

Pulp - Issue #8791

Usability of "fields" when querying rpm content types

05/20/2021 06:48 PM - wibbit

Status: NEW	Start date:
Priority: Normal	Due date:
Assignee:	Estimated time: 0:00 hour
Category:	
Sprint/Milestone:	
Severity: 2. Medium	Groomed: No
Version:	Sprint Candidate: No
Platform Release:	Tags:
OS:	Sprint:
Triaged: Yes	Quarter:

Description

While working with Different RPM content types, and listing them, I hit the below error.

```
In [24]: rpm_content_package_categories.list(limit='500', repository_version='/pulp/api/v3/repositories/rpm/rpm/954de407-6872-4e62-b3d8-6019d30809aa/versions/1/', fields='pulp_href')
```

```
[11/15244]
```

```
-----  
ValueError                                Traceback (most recent call last)  
<ipython-input-24-2e19224fcdff> in <module>  
----> 1 rpm_content_package_categories.list(limit='500', repository_version='/pulp/api/v3/repositories/rpm/rpm/954de407-6872-4e62-b3d8-6019d30809aa/versions/1/', fields='pulp_href')
```

```
/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api/content_packagecategories_api.py in list(self, **kwargs)  
    68     """  
    69     kwargs['_return_http_data_only'] = True  
--> 70     return self.list_with_http_info(**kwargs) # noqa: E501  
    71  
    72     def list_with_http_info(self, **kwargs): # noqa: E501
```

```
/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api/content_packagecategories_api.py in list_with_http_info(self, **kwargs)  
    167     auth_settings = ['basicAuth', 'cookieAuth'] # noqa: E501  
    168  
--> 169     return self.api_client.call_api(  
    170         '/pulp/api/v3/content/rpm/packagecategories/', 'GET',  
    171         path_params,
```

```
/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api_client.py in call_api(self, resource_path, method, path_params, query_params, header_params, body, post_params, files, response_type, auth_settings, async_req, _return_http_data_only, collection_formats, _preload_content, _request_timeout, _host)  
    363     """  
    364     if not async_req:  
--> 365         return self.__call_api(resource_path, method,  
    366                                 path_params, query_params, header_params,  
    367                                 body, post_params, files,
```

```
/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api_client.py in __call_api(self, resource_path, method, path_params, query_params, header_params, body, post_params, files, response_type, auth_settings, _return_http_data_only, collection_formats, _preload_content, _request_timeout, _host)  
    207         # deserialize response data  
    208         if response_type:
```

```

--> 209         return_data = self.deserialize(response_data, response_type)
      210         else:
      211             return_data = None

/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api_client.py in deserialize(self,
response, response_type)
      279         data = response.data
      280
--> 281         return self.__deserialize(data, response_type)
      282
      283     def __deserialize(self, data, klass):

/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api_client.py in __deserialize(self
, data, klass)
      318         return self.__deserialize_datetime(data)
      319     else:
--> 320         return self.__deserialize_model(data, klass)
      321
      322     def call_api(self, resource_path, method,

/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api_client.py in __deserialize_mode
l(self, data, klass)
      657         if klass.attribute_map[attr] in data:
      658             value = data[klass.attribute_map[attr]]
--> 659             kwargs[attr] = self.__deserialize(value, attr_type)
      660
      661         instance = klass(**kwargs)

/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api_client.py in __deserialize(self
, data, klass)
      295         if klass.startswith('list'):
      296             sub_kls = re.match(r'list\[([.]*)\]', klass).group(1)
--> 297             return [self.__deserialize(sub_data, sub_kls)
      298                     for sub_data in data]
      299

/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api_client.py in <listcomp>(.)
      295         if klass.startswith('list'):
      296             sub_kls = re.match(r'list\[([.]*)\]', klass).group(1)
--> 297             return [self.__deserialize(sub_data, sub_kls)
      298                     for sub_data in data]
      299

/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api_client.py in __deserialize(self
, data, klass)
      318         return self.__deserialize_datetime(data)
      319     else:
--> 320         return self.__deserialize_model(data, klass)
      321
      322     def call_api(self, resource_path, method,

/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/api_client.py in __deserialize_mode
l(self, data, klass)
      659             kwargs[attr] = self.__deserialize(value, attr_type)
      660
--> 661         instance = klass(**kwargs)
      662
      663         if has_discriminator:

/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/models/rpm_package_category_respons
e.py in __init__(self, pulp_href, pulp_created, id, name, description, display_order, group_ids, d
esc_by_lang, name_by_lang, digest, local_vars_configuration)
      82         if pulp_created is not None:
      83             self.pulp_created = pulp_created
--> 84         self.id = id
      85         self.name = name
      86         self.description = description

```

```
/opt/bats/lib/python3.8/site-packages/pulpcore/client/pulp_rpm/models/rpm_package_category_respons
e.py in id(self, id)
    156         """
    157         if self.local_vars_configuration.client_side_validation and id is None: # noqa: E
501
--> 158             raise ValueError("Invalid value for `id`, must not be `None`") # noqa: E501
    159
    160         self._id = id
```

```
ValueError: Invalid value for `id`, must not be `None`
```

This appears to be due to the fact that numerous content types have "mandatory" fields, and if specifying fields = 'xyz' when performing a list, and a mandatory field is not included, you get a stack trace akin to the above.

I think, if there are required/mandatory fields for specific content, then the fields provided to list should be appended to a pre-defined list of mandatory fields.

Examples of mandatory fields.

distribution trees: header_version, release_name, release_short, release_version, release_is_layered, arch, build_timestamp, addons, checksums, images, variants

Module MD Defaults: module, stream, profiles

ModuleMD's: name, stream, versino, context, arch

Package Categories: id, name, description, digest

Package Environments: id, name, description, digest

Package Groups, id, name, description, digest

Metadata: relative_path, data_type, checksum_type, checksum

Not sure if this sort of issue is also applicable to other plugins.

History

#1 - 05/25/2021 05:17 PM - ttereshc

- Description updated

#2 - 05/25/2021 05:23 PM - daviddavis

- Triaged changed from No to Yes