Infrastructure - Bug #8122

Metacat is double-decoding incoming urls on the CNs

2017-06-30 23:55 - Rob Nahf

Status:	Closed	**Start date:**	2017-06-30
Priority:	Normal	**Due date:**	
Assignee:	Jing Tao	**% Done:**	100%
Category:	dataone-cn-metacat	**Estimated time:**	0.00 hour
Target version:	CCI-2.3.5		
Milestone:	None	**Story Points:**	
Product Version:	*		

Description

v1 and v2 CN and MN ResourceHandler.handle method is decoding the request pathInfo for most request types, leading to potentially mangled identifiers. Since apache decodes the percent-escaping, metacat should not decode again (double-decoding isn't safe).

If there are certain calls where decoding is needed, note that the URLEncoder D1ResourceHandler is using decodes '+' to space characters, which will mangle identifiers containing spaces. Use the safer org.dataone.service.util.EncodingUtilities.decodeString(...) from d1_common_java which was designed to get around this problem.

A good test is to call getSystemMetadata on this tricky identifier from a browser:
https://cn.dataone.org/cn/v2/meta/anIdentifierContainingAPlus%2BNotASpace

If the returned NotFound error has a space in the identifier instead of a '+' - it's still broken.

Related issues:

Blocks Infrastructure - Task #8121: Fix MIME type and formatId for KML files	**Closed**	**2017-06-30**

History

#1 - 2017-07-01 11:07 - Dave Vieglais

- Blocks Task #8121: Fix MIME type and formatId for KML files added

#2 - 2017-07-03 21:18 - Jing Tao

The url on the MN works fine:
https://mn-sandbox-ucsb-1.test.dataone.org/knb/d1/mn/v2/meta/identifier%2Bnospace

#3 - 2017-07-03 23:04 - Matthew Jones

Interesting. So, maybe Metacat assumes it is handling the decoding. Is the CN rest servlet decoding it, then making another http call to Metacat with an already decoded URI?

#4 - 2017-07-03 23:20 - Jing Tao

I guess so.

#5 - 2017-07-03 23:22 - Jing Tao

Using the method decodeString in the org.dataone.service.util.EncodingUtilities class seems working. Need more test.
The difference is this method replaces the "+" by "%2B" at the original string.

#6 - 2017-07-05 22:40 - Jing Tao

- % Done changed from 0 to 30

- Status changed from New to In Progress

I used the org.dataone.service.util.EncodingUtilities to decode the url and it works:

https://cn-dev.test.dataone.org/cn/v2/meta/myidentifier%2Bnospaces1

Changes was made to trunk (2.9.0) and 2.8 branch.

#7 - 2017-07-05 23:32 - Matthew Jones

And Jing, did you verify that the old Metacat API also still works correctly? Do all metacat and DataONE tests pass?

#8 - 2017-07-06 19:05 - Rob Nahf

cn_rest is no longer involved in proxying to metacat, it is done directly from apache2, via metacat_proxy.conf (found in dataone-cn-metacat).

Apparently, apache is allowed to decode paths before parsing rewrite rules, so that's probably where the decoding is happening. (https://serverfault.com/a/683109)

Note that the serverfault poster recommends what we were doing before - having a front-controller that handles all decoding. oh well...

Here's a thorough mod_rewrite summary, again, on serverfault.
https://serverfault.com/questions/214512/redirect-change-urls-or-redirect-http-to-https-in-apache-everything-you-ever

#9 - 2017-07-06 19:27 - Jing Tao

Matt: I haven't done thorough tests yet. The basic test works. But this only involves with DataONE API and doesn't relate to the old Metacat API.

#10 - 2017-07-06 19:40 - Jing Tao

Rob:
I read online and found:
Normally, mod_proxy will canonicalise ProxyPassed URLs. But this may be incompatible with some backends, particularly those that make use of PATH_INFO. The optional nocanon keyword suppresses this and passes the URL path "raw" to the backend. Note that this keyword may affect the security of your backend, as it removes the normal limited protection against URL-based attacks provided by the proxy
on this link:
https://serverfault.com/questions/715242/encode-url-wihthin-url-apache-mod-proxy-proxypass

So in the metacat_proxy file which locate on dataone-cn-metacat, if I change the code to (adding the key word "nocanon")
ProxyPassMatch "^/(cn/v[12]/(?:meta|formats|object|views)/?)$" "ajp://localhost:8009/metacat/d1/$1" nocanon
ProxyPassMatch "^/(cn/v[12]/(?:checksum|formats|isAuthorized|meta|object|replicaAuthorizations|views)/.+)$" "ajp://localhost:8009/metacat/d1/$1" nocanon

The cn proxy will not decode the uri. Here is the url and catalina.out (with the key word "nocanon"):

https://cn-dev.test.dataone.org/cn/v2/meta/myidentifier%2Bnospaces1
In D1URLFilter.
HTTP Verb: GET
original pathInfo: /meta/myidentifier+nospaces1
original requestURI: /metacat/d1/cn/v2/meta/myidentifier%2Bnospaces1
stripping /metacat/d1/cn/v2 from requestURI
new pathinfo: /meta/myidentifier%2Bnospaces1
After decoded: myidentifier+nospaces1

Without the key word:
In D1URLFilter.
HTTP Verb: GET
original pathInfo: /meta/myidentifier+nospaces1
original requestURI: /metacat/d1/cn/v2/meta/myidentifier+nospaces1
stripping /metacat/d1/cn/v2 from requestURI
new pathinfo: /meta/myidentifier+nospaces1
After decoded: myidentifier+nospaces1

So we can see with the key word, the proxy doesn't decode the "%2B" to "+" in the original request URI.

#11 - 2017-07-11 18:05 - Jing Tao

- Assignee set to Jing Tao

- Target version set to CCI-2.3.5

#12 - 2017-07-11 18:15 - Jing Tao

- Category changed from Metacat to dataone-cn-metacat

#13 - 2017-07-13 22:20 - Jing Tao

- Status changed from In Progress to Closed

- % Done changed from 30 to 100

#14 - 2017-09-08 17:28 - Jing Tao

- % Done changed from 100 to 30

- Target version changed from CCI-2.3.5 to CCI-2.4.0

- Status changed from Closed to In Progress

Push back to Metacat 2.9.0. The solution to use the EncodingUtilities messed up the solr query.

#15 - 2017-09-08 17:40 - Jing Tao

- % Done changed from 30 to 100

- Target version changed from CCI-2.4.0 to CCI-2.3.5

- Status changed from In Progress to Closed

My mistake. I read the comments and found the change is on dataone-cn-metacat rather Metacat itself. So it still works.