the clients might be configuring one or more collections (using pdh or uuid) explicitly than using "\*.collections ..." etc?
- Yes, but there's an "if arv == nil { ... return }" right above it, so this is impossible... right? — This was more an observation about clientPool not doing a null checking on the input param ... You can ignore it.
- As far as I know a persistent cookie will not be stored as a session cookie if the user does not accept. It does not seem like we are using persistent cookies elsewhere. So it would be nice if we do not use them when we can do with session cookies, where we do not need permission from user.
- anonymous.go — "API token to try when none of the tokens provided in an HTTP request succeed in reading the desired collection. If this flag is used more than once, each token will be attempted in turn until one works" — Can you add "you can provide multiple anonymous tokens by using the flag more than once. If this flag is used more than once, ..."
- The rest looks good or no big concerns. I haven't yet looked at the workbench end of the updates.

## #23 - 11/03/2015 03:08 PM - Tom Clegg

Radhika Chippada wrote:

- Question about "Replacing "+" with "-" is mandatory when "uuid\_or\_pdh" appears in the domain name [...]"

Changed to:

(When "uuid\_or\_pdh" appears in the domain name, replacing "+" with "-" is mandatory, because "+" is not a valid character in a domain name.)

- As far as I know a persistent cookie will not be stored as a session cookie if the user does not accept. It does not seem like we are using persistent cookies elsewhere. So it would be nice if we do not use them when we can do with session cookies, where we do not need permission from user.

Good point, Workbench seems to have gotten this far on session cookies. Updated.

- anonymous.go — "API token to try when [...]"

Changed to:

API token to try when none of the tokens provided in an HTTP request succeed in reading the desired collection. Multiple anonymous tokens can be provided by using this flag more than once; each token will be attempted in turn until one works.

## #24 - 11/09/2015 08:22 PM - Tom Clegg

Radhika Chippada wrote:

- First of, install guide says "API config" instead of "workbench config"
- Install guide says "keep-web" where as workbench uses "keep\_web\_url" as the parameter name

(already fixed in other branch)

- In "Configure DNS" section, the Install guide says that when the third form "dl.uuid\_prefix.your.domain" is used, only public data is served. And, the third form is listed in "Tell the API server ..." However, it seems the workbench is using the given token if one is available, which means non-public data also is server? Should the `if Rails.configuration.keep_web_url` block in `collections_controller.rb` -> `show_file` check the format of the configuration to determine whether or not to use token? Also, the example in "application.default.yml" does not list the third form as example.

We resort to query\_token only if that's the only way to read the collection. At this point, the only possibilities are

- keep\_web\_url is a multi-origin form, like "%{uuid\_or\_pdh}.collections.example" (so ?api\_token=X will work)
- keep\_web\_url is a single-origin form, like "collections.example/c=%{uuid\_or\_pdh}" and keep-web is running with -attachment-only-host (so ?api\_token=X will work)
- keep\_web\_url is a single-origin form, but keep-web is *not* running with -attachment-only-host (so ?api\_token=X will not work)

The "will not work" case amounts to a configuration error.

I think we can cover all the cases that are technically possible if we add one more config:

- Single-origin URL, need to pass token in query → this works if the URL is recognized as -attachment-only-host (but of course that means "view inline" won't happen, so the "preview" button will always download instead of previewing)
- Single-origin URL, do not need to pass token in query (i.e., anonymous/reader token) → this works for both preview and download, if the URL is **not** recognized as -attachment-only-host

```
keep_web_url: https://collections.uuid_prefix.arvadosapi.com/c=%{uuid_or_pdh}
keep_web_attachment_only_url: https://dl.uuid_prefix.arvadosapi.com/c=%{uuid_or_pdh}
```

- collections\_controller\_test.rb -> I think it would be nice if we also do Rails.configuration.anonymous\_user\_token in "test No redirect to keep\_web\_url if collection not found" to verify that it fails even when anonymous token is configured

Added.

- download\_test.rb
  - Can the test be improved to use "id\_type" to cover both the uuid and pdh cases as in collection\_controller\_test?

Added.

- The test did not pass for me (I get 502 Bad Gateway error); but I only ran this one test.
- Several collection related workbench tests failed; I did not rerun yet.

I'm hoping that means you didn't have the 5824-keep-web-workbench branch on arvados-dev. (Without that, you won't have a keep-web running.)

#### #25 - 11/10/2015 06:37 AM - Tom Clegg

Tom Clegg wrote:

I think we can cover all the cases that are technically possible if we add one more config:

- Single-origin URL, need to pass token in query → this works if the URL is recognized as -attachment-only-host (but of course that means "view inline" won't happen, so the "preview" button will always download instead of previewing)
- Single-origin URL, do not need to pass token in query (i.e., anonymous/reader token) → this works for both preview and download, if the URL is **not** recognized as -attachment-only-host

Addressed this in [4d0133c](#) by adding a separate keep\_web\_download\_url config.

- If keep\_web\_download\_url is set, disposition=attachment links (i.e., downloads) are routed to the given URL (typically https://download.uuid\_prefix.arvadosapi.com/c=%{uuid\_or\_pdh}). This is expected to match keep-web's -attachment-only-host argument.
- If keep\_web\_download\_url is set and keep\_web\_url is *not* set, preview is disabled and all preview/download links are routed to keep\_web\_download\_url.
- If both are set, but keep\_web\_url isn't a wildcard (one-vhost-per-collection) form, showing inline content from a non-public non-shareable-link collection will result in a download instead of preview.
- If keep\_web\_download\_url is not set, and keep\_web\_url isn't a wildcard form, showing inline content from a non-public non-shareable-link collection will produce an error: with that configuration, we can't authenticate safely without creating XSS problems.

#### #26 - 11/10/2015 11:15 PM - Radhika Chippada

- Install guide -> Tell Workbench about the keep-web service
  - It probably would help a little to add a sentence saying, configure the following two (as needed) to download and preview
  - Alternatively, "Add the following entry to your Workbench configuration file" — can you add a bit more explanation such as "to be used as the link url for downloading ...". Same about the next one.
- Workbench application.default.yml file
  - The description for keep\_web\_url says: ". . . Workbench's built-in download mechanism". Shouldn't it say "Workbench's built-in PREVIEW mechanism"?
  - The description for keep\_web\_download\_url says: "The host here should match the -attachment-only-host argument given to keep-web" -> The host here should be the "same value" as the argument given to keep-web, correct?
- In collection\_controller.rb -> keep\_web\_url method -> the variable "test\_uri" probably can be renamed as "check\_uri" or "verify\_uri". The word "test" seems to mean something related to "testing". Just an observation and you can ignore if you don't like it
- In collection\_controller.rb -> show\_file method
  - Can you add two block level comments at line #140 (where we are first checking keep\_web config) and the second for the block at line 156 (where no keep\_web config found and hence defaulting to the previous implementation)?
  - Also, question about the following. Let's say the file does not exist and we used the first flow (keep\_web config exists). I am assuming I

would see 404. Do we instead want to make the following check before redirecting to keep\_web url?

```
file_name = params[:file].andand.sub(/^(\\.\\|\\/|\\/)/, './')
if file_name.nil? or not coll.manifest.has_file?(file_name)
  return render_not_found
end
```

- All workbench and other tests passed. However, two tests in keep-rsync are failing.

#### #27 - 11/11/2015 01:56 AM - Tom Clegg

Radhika Chippada wrote:

- Install guide -> Tell Workbench about the keep-web service
  - It probably would help a little to add a sentence saying, configure the following two (as needed) to download and preview
  - Alternatively, "Add the following entry to your Workbench configuration file" — can you add a bit more explanation such as "to be used as the link url for downloading ...". Same about the next one.

Did both: introductory paragraph, plus stuck "This URL will be used for [...]" on the two config sections.

Also updated the "SSL certificate is required" bit, since it's not strictly required except for one specific category.

- The description for keep\_web\_url says: "... Workbench's built-in download mechanism". Shouldn't it say "Workbench's built-in PREVIEW mechanism"?

Yes. (Changed to "download/preview" since it does both.)

- The description for keep\_web\_download\_url says: "The host here should match the -attachment-only-host argument given to keep-web" -> The host here should be the "same value" as the argument given to keep-web, correct?

Tried a few different wordings and then decided an example would be helpful:

```
+ # The host part of the keep_web_download_url value here must match
+ # the -attachment-only-host argument given to keep-web: if
+ # keep_web_download_url is "https://FOO.EXAMPLE/c=..." then keep-web
+ # must run with "-attachment-only-host=FOO.EXAMPLE".
```

- In collection\_controller.rb -> keep\_web\_url method -> the variable "test\_uri" probably can be renamed as "check\_uri" or "verify\_uri". The word "test" seems to mean something related to "testing". Just an observation and you can ignore if you don't like it

Agreed, changed to check\_uri

- Can you add two block level comments at line #140 (where we are first checking keep\_web config) and the second for the block at line 156 (where no keep\_web config found and hence defaulting to the previous implementation?)

Added

- Also, question about the following. Let's say the file does not exist and we used the first flow (keep\_web config exists). I am assuming I would see 404. Do we instead want to make the following check before redirecting to keep\_web url?

We could, but keep-web will have to re-check that anyway, so I think it's better to do as little as possible before redirecting. (I'm hoping we'll soon change the links in Workbench so they go directly to keep-web instead of going through this handler...)

- All workbench and other tests passed. However, two tests in keep-rsync are failing.

Hm. keep-rsync is passing for me. Which tests? What version of Go? (I have 1.5.1)

#### #28 - 11/11/2015 04:26 PM - Radhika Chippada

All those updates look good. Also, I reran the whole gamut again and keep-rsync tests are passing now. LGTM

#### #29 - 11/12/2015 04:21 PM - Radhika Chippada

Just a couple comments for the latest at [95a589bcfe740fed72163f3e76e516b8bba5b0cd](https://github.com/95a589bcfe740fed72163f3e76e516b8bba5b0cd)

- collections\_controller.rb -> do we still need this in this file? -> require "cgi"

- download\_helper.rb -> why did you use rm\_f initially? That is, would you rather use rm\_r with force true or rm\_rf instead of rm\_r in this updated version?
- With this latest update to \_show\_files.html, I am thinking we will see links instead of images in file rows. I am adding this collection here so that I can go back and verify after the code is in staging.

<https://workbench.4xphq.arvadosapi.com/collections/4xphq-4zz18-mb3uhfwvr32tmh0>

### #30 - 11/12/2015 05:08 PM - Tom Clegg

Radhika Chippada wrote:

Just a couple comments for the latest at [95a589bcfe740fed72163f3e76e516b8bba5b0cd](https://workbench.4xphq.arvadosapi.com/collections/95a589bcfe740fed72163f3e76e516b8bba5b0cd)

- collections\_controller.rb -> do we still need this in this file? -> require "cgi"

Indeed, no. Removed.

- download\_helper.rb -> why did you use rm\_f initially? That is, would you rather use rm\_r with force true or rm\_rf instead of rm\_r in this updated version?

I think I started with "rm\_rf" and then removed the "r" when I meant to remove the "f". I don't expect anything to need forcing -- but it definitely needs to be a recursive delete. I'm a little surprised it didn't throw an error when it failed to delete anything; I only noticed because I looked at my download dir and found a bunch of "foo (1)" and so on. The tests were checking some old "foo" file that had been saved long ago... :/

- With this latest update to \_show\_files.html, I am thinking we will see links instead of images in file rows. I am adding this collection here so that I can go back and verify after the code is in staging.

In my manual testing the browser ignored content-disposition and showed the image inline as desired. But I've noticed this change isn't really necessary, since CollectionsController#keep\_web\_url already redirects to a disposition=attachment link when needed, so I've backed it out. [#7743](#) will obsolete whatever we do here anyway.

### #31 - 11/12/2015 08:05 PM - Tom Clegg

- Status changed from *In Progress* to *Resolved*

Applied in changeset arvados|commit:209d83a0ce0bd22669a39f49daddca432babeae7.