

Redmine - Feature #5997

REST API should provide valid xml response with status for EVERY request

2010-07-30 01:12 - Alex Last

Status:	Closed	Start date:	2010-07-30
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	REST API	Estimated time:	0.00 hour
Target version:			
Resolution:	Wont fix		

Description

Redmine's REST API provides valid XML response only if all request parameters are valid and accepted. So for this URL

```
http://demo.redmine.org/projects/ace/issues.xml?query_id=302&key=1ddd3b8a5f4484bc658727586681f15189b42ac7
```

a valid XML with list of issues will be returned, but if I change query_id to some random value:

```
http://demo.redmine.org/projects/ace/issues.xml?query_id=6666&key=1ddd3b8a5f4484bc658727586681f15189b42ac7
```

- then response shown in Google chrome browser is:

```
Oops! This link appears to be broken.
```

and in Java program:

```
java.lang.RuntimeException: java.io.FileNotFoundException: http://demo.redmine.org/projects/ace/issues.xml?query_id=6666&key=1ddd3b8a5f4484bc658727586681f15189b42ac7
    at org.alskor.mspsync.connector.redmine.RedmineRESTAPIConnector.loadData (RedmineRESTAPIConnector.java:77)
    at org.alskor.mspsync.ui.editors.ConnectorPanel$2.run (ConnectorPanel.java:128)
    at org.eclipse.core.internal.jobs.Worker.run (Worker.java:54)
Caused by: java.io.FileNotFoundException: http://demo.redmine.org/projects/ace/issues.xml?query_id=6666&key=1ddd3b8a5f4484bc658727586681f15189b42ac7
    at sun.net.www.protocol.http.HTTPURLConnection.getInputStream (HTTPURLConnection.java:1311)
    at sun.net.www.protocol.https.HttpsURLConnectionImpl.getInputStream (HttpsURLConnectionImpl.java:234)
```

I propose to change xml returned by Redmine REST API for ALL incoming requests to something like this:

```
<redmine_rest_api_response responseCode="0" responseDescription="...">
  <issues>.....
</issues>
</redmine_rest_api_response>
```

the new wrapping "redmine_rest_api_response" header node (or whatever you want to call it) should provide

1. status code (0=OK, 1=authentication failure, 2=parameter invalid, 3=access key valid, but access denied to the specified resource - e.g. query ID belonging to other user, ...)
2. response description - some human-readable text like "authentication failed: key 123 is not recognized" or "value '6666' or parameter 'query_id' is not allowed for the given API access key because..."

this would be really useful for building external applications on top of Redmine REST API.

History

#1 - 2010-08-02 02:11 - Gerrit Kaiser

The "status code" you're proposing is already provided by HTTP. To detect non-existing queries for instance, you could check for HTTP Status 404.

#2 - 2010-08-06 02:06 - Eric Davis

- Status changed from New to Closed

- Resolution set to Wont fix

When using the rest API you need to check the HTTP status code first. Making every request return a 200 response and embedding the status code into the XML body is just duplicating the HTTP standards.

In your Java program, you should be able to catch the 400 status codes and handle it in code (e.g. retry, show message to user). Same for the server error codes (500).

#3 - 2010-08-06 02:13 - Alex Last

Agreed, many of these situations can be handled by HTTP status reports, but I'm not sure if the REST API provides meaningful error messages in addition to the HTTP response code (like "parameter XX can be ... but you provided", "project with id ... can't be found" or whatever).

#4 - 2010-08-06 18:41 - Eric Davis

Some responses will. For example if you try to create an issue but are missing fields, an array of errors should be returned.

#5 - 2010-08-06 18:44 - Alex Last

good, thanks. one of the reasons why I submitted this request is because I was frustrated with creating issues via the REST API (I posted a question here: <http://www.redmine.org/boards/2/topics/16260>).

The error message returned by Redmine is:

```
Resp Code:405
Resp Message:Method Not Allowed
```

which is very frustrating - no idea what to do with it.

#6 - 2010-08-07 00:54 - Eric Davis

Alexey Skor wrote:

which is very frustrating - no idea what to do with it.

Anytime you get a 400 or 500 series HTTP status code you need to look up what they mean. In this case (405), you are using the wrong HTTP verb, e.g. GET instead of POST.

Reference: http://en.wikipedia.org/wiki/List_of_HTTP_status_codes#4xx_Client_Error

#7 - 2010-08-07 02:01 - Alex Last

I understand that, but I'm using "POST /issues.xml" as Redmine REST API webpage describes:

http://www.redmine.org/wiki/redmine/Rest_Issues

```
Creating an issue
POST /issues.xml
```

(see the code provided in that forum topic) - and Redmine complains that only GET is allowed.

so I suspect that I'm not building the request right (almost sure about this) and I should get a real explanation from Redmine instead of "only get is allowed".

#8 - 2010-08-07 13:49 - Felix Schäfer

Alexey Skor wrote:

I should get a real explanation from Redmine

Everything more than "go read the docs" would require envisioning a lot of possible wrong requests you could send it and the corresponding fix, so: no. You can read in your development logs (you do develop your application against a redmine in development mode, right?) at which point redmine borks. In that case: POST /issues.xml, where's the API key needed to authenticate you? (unless anonymous can create issues, of course).

#9 - 2010-08-07 13:50 - Felix Schäfer

Felix Schäfer wrote:

In that case: POST /issues.xml, where's the API key needed to authenticate you? (unless anonymous can create issues, of course).

Ok, forget about that part, just noticed you did call them with the key.

#10 - 2010-08-10 02:35 - Alex Last

I'm still stuck with this "POST" request for issues creation. Does anyone has an example of valid HTTP POST request for this? I don't need Java code, just a request body itself is fine.

#11 - 2010-08-13 01:37 - Eric Davis

Alexey Skor wrote:

I'm still stuck with this "POST" request for issues creation. Does anyone has an example of valid HTTP POST request for this? I don't need Java code, just a request body itself is fine.

One with username and password, one with an API key:

```
$ curl -X POST -H "Content-Type: text/xml" -d "<issue><subject>api</subject><project_id>weimann3</project_id></issue>" http://admin:password@redmine.acheron/issues.xml
<?xml version="1.0" encoding="UTF-8"?>
<issue>
  <id>226</id>
  <project name="Customer-focused analyzing inf" id="4"/>
  <tracker name="Bug" id="1"/>
  <status name="New" id="1"/>
  <priority name="Normal" id="4"/>
  <author name="Eric Davis" id="1"/>
  <subject>api</subject>
  <description></description>
  <start_date>2010-08-12</start_date>
  <due_date></due_date>
  <done_ratio>0</done_ratio>
  <estimated_hours></estimated_hours>
  <spent_hours>0.0</spent_hours>
  <created_on>Thu Aug 12 23:37:30 +0000 2010</created_on>
  <updated_on>Thu Aug 12 23:37:30 +0000 2010</updated_on>
  <relations>
</relations>
</issue>
```

```
$ curl -X POST -H "Content-Type: text/xml" -d "<issue><subject>api</subject><project_id>weimann3</project_id></issue>" http://8e8dc0e6a24a95b228a3c046b9e7fe4be3411412:X@redmine.acheron/issues.xml
<?xml version="1.0" encoding="UTF-8"?>
<issue>
  <id>227</id>
  <project name="Customer-focused analyzing inf" id="4"/>
  <tracker name="Bug" id="1"/>
  <status name="New" id="1"/>
  <priority name="Normal" id="4"/>
  <author name="Eric Davis" id="1"/>
  <subject>api</subject>
  <description></description>
  <start_date>2010-08-12</start_date>
  <due_date></due_date>
  <done_ratio>0</done_ratio>
  <estimated_hours></estimated_hours>
  <spent_hours>0.0</spent_hours>
  <created_on>Thu Aug 12 23:38:50 +0000 2010</created_on>
  <updated_on>Thu Aug 12 23:38:50 +0000 2010</updated_on>
  <relations>
</relations>
</issue>
```

#12 - 2010-08-13 06:50 - Alex Last

thanks a lot.

a question: in this sample URL

<http://8e8dc0e6a24a95b228a3c046b9e7fe4be3411412:X@redmine.acheron/issues.xml>

- does ":X@" stand for some value I need to put or it must be exactly ":X@" in the url (which didn't work for me)?

btw, this is not very secure to send api key in the URL. moving it inside the request body would be much better if HTTPS connection is used.

#13 - 2010-08-13 11:32 - Felix Schäfer

Alexey Skor wrote:

- does ":X@" stand for some value I need to put or it must be exactly ":X@" in the url (which didn't work for me)?

It can be whatever value you want, IIRC you can give the key as username and anything as password.

btw, this is not very secure to send api key in the URL. moving it inside the request body would be much better if HTTPS connection is used.

No, SSL/TLS encryption works at a lower level in the OSI model, i.e. the whole HTTP part of an https connection is completely encrypted.

#14 - 2010-08-17 02:46 - Alex Last

tried

```
curl -X POST -H "Content-Type: text/xml" -d "<issue><subject>api</subject><project_id>taskconnector-test</project_id></issue>"  
https://fc20e8f5f1885fc194b7b7865e6d6ce9a44a4b88:X@www.hostedredmine.com/issues.xml
```

- does not work for HTTPS connection (no response is shown for CURL, similar request from a Java program shows "401 not authorized").

at the same time similar request works fine with my local Redmine through HTTP.

is this some server configuration issue on www.hostedredmine.org or the request must be different in case of HTTPS protocol?

#15 - 2010-08-25 11:08 - Brad Rushworth

Alexey Skor wrote:

is this some server configuration issue on www.hostedredmine.org or the request must be different in case of HTTPS protocol?

See: <http://www.redmine.org/boards/2/topics/16260#message-16787>