Redmine - Patch #25240
Use SHA256 for attachment digest computation

2017-03-02 04:28 - Jens Krämer

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Jean-Philippe Lang	**% Done:**	0%
Category:	Attachments	**Estimated time:**	0.00 hour
Target version:	3.4.0		

Description

As discussed in #25215 we should change the digest method used for attachment checksums to something stronger. The attached patches

- widen the attachments.digest column to 64 chars
- change the digest used for new attachments to SHA256
- provide a rake task for upgrading the digest value of existing attachments

I also changed the label of the 'MD5' column header in the files list to 'Checksum'.

Related issues:

Blocks Redmine - Patch # 25215: Re-use existing identical disk files for new ...	**Closed**

Associated revisions

Revision 16454 - 2017-04-03 13:11 - Jean-Philippe Lang

Changes the digest used for attachments to SHA256 (#25240).

Patch by Jens Kraemer.

Revision 16455 - 2017-04-03 13:38 - Jean-Philippe Lang

Adds a rake task to update attachments digests to SHA256 (#25240).

Patch by Jens Krämer.

Revision 16456 - 2017-04-03 13:41 - Jean-Philippe Lang

Change MD5 table header to Checksum (#25240).

Patch by Jens Krämer.

Revision 16457 - 2017-04-03 13:42 - Jean-Philippe Lang

Adds :field_digest string to locales (#25240).

History

#1 - 2017-03-02 04:47 - Go MAEDA

- Blocks Patch #25215: Re-use existing identical disk files for new attachments added

#2 - 2017-03-02 05:46 - Go MAEDA

Thanks for the patch.
But I encountered the following error while running "rake redmine:attachments:update_digest_to_sha256" on my test environment.

```
rake aborted!
Errno::ENOENT: No such file or directory @ rb_sysopen - /Users/maeda/redmines/redmine-trunk/files/2006/07/060719210727_error281.txt
```

I think that Attachment#update_digest_to_sha256! should simply ignore the record if the corresponding file for the record is not exists.

#3 - 2017-03-02 07:17 - Jens Krämer

- File 0002-adds-a-rake-task-to-convert-the-digests-of-existing-.patch added

Yes. Here's the updated patch no. 2 with the readable? check added.

#4 - 2017-03-02 13:47 - Go MAEDA

- File add-algorithm-name.png added

Jens Krämer wrote:

> *Yes. Here's the updated patch no. 2 with the readable? check added.*

Thanks, now it works fine for me.

I have a suggestion. What about adding "MD5: " or "SHA256: " before hash values in app/views/files/index.html.erb? In the current implementation, it is a little bit difficult for users to know what hash algorithm is used to calculate the checksum.

```
diff --git a/app/views/files/index.html.erb b/app/views/files/index.html.erb
index 05fe37a..f111744 100644
--- a/app/views/files/index.html.erb
+++ b/app/views/files/index.html.erb
@@ -31,7 +31,7 @@
     <td class="created_on"><%= format_time(file.created_on) %></td>
     <td class="filesize"><%= number_to_human_size(file.filesize) %></td>
     <td class="downloads"><%= file.downloads %></td>
-    <td class="digest"><%= file.digest %></td>
+    <td class="digest"><%= file.digest.size < 64 ? "MD5" : "SHA256" %>: <%= file.digest %></td>
     <td class="buttons">
     <%= link_to(image_tag('delete.png'), attachment_path(file),
                                 :data => {:confirm => l(:text_are_you_sure)}, :method => :delete) if delete_allowed %>
```

add-algorithm-name.png

#5 - 2017-03-03 05:55 - Jens Krämer

Yes, that makes sense. The check for length of the digest to find out which it is isn't particularly nice (I know I did the same in my query for finding the

attachments to upgrade) but I'm still not sure adding the hashing algorithm as a field to the Attachment model is any better. Maybe we could have a method named Attachment#digest_type to at least clean up the view?

#6 - 2017-03-03 06:05 - Go MAEDA

Jens Krämer wrote:

> *Maybe we could have a method named Attachment#digest_type to at least clean up the view?*

Yes, absolutely agree.

#7 - 2017-03-03 21:42 - Jean-Philippe Lang

- Target version set to 3.4.0

#8 - 2017-03-12 00:57 - Jens Krämer

- File 0003-change-MD5-table-header-to-Checksum.patch added

here's the updated patch 3 showing the digest used for each file in the files list

#9 - 2017-04-03 13:48 - Jean-Philippe Lang

- Status changed from New to Closed
- Assignee set to Jean-Philippe Lang

Patches are committed, thanks Jens. I've made a few changes to the patches and changed the fixture used in the test to a binary file (possible failure due to \r\n EOLs).

#10 - 2021-04-27 00:08 - Shane Coles

I know this issue is really old at this point, but if anyone is still watching it by chance I could use some help. I am migrating a Redmine server to a FIPS validated server and running into issues because of the MD5 validation. I found this issue and it sounds like it could solve the problem. Unfortunately when I tried to run the Patches, it prompted me for which files I would like patched, and the answer is that I do now know.
If these patches can make it so that Redmine attachments/repos can be viewed on a FIPS server that would be great, and any instructions to that end would also be nice.
Thanks!

#11 - 2021-04-27 01:11 - Pavel Rosický

IIRC, even require 'digest/md5' is a problem on FIPS.

but Redmine use it in many other places
https://github.com/redmine/redmine/search?q=require+%27digest%2Fmd5%27

it's usually for cache keys calculations or gravatars, which is safe from a security perspective, but since the algorithm itself isn't allowed, the app won't work.

try to replace these occurrences with a different algorithm, but it may introduce incompatibilities. Do you know a way how to reliably test it? (I don't have a FIPS SW available)

you should open a new ticket if you want to discuss further since this one is already closed.

#12 - 2021-04-27 01:20 - Shane Coles

Thanks for the response. I will open a new ticket. I'm not a server expert at all, we've just always had this and I need to move it and ran into this. Hopefully someone on the new ticket will know how to do it.

Files

0002-adds-a-rake-task-to-convert-the-digests-of-existing-.patch	2.62 KB	2017-03-02	Jens Krämer
0001-changes-the-digest-used-for-attachments-to-SHA256.patch	5.49 KB	2017-03-02	Jens Krämer
0003-change-MD5-table-header-to-Checksum.patch	1.28 KB	2017-03-02	Jens Krämer
0002-adds-a-rake-task-to-convert-the-digests-of-existing-.patch	2.66 KB	2017-03-02	Jens Krämer
add-algorithm-name.png	65.3 KB	2017-03-02	Go MAEDA
0003-change-MD5-table-header-to-Checksum.patch	2.43 KB	2017-03-11	Jens Krämer