

Pulp - Task #5944

[Epic] As a plugin writer, I can use a SigningService to produce ascii-armored signatures

01/07/2020 09:05 PM - dkliban@redhat.com

Status:	CLOSED - CURRENTRELEASE	Start date:	
Priority:	Normal	Due date:	
Assignee:	Imjachky	% Done:	100%
Category:		Estimated time:	0:00 hour
Sprint/Milestone:	3.1.0	Tags:	Documentation
Platform Release:		Sprint:	Sprint 66
Groomed:	Yes	Quarter:	
Sprint Candidate:	Yes		
Description			
<p>This is an epic that will enable plugin writers to use scripts provided by Pulp administrators to sign files.</p> <p>Pulp Administrators will use django-admin shell_plus or a python script to CRUD an instance of a SigningService model.</p> <p>Pulp Administrators will be able to retrieve an instance of a SigningService using the REST API.</p> <p>Plugin writers will be able to use SigningService.sign_file(filename) to retrieve an ascii-armored detached gpg signature and public key for verifying it.</p> <p>Administrators will have documentation that states the signing script needs to print a dictionary that looks like this:</p> <pre>{"file": "repomd.xml", "signature": "repomd.xml.asc", "key": "public.key"}</pre>			
Subtasks:			
Story # 5943: Add a SigningService model, viewset, and serializer			CLOSED - CURRENTRELEASE
Task # 5946: Add sign_file(filename) interface to SigningService model			CLOSED - CURRENTRELEASE

History

#1 - 01/07/2020 10:02 PM - bmbouter

Could we get away with having users use shell_plus for the nearterm? Otherwise we'll need to made the create, list, and delete interfaces too.

#2 - 01/08/2020 07:51 PM - dkliban@redhat.com

- Subject changed from *As an administrator I can create a Signing Service* to *Document how an administrator can create a Signing Service*
- Description updated
- Tags Documentation added

[bmbouter](#), yes we can get away with having admins use django-admin shell_plus command to create the Signing Service. I updated the description to reflect that this is simply a documentation task.

#3 - 01/09/2020 09:12 PM - bmbouter

- Sprint/Milestone set to 3.1.0
- Sprint Candidate changed from No to Yes

#4 - 01/10/2020 04:05 PM - bmbouter

- Groomed changed from No to Yes

#5 - 01/10/2020 04:52 PM - rchan

- Sprint set to Sprint 64

#6 - 01/10/2020 05:09 PM - davidDavis

- Tracker changed from Issue to Task
- % Done set to 0

#7 - 01/17/2020 06:53 PM - dkliban@redhat.com

- Subject changed from *Document how an administrator can create a Signing Service* to *As a plugin writer, I can use a SigningService to produce ascii-armored signatures*

- Description updated

#8 - 01/17/2020 07:37 PM - Imjachky

- Status changed from *NEW* to *ASSIGNED*

- Assignee set to *Imjachky*

#9 - 01/20/2020 11:03 PM - bmbouter

- Subject changed from *As a plugin writer, I can use a SigningService to produce ascii-armored signatures* to *[Epic] As a plugin writer, I can use a SigningService to produce ascii-armored signatures*

#10 - 01/24/2020 02:58 PM - rchan

- Sprint changed from *Sprint 64* to *Sprint 65*

#11 - 01/28/2020 02:32 PM - Imjachky

- At the end, it was decided that a detached signature will have the same name as a signed file. This consensus will be mentioned in the docs. The file type ".asc" shall be then appended to the signed filename, like so:

```
"repomd.xml" -> "repomd.xml.asc"
```

- Furthermore, in the future, there might be used a different signing method for a different type of content. Due to that, the model *SigningService* should be a parent class of more specific models (in our case, we currently have one example of such a model - *AsciiArmorDetachedSigningService*). The interface for the signing method remains untouched:

```
SigningService(Model) :
```

```
def sign() :  
    """Invoke an external signing service."""
```

```
AsciiArmorDetachedSigningService(SigningService) :
```

```
def sign() :  
    """Invoke inherited SigningService.sign() and validate returned data."""
```

#12 - 02/07/2020 03:20 PM - rchan

- Sprint changed from *Sprint 65* to *Sprint 66*

#13 - 02/11/2020 03:31 PM - daviddavis

- Sprint/Milestone changed from *3.1.0* to *3.2.0*

#14 - 02/17/2020 08:26 PM - bmbouter

- Description updated

- Status changed from *ASSIGNED* to *CLOSED - CURRENTRELEASE*

- Sprint/Milestone deleted (*3.2.0*)

All subtasks are complete, so I'm moving to *CLOSED - CURRENT RELEASE*

#15 - 02/17/2020 08:27 PM - bmbouter

- Sprint/Milestone set to *3.1.0*