Arvados - Story #10849

[API] [SDKs] When client and server versions differ, the server must either implement the API the client expects, or return an error.

01/09/2017 11:10 PM - Tom Clegg

Status:	New	**Start date:**	01/09/2017
Priority:	Normal	**Due date:**	
Assigned To:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:	Arvados Future Sprints		

Description

Background

After an upgrade, an API server will receive some requests that were intended for an older version.

- Some nodes have older SDK code installed, because the API server gets upgraded first.
- Some nodes have older SDK code installed, because they are not covered by the cluster upgrade procedure (e.g., users' workstations).
- Even on nodes with the most current SDK, some processes are still using the old SDK code and discovery document, because they started before the API server was upgraded.

It is also possible for a new client to connect to an old API server (e.g., arv-copy running on a new cluster), which can produce unpredictable results and obscure error messages.

Currently, the discovery document provides an API "revision" which clients can check *before* making API calls. However:

- Our testing framework does not remind/force us to update this API version when the API changes (it has changed only twice since being introduced as "20121015", and is now "20131114").
- SDKs do not check it.
- Even if SDKs checked it, an API-side check would still be necessary to cover the cases where the API server is upgraded between discovery document retrieval and API request.

Requirements

An application that tells the Arvados SDK an expected API version should fail with an "API version not supported" message *as soon as possible* (e.g., as soon as the discovery document arrives) if the server does not offer a compatible API.

When the API server receives a request that was intended for a different API version than the current version (for example, the client is still using a discovery document that was fetched before the last API upgrade) the API server must either implement the old API or reject the request with an "unsupported API version" error.

Implementation

In the discovery document, under the "apiVersions" key, the API server should provide a list of supported API versions, sorted with preferred (more recent) versions first. When a schema or API change breaks support for an older version, the old version should be dropped from the list.

Use semantic versions, like "0.1.20170110".

Before making an API request, a client should check the list of supported versions (in the discovery doc) and choose which version to use. Depending on the client, this could be the latest advertised version, or a specific version the client was developed for.

SDKs should allow the caller to specify an API version, and choose a compatible version (according to semver) from the list advertised in the discovery document. For example, if the application asks the SDK for "1.2.3" and the server offers ["1.2.0", "1.4.5", "2.0.0"] then the SDK should use "1.4.5". (Note: Current version numbers are "0.x.y", so "compatible" implies "identical.")

When making an API request, a client should provide the expected API version in a request header "x-arvados-api-version" or (if the client has trouble adding http headers) a form value or query parameter "api_version".

When receiving an API request, the API server should check the client-supplied version string against the list of supported versions, and respond 4xx if the requested version is not supported. When the major version reaches 1, this check will need to compare versions using semver, and the list of supported versions will be short; until then, only exact matches are compatible, and the list of supported versions will be longer.

The existing "revision" field in the discovery document should be dropped.

Future work

When forwarding an API request, the API gateway should check the requested API version and forward the request to an appropriate API server or version-translating proxy server.

Build/test tooling should try to detect (and prevent) API changes that slip through without updating the API version.

Related issues:

Related to Arvados - Story #3812: [SDKs] Clients should warn when they're exp...	**Closed**	**09/04/2014**
Copied from Arvados - Story #3362: [SDKs] Notify user if client library versi...	**Closed**	
Copied from Arvados - Story #4308: [API] [SDKs] Enable API versioning	**Closed**	

History

#1 - 01/10/2017 03:41 PM - Tom Clegg

- Description updated

#2 - 01/10/2017 03:44 PM - Tom Clegg

- Description updated

#3 - 01/10/2017 03:54 PM - Tom Clegg

- Description updated

#4 - 01/10/2017 08:12 PM - Tom Clegg

- Subject changed from [API] [SDKs] When client and server versions differ, the server must either implement the old API, or return an error. to [API] [SDKs] When client and server versions differ, the server must either implement the API the client expects, or return an error.

- Description updated

#5 - 01/11/2017 04:27 PM - Tom Clegg

- Description updated

#6 - 08/28/2017 09:21 PM - Tom Morris

- Target version set to Arvados Future Sprints