

RPM Support - Story #3919

As a user, I can manage modular Errata content

08/14/2018 08:07 PM - ttereshc

Status:	CLOSED - CURRENTRELEASE	Start date:	
Priority:	Normal	Due date:	
Assignee:	milan	% Done:	100%
Category:		Estimated time:	0:00 hour
Sprint/Milestone:	2.18.0	Tags:	Pulp 2
Platform Release:	2.18.0	Sprint:	Sprint 43
Groomed:	Yes	Quarter:	
Sprint Candidate:	No		

Description

If an erratum is released for modular content, it will contain information about a module the RPMs are relevant to.

Information about a module is added to a collection in a pkglist.

Each collection corresponds to one module or to non-modular content.

There can be multiple collections in one pkglist (this is just for information, no changes here, it's this way now).

The module element is **optional**. Absence of the module element means that the collection is for normal non-modular RPMs and should be processed as usual.

Example of the modular errata content can be found in updateinfo.xml in the fixtures repo:

<https://repos.fedorapeople.org/pulp/pulp/fixtures/rpm-with-modules/>

Here is the snippet of the new format:

```
<pkglist>
  <collection short="">
    <name>coll_name1</name>
    <module name="kangaroo" stream="0" version="20180730223407" context="deadbeef" arch="noarch"/>
    <package arch="noarch" name="kangaroo" release="1" src="http://www.fedoraproject.org" version="0.3">
      <filename>kangaroo-0.3-1.noarch.rpm</filename>
    </package>
    <package ... >
      ...
    </package>
  </collection>
  <collection short="">
    <name>coll_name2</name>
    ...
  </collection>
</pkglist>
```

For upload, JSON is usually used, so the format will be:

```
{
  "issued": "2015-03-05 05:42:53 UTC",
  "pkglist": [{
    "name": "coll_name1",
    "module": {
      "name": "duck",
      "stream": "0",
      "version": "20180730233102",
      "context": "deadbeef",
      "arch": "noarch"
    }
  }],
}
```

```

    "packages": [{
      "arch": "noarch",
      "filename": "duck-0.7-1.noarch.rpm",
      "name": "duck",
      "release": "1",
      "sum": ["sha256", "ceb0f0bb58be244393cc565e8ee5ef0ad36884d8ba8
eec74542ff47d299a34c1"],
      "version": "0.7",
    }, {
      ...
    }],
  },
  {
    "name": "coll_name2",
    ...
  }],
  ...
}

```

Deliverables

- **Sync** should recognize module element in the erratum and save it on the model
- no changes to the model itself are needed (aka migration is not needed) due to changes in the nested structure, inside a pkglist.
- **Publish** should respect the module element and publish info about the module in the errata if this info is available
 - see the fixture repo
 - publish should filter by module being available in the repo; modular errata rpms not attached to a module present in the repo should be excluded from publishing
 - publish should continue filtering by rpms being available in the repo
 - if no rpms are found for/associated with an erratum but module information is specified, such erratum has to be published. It's possible that just module metadata changed and no rpms has been updated, e.g an rpm was removed from the artifacts to fix the module issue.
 - document that filtering of errata being published is performed according to rpms and modules present in a repo
 - make sure collection names are unique per erratum (Pulp takes care of that and regenerates collection names to make them unique. I think at the moment coll_name = repo_id + idx, the suggestion is to make it, repo_id + module info + idx)
- **Upload** should support the module element
 - see the JSON format in the description above
- **Copy**
 - simple copy is expected to just work
 - copy with --recursive flag should respect the module info in the erratum
 - a whole module should be copied in this case: **all** its RPMs, module itself but related module_defaults should not be copied.
- **Search API**
 - probably no changes needed, it should just work

Out of scope: **Applicability** (it will be implemented as a separate story).

Note

Mind that the module and rpms listed in the erratum are the fixed ones (the ones present in the repo)

Subtasks:

Story # 3959: Modular errata sync and publish

Issue # 4002: pulp-admin doesn't support uploading modular errata

CLOSED - CURRENT

CLOSED - WONTFIX

Associated revisions

Revision 7bb8f881 - 09/18/2018 05:58 PM - milan

Introduce module--erratum recursive copy

Previously, the modulemd units would be skipped during the recursive copy.

Fixes: #3919 <https://pulp.plan.io/issues/3919>

History

#2 - 08/15/2018 09:44 AM - ttereshc

- Description updated

#3 - 08/15/2018 10:06 AM - ttereshc

- Description updated

#4 - 08/15/2018 10:27 AM - ttereshc

- Description updated

#5 - 08/15/2018 11:28 AM - milan

- Description updated

#6 - 08/15/2018 11:42 AM - milan

- Description updated

#7 - 08/16/2018 05:08 PM - dkliban@redhat.com

- Sprint/Milestone set to 2.18.0

#8 - 08/16/2018 11:54 PM - daviddavis

- Groomed changed from No to Yes

- Sprint set to Sprint 41

#9 - 08/17/2018 01:05 PM - ipanova@redhat.com

from publish part:

```
publish should continue filtering by rpms being available in the repo
```

```
publish should filter by module being available in the repo
```

I suggest we explicitly first filter by module being available in the repo and based on result:

a) positive, continue with filtering by rpms being available in the repo

b) negative, to do not publish the collection at all(module name + packages)and do not proceed with filtering by rpms available in the repo. We do not want to end up to publish in updateinfo.xml module which is not present in the repo, but for some reason rpms are present in the repo)

#10 - 08/27/2018 03:04 PM - rchan

- Sprint changed from Sprint 41 to Sprint 42

#11 - 08/27/2018 07:47 PM - milan

- Status changed from NEW to ASSIGNED

- Assignee set to milan

#12 - 08/28/2018 01:41 PM - milan

- Description updated

ipanova@redhat.com wrote:

from publish part:
[...]

I suggest we explicitly first filter by module being available in the repo and based on result:

- a) positive, continue with filtering by rpms being available in the repo
- b) negative, to do not publish the collection at all(module name + packages)and do not proceed with filtering by rpms available in the repo. We do not want to end up to publish in updateinfo.xml module which is not present in the repo, but for some reason rpms are present in the repo)

+1 makes sense; I'm updating the description accordingly

- if no rpms are found for/associated with an erratum but module information is specified, such erratum has to be published. It's possible that just module metadata changed and no rpms has been updated, e.g an rpm was removed from the artifacts to fix the module issue.

I guess we need to distinguish between a situation where all the RPMs were filtered out by the (module) repo presence and a genuine, metadata-only modular erratum; a consequence of neglect in the former case would be an unwanted, "empty" update on the consumer side. The genuine, metadata-only erratum of a module might have to, in case of (modular) dependency updates, pull-in additional content into the repository to avoid dependency issues on the consumer side.

Probably we could address both these concerns by calculating implied dependencies of modular errata:

- publish should filter by module being available in the repo; modular errata rpms not attached to a module present in the repo should be excluded from publishing
- publish should continue filtering by rpms being available in the repo
- skip the module erratum publish altogether if there are unsatisfied (modular) dependencies in the repository
- if no rpms are found for/associated with an erratum but module information is specified, such erratum has to be published. It's possible that just module metadata changed and no rpms has been updated, e.g an rpm was removed from the artifacts to fix the module issue.

#13 - 08/28/2018 05:14 PM - ipanova@redhat.com

i **think** whenever erratum provides some changes/updates it would provide the updated content as well. So if there is some dependency change in the module metadata i would expect this dependency provided in the erratum. do we check on the erratum integrity or we rely on the ET correctness?

#14 - 08/29/2018 06:49 PM - milan

ipanova@redhat.com wrote:

i **think** whenever erratum provides some changes/updates it would provide the updated content as well. So if there is some dependency change in the module metadata i would expect this dependency provided in the erratum. do we check on the erratum integrity or we rely on the ET correctness?

I'm afraid it is also about the case when we filter all the rpms from an erratum so we make it metadata-only even tho it wasn't supposed to be. Anyway, I guess I'm being too cautious about the edge cases here; we can always bugfix later or wave in case nobody reports issues with that.

#15 - 08/29/2018 06:55 PM - milan

I just realized we might have to store the relation to the modulemd affected by the erratum; once parsed, the <module name=".." /> tag is lost and since we don't store the original erratum XML (because of the publish-time adjustments), we need to record the erratum--module relationship in the DB otherwise we won't be able to publish it later. I guess the most appropriate model is [ErratumPkgList](#) and I suppose the field should be a foreign key linking back to the Modulemd collection

Thoughts?

#16 - 08/30/2018 04:12 PM - milan

milan wrote:

I just realized we might have to store the relation to the modulemd affected by the erratum; once parsed, the <module name=".." /> tag is lost and since we don't store the original erratum XML (because of the publish-time adjustments), we need to record the erratum--module relationship in the DB otherwise we won't be able to publish it later. I guess the most appropriate model is [ErratumPkgList](#) and I suppose the field should be a foreign key linking back to the Modulemd collection

Thoughts?

Actually, I'm wrong \o/ :)

```
db.erratum_pkglists.find({'repo_id': 'modulerr'})[4]
{
  "_id" : ObjectId("5b87f4c030f252463b42b28a"),
  "errata_id" : "RHEA-2012:0059",
  "repo_id" : "modulerr",
  "collections" : [
    {
      "packages" : [
        {
          "src" : "http://www.fedoraproject.org",
          "name" : "kangaroo",
          "sum" : null,
          "filename" : "kangaroo-0.3-1.noarch.rpm",
          "epoch" : null,

```

```
        "version" : "0.3",
        "release" : "1",
        "arch" : "noarch"
    }
},
"name" : "coll_name1",
"module" : {
    "stream" : "0",
    "version" : "20180730223407",
    "arch" : "noarch",
    "name" : "kangaroo",
    "context" : "deadbeef"
},
"short" : ""
},
// ----->%-----
]
}
```

#17 - 08/31/2018 10:54 AM - milan

- Status changed from ASSIGNED to POST

sync&publish PR: https://github.com/pulp/pulp_rpm/pull/1173

#18 - 08/31/2018 12:21 PM - milan

- Status changed from POST to NEW

- Assignee deleted (milan)

#19 - 09/05/2018 08:27 PM - dkliban@redhat.com

- Status changed from NEW to ASSIGNED

- Assignee set to milan

#20 - 09/07/2018 05:22 PM - milan

- Checklist item [x] Upload supports modular errata set to Done

#21 - 09/07/2018 05:27 PM - milan

The [upload works](#) out of the box with the [sync&publish PR](#) applied.

#22 - 09/10/2018 05:33 PM - milan

- Checklist item [x] Simple copy works for modular errata set to Done

Just checked that the simple copy works OK

```
[vagrant@pulp2 pulp_rpm]$ http POST https://admin:admin@pulp2.dev/pulp/api/v2/repository
```

```
ies/modulerr_copy/actions/associate/ source_repo_id=modulerr criteria:="{\"type_ids\": [\"erratum\"], \"filters\": {\"unit\": {\"errata_id\": \"RHEA-2018:0042\"}}}"
HTTP/1.1 202 Accepted
Connection: Keep-Alive
Content-Length: 172
Content-Type: application/json; charset=utf-8
Date: Mon, 10 Sep 2018 15:30:43 GMT
Keep-Alive: timeout=5, max=10000
Server: Apache/2.4.33 (Fedora) OpenSSL/1.1.0h-fips mod_wsgi/4.5.15 Python/2.7
```

```
{
  "error": null,
  "result": null,
  "spawned_tasks": [
    {
      "_href": "/pulp/api/v2/tasks/e9e5b53c-ce69-465a-9f24-1a236964ca21/",
      "task_id": "e9e5b53c-ce69-465a-9f24-1a236964ca21"
    }
  ]
}
```

```
[vagrant@pulp2 pulp_rpm]$ http https://admin:admin@pulp2.dev/pulp/api/v2/tasks/e9e5b53c-ce69-465a-9f24-1a236964ca21/
HTTP/1.1 200 OK
Connection: Keep-Alive
Content-Length: 832
Content-Type: application/json; charset=utf-8
Date: Mon, 10 Sep 2018 15:31:01 GMT
ETag: "c75cc3612b1cbf4073d5602e0eab95ef"
Keep-Alive: timeout=5, max=10000
Server: Apache/2.4.33 (Fedora) OpenSSL/1.1.0h-fips mod_wsgi/4.5.15 Python/2.7
```

```
{
  "_href": "/pulp/api/v2/tasks/e9e5b53c-ce69-465a-9f24-1a236964ca21/",
  "_id": {
    "$oid": "5b968e237f513442b097f791"
  },
  "_ns": "task_status",
  "error": null,
  "exception": null,
  "finish_time": "2018-09-10T15:30:43Z",
  "id": "5b968e237f513442b097f791",
  "progress_report": {},
  "queue": "reserved_resource_worker-4@pulp2.dev.dq2",
  "result": {
    "units_failed_signature_filter": [],
    "units_successful": [
      {
        "type_id": "erratum",
        "unit_key": {
          "id": "RHEA-2018:0042"
        }
      }
    ]
  },
  "spawned_tasks": [],
  "start_time": "2018-09-10T15:30:43Z",
  "state": "finished",
  "tags": [
    "pulp:repository:modulerr_copy",
    "pulp:repository:modulerr",
    "pulp:action:associate"
  ],
  "task_id": "e9e5b53c-ce69-465a-9f24-1a236964ca21",
  "task_type": "pulp.server.managers.repo.unit_association.associate_from_repo",
  "traceback": null,
  "worker_name": "reserved_resource_worker-4@pulp2.dev"
}
```

```
[vagrant@pulp2 pulp_rpm]$
```

Double checking Mongo:

```
> db.repo_content_units.findOne({"repo_id": "modulerr_copy", "unit_type_id": "erratum"})
{
  "_id" : ObjectId("5b966c0c7f513442b096c9e7"),
  "repo_id" : "modulerr_copy",
  "unit_id" : "01f04ca4-0788-4e1c-84e8-9b2f21efab1f",
  "unit_type_id" : "erratum",
  "updated" : "2018-09-10T15:39:05Z",
  "created" : "2018-09-10T13:05:16Z"
}
> db.units_erratum.findOne({"_id": "01f04ca4-0788-4e1c-84e8-9b2f21efab1f"})
{
  "_id" : "01f04ca4-0788-4e1c-84e8-9b2f21efab1f",
  "pulp_user_metadata" : {
    },
  "_last_updated" : 1536332456,
  "errata_id" : "RHEA-2018:0042",
  "status" : "stable",
  "updated" : "2018-09-05 11:23:42",
  "description" : "Walrus Erratum; metadata-only",
  "issued" : "2002-02-20 00:22:00",
  "pushcount" : "1",
  "references" : [ ],
  "reboot_suggested" : false,
  "relogin_suggested" : false,
  "restart_suggested" : false,
  "from" : "betelgeuze",
  "version" : "1",
  "release" : "1",
  "type" : "enhancement",
  "pkglist" : [ ],
  "title" : "Walrus",
  "_ns" : "units_erratum",
  "_content_type_id" : "erratum"
}
>
```

#23 - 09/11/2018 10:31 AM - milan

Recursive copy will require a fix

```
[vagrant@pulp2 pulp_rpm]$ pulp-admin rpm repo create --repo-id modulerr_copy  
Successfully created repository [modulerr_copy]
```

```
[vagrant@pulp2 pulp_rpm]$ pulp-admin rpm repo copy errata -f modulerr -t modulerr_copy  
--str-eq 'errata_id=RHEA-2012:0058' --recursive  
This command may be exited via ctrl+c without affecting the request.
```

```
[\]  
Running...
```

```
Copied:  
erratum:  
  RHEA-2012:0058  
rpm:  
  gorilla-0.62-1-noarch
```

```
[vagrant@pulp2 pulp_rpm]$ pulp-admin rpm repo copy errata -f modulerr -t modulerr_copy  
--str-eq 'errata_id=RHEA-2018:0042' --recursive  
This command may be exited via ctrl+c without affecting the request.
```

```
[\]  
Running...
```

```
Copied:  
  RHEA-2018:0042
```

```
[vagrant@pulp2 pulp_rpm]$
```

#24 - 09/12/2018 03:39 PM - milan

- Checklist item [x] Recursive copy works for modular errata and copies modules as a whole set to Done

#25 - 09/12/2018 04:48 PM - milan

- Checklist item [x] Update documentation set to Done
Checklist item [x] Add release notes set to Done

#26 - 09/12/2018 05:52 PM - milan

- Checklist item [x] File a pulp-smash issue set to Done

Pulp issue: <https://github.com/PulpQE/Pulp-2-Tests/issues/94>

#27 - 09/17/2018 07:19 PM - rchan

- Sprint changed from Sprint 42 to Sprint 43

#28 - 09/18/2018 06:35 PM - milan

- Status changed from ASSIGNED to MODIFIED

- % Done changed from 50 to 100

Applied in changeset [7bb8f8813bde09760635fdaa30eb049800d2b652](#).

#29 - 09/18/2018 06:49 PM - ttereshc

- Platform Release set to 2.18.0

#30 - 10/29/2018 02:37 PM - kersom

This feature was already automated. See [Pulp 2 Test Case](#)

#31 - 10/30/2018 12:41 PM - rchan

kersom wrote:

This feature was already automated. See [Pulp 2 Test Case](#)

[kersom](#) - do we not list the Smash Test # linked to an story anymore? I'm glad to have this note here linking this story to the Smash Test, but curious why it's a note and not the link to the story?

#32 - 11/19/2018 10:38 PM - kersom

We already have automation in place for this one. [ManageModularErrataTestCase](#)

#33 - 11/20/2018 08:20 PM - ttereshc

- Status changed from MODIFIED to 5

#34 - 12/04/2018 11:11 PM - ttereshc

- Status changed from 5 to CLOSED - CURRENTRELEASE

#35 - 04/15/2019 10:09 PM - bmbouter

- Tags Pulp 2 added