

## Redmine - Feature #2770

### Display of inline attached images in email notification

2009-02-18 02:22 - Chaoqun Zou

<b>Status:</b>	Reopened	<b>Start date:</b>	2009-02-18
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>	Attachments	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	Candidate for next minor release		
<b>Resolution:</b>			
<b>Description</b>			
<p>Now if we would like to include attached images in the issue's description, we can use !image_name.jpg!</p> <p>This works well when the issue is displayed in browser, but the image will not displayed correctly in the notification mail.</p> <p>So I thought that if we could upload the image before saving the issue could be a useful and flexible way.</p>			
<b>Related issues:</b>			
Related to Redmine - Feature # 3760: E-mail notifications for issue update/cr...		<b>New</b>	<b>2009-08-19</b>
Related to Redmine - Defect # 16989: Inline images in email does not appear w...		<b>Closed</b>	
Duplicated by Redmine - Defect # 9131: Link to attachment broken in notificat...		<b>Closed</b>	<b>2011-08-26</b>
Duplicated by Redmine - Defect # 5672: Inline images in issues description do...		<b>Closed</b>	<b>2010-06-11</b>

#### History

##### #1 - 2009-02-21 14:38 - Jean-Philippe Lang

- File *ticket\_with\_inline\_image.png* added

- Subject changed from *Add a upload button to the adding attachment area (useful for including attached images in the description) to Display of inline attached images in email notification*

The attachments are saved before the notification is sent, so this is not the problem.

I made a few tests with latest trunk and its works fine:

*ticket\_with\_inline\_image.png*

Notes:

- the image is displayed in the HTML email only (no formatting is done for the plain text part)
- you may need to configure your email client to authorize inline images to be displayed

##### #2 - 2009-02-23 03:37 - Chaoqun Zou

- Status changed from *New* to *Closed*

Yes, you are right. This feature works well.

I have encountered with a corrupted file which can't be displayed inline.

And I have test about 10 other attached images, all of them can be displayed correctly.

##### #3 - 2010-06-15 13:14 - Evgeny Mukhin

- Status changed from Closed to Reopened

It seems it doesn't work any longer. I've created Defect #5672

#### #4 - 2010-10-11 10:06 - Anonymous

Same here: images are not displayed in email notifications.

#### #5 - 2011-07-01 00:16 - Gilbert Flamino

- File 6-30-2011\_3-12-19\_PM.png added

- File 6-30-2011\_3-11-28\_PM.png added

Was this problem ever discovered? We're still experiencing problems with the initial notification emails - all update emails for the same issue go out are fine (see attachments).

The initial notification email that goes out contains just the original filename (e.g. ) any future update emails for the issue that go out contain the proper reference (i.e. ).

#### #6 - 2011-07-21 17:17 - Brian Crowell

The problem is that the issues.attachment attribute doesn't have anything in it by the time it reaches parse\_inline\_attachments (source:trunk/app/helpers/application\_helper.rb#L504). (The attachments are saved in source:trunk/app/controllers/issues\_controller.rb#L138 in the create action.)

I'm not a Ruby developer, so I'm not sure how to go about fixing it. I tried obj.attachments(true) because the ActiveRecord docs say that skips caching ([here](#)), but that didn't work.

How do you go about re-loading those?

#### #7 - 2011-07-21 19:23 - Brian Crowell

Figured it out. The mailer is sending out the message as soon as the issue is created, which is before attachments are added: source:trunk/app/models/issue\_observer.rb.

Now-- how to fix it? Delay the mailer until later in the creation process, or create the attachments in such a way that they are already on the issue when it gets saved?

#### #8 - 2011-07-21 20:38 - Brian Crowell

- File hack\_fix\_missing\_attachments.patch added

This patch fixes the issue, but in a hackish way.

#### #9 - 2011-07-21 23:26 - Brian Crowell

And now I see why it was done that way: The same code is needed in app/models/mail\_handler.rb, and the same probably needed for issue updates after their attachments are... attached.

**#10 - 2011-08-26 14:12 - Etienne Massip**

- Target version set to Candidate for next minor release

Indeed, generation of link URL needs an already persisted Attachment object and attachments are persisted after issue at creation (but before issue on issue update).

**#11 - 2012-02-29 15:55 - Terence Mill**

+1

**#12 - 2013-01-31 19:25 - Robert Hailey**

I'm not sure if what I'm experiencing is exactly this issue (or just a related one, as it effects only non-public projects), so feel free to point to the correct issue & I'll copy my comment there.

What we are seeing is that the requests for images from the email's being denied for security reasons (whereas gmail requests are within the same browser that is logged into redmine, so they work).

```
127.6.12.34 - - [2013-01-31T18:03:36+00:00] "GET /attachments/download/552 HTTP/1.1" 302 169 0.020
127.6.12.34 - - [2013-01-31T18:03:36+00:00] "GET
/login?back_url=http%3A%2F%2Four.redmine.host.com%2Fattachments%2Fdownload%2F552 HTTP/1.1" 200 4170 0.040
```

As such, my recommended fix (while maintaining **some** form of security) is that:

1. a new 'shared secret' field be added to the attachment table
2. that it be auto-populated with a random but ascii-nice hash
3. that urls to attachments from emails should include this shared secret (e.g. "/attachments/download/552/3MXtKsTQ")
4. that the attachment controller accept the hash and release the data to the email client

Variations might include the secret being per-ticket, or per-user (!), or changing with time/privileges (but breaks old emails).

**#13 - 2013-01-31 19:38 - Robert Hailey**

Come to think of it... lacing the url with the user's api key (e.g. as a query string) might already be a workable solution.

**#14 - 2013-02-14 13:14 - Al McNicoll**

While not everyone would be happy with this, I would personally (on my particular install) not have an issue with removing the authentication requirement on the attachments folder. Could this be done by modifying the apache config upstream of any processing by thin/mongrel/etc.?

If so, would anyone with better Apache than me mind suggesting how to prefix the existing config? Mine (the default Bitnami install) currently reads:

```
ProxyPass /redmine balancer://redminecluster
ProxyPassReverse /redmine balancer://redminecluster
```

```
<Proxy balancer://redminecluster>
  BalancerMember http://127.0.0.1:3001/redmine
  BalancerMember http://127.0.0.1:3002/redmine
</Proxy>
```

#### #15 - 2013-02-14 13:34 - Al McNicoll

Scratch that - just realised that you can't retrieve the full file path from the URL request without DB lookups, so the request has to be processed by redmine.

#### #16 - 2013-02-14 13:50 - Al McNicoll

- File `attachments_controller.allow_without_auth.patch` added

OK, so a quick hack for those in the same position as me (i.e. not too concerned about login security for attachment retrieval) is to use the attached patch.

**NOTE:** I've not used patch/diff before, so it may not be in the right format. If in doubt, the line proposed should be added at `attachments_controller.rb` after line 22.

I would love a proper (read: secure) fix, however, along the lines suggested by Robert Hailey. Is there a security concern with handing out the API key over email and putting it in every image URL?

#### #17 - 2013-02-14 16:55 - Robert Hailey

- File `f2770-unauthorized-attachments.patch` added

Wow thanks for the patch, this is really helpful.

Al McNicoll wrote:

```
NOTE: I've not used patch/diff before, so it may not be in the right format. If in doubt, the line proposed should be added at
attachments_controller.rb after line 22.
```

Yea, even a line number is still a bit shaky without a version number (& unmodified sources).

I was told some time ago that the customary way to run diff is one of two equivalent ways:

```
diff -wub $ORIGINAL $MODIFIED
diff -wub original/file.ext{.bak,}
```

...and that is now a habit, I don't think I use diff any other way.

I've taken the liberty of reformulating your patch (attached & seen below). Although I'm quite sure, it might be good to object if the added line is in the wrong place.

```
--- app/controllers/attachments_controller.rb.bak 2013-02-14 09:43:05.000000000 -0600
```

```
+++ app/controllers/attachments_controller.rb    2013-02-14 09:43:26.000000000 -0600
@@ -20,6 +20,7 @@
  before_filter :file_readable, :read_authorize, :only => [:show, :download]
  before_filter :delete_authorize, :only => :destroy
  before_filter :authorize_global, :only => :upload
+ skip_before_filter :check_if_login_required

  accept_api_auth :show, :download, :upload
```

#### #18 - 2013-02-14 17:23 - Robert Hailey

Unfortunately McNicoll's patch does not work for me (version 1.4.x), I have to disable the read\_authorize filter too:

```
diff --git a/app/controllers/attachments_controller.rb b/app/controllers/attachments_controller.rb
index cb975e8..46bd8ef 100644
--- a/app/controllers/attachments_controller.rb
+++ b/app/controllers/attachments_controller.rb
@@ -17,9 +17,10 @@

class AttachmentsController < ApplicationController
  before_filter :find_project, :except => :upload
- before_filter :file_readable, :read_authorize, :only => [:show, :download]
+ before_filter :file_readable, :read_authorize, :only => :show
  before_filter :delete_authorize, :only => :destroy
  before_filter :authorize_global, :only => :upload
+ skip_before_filter :check_if_login_required, :only => :download

  accept_api_auth :show, :download, :upload
```

#### #19 - 2013-03-21 17:59 - Fernando Hartmann

+1 to solve this.

A lot of my users complain about to have to log in Redmine just to see the in-line image.

How about to have an option to attach the image on the email body ? The mail will become bigger it may be a less price to pay to have the image in the email.

#### #20 - 2013-03-21 18:02 - Brian Crowell

Fernando Hartmann wrote:

```
| How about to have an option to attach the image on the email body ? The mail will become bigger it may be a less price to pay to have the image
| in the email.
```

That's actually what my patches in #3760 do.

#### #21 - 2013-03-21 18:06 - Fernando Hartmann

Brian Crowell wrote:

| *That's actually what my patches in #3760 do.*

So I suggest this two and #12516 should set as related

**#22 - 2013-03-21 18:07 - Brian Crowell**

Sorry for the duplicate update. Hit Refresh in the browser.

**#23 - 2013-05-21 11:24 - Christian Baus**

+1 for this :)

**#24 - 2013-08-04 14:11 - Jiří Křivánek**

+1

I would prefer having the configuration options:

- Either allow downloading the email attached images without login.
- Or enclose the images directly into the email body (multipart emails work fine these days to me with the QR-code images).
- Or keep the current behavior - i.e. the email images are completely useless as the email client cannot log into Redmine to download it (well, may be that some email clients can be configured for this - I am using Apple Mail and have no idea how to achieve this).

**#25 - 2014-06-24 13:32 - Toshi MARUYAMA**

- *Related to Defect #16989: Inline images in email does not appear when thumbnail macro is used. added*

**#26 - 2014-06-27 09:45 - Kevin Palm**

+1 for **any** of the proposed solutions

**#27 - 2014-11-05 11:27 - Sylvain Tissot**

+1 we encounter the same problem on a Redmine server running over HTTPS.

The inline images do not display in Apple Mail, instead the user sees a question mark.

**#28 - 2015-02-10 14:13 - Dima Kalachov**

Hey guys!

Here is solution using multipart emails: [https://github.com/dkalachov/redmine\\_email\\_images](https://github.com/dkalachov/redmine_email_images)

Tested on redmine 2.6 and 3.0.

**#29 - 2015-06-26 09:40 - huang huang**

Can't upload attachment when create new issue or update via email.

```
huanghanzhen@ee-200:~/rubyworkspace/function_tracker/redmine-3.0.3$ rake redmine:email:receive_pop3 RAILS_ENV="production"
host=fg-pop.chthibox.net username=xxxx password=xxxx folder=Inbox project=test allow_override=project --trace
** Invoke redmine:email:receive_pop3 (first_time)
** Invoke environment (first_time)
** Execute environment
** Execute redmine:email:receive_pop3
undefined method `charset' for nil:NilClass
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/plugins/redmine_email_images/lib/email_receive_inline_patch.rb:43:in
`decode_part_body'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/plugins/redmine_email_images/lib/email_receive_inline_patch.rb:28:in
`decoded_html'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/plugins/redmine_email_images/lib/email_receive_inline_patch.rb:15:in
`add_attachments_with_remove_inline_images'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/app/models/mail_handler.rb:206:in `receive_issue'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/app/models/mail_handler.rb:182:in `dispatch_to_default'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/app/models/mail_handler.rb:167:in `dispatch'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/app/models/mail_handler.rb:142:in `receive'
/var/lib/gems/2.1.0/gems/actionmailer-4.2.1/lib/action_mailer/base.rb:530:in `block in receive'
/var/lib/gems/2.1.0/gems/activesupport-4.2.1/lib/active_support/notifications.rb:164:in `block in instrument'
/var/lib/gems/2.1.0/gems/activesupport-4.2.1/lib/active_support/notifications/instrumenter.rb:20:in `instrument'
/var/lib/gems/2.1.0/gems/activesupport-4.2.1/lib/active_support/notifications.rb:164:in `instrument'
/var/lib/gems/2.1.0/gems/actionmailer-4.2.1/lib/action_mailer/base.rb:527:in `receive'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/app/models/mail_handler.rb:46:in `receive'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/app/models/mail_handler.rb:51:in `safe_receive'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/lib/redmine/pop3.rb:51:in `block (2 levels) in check'
/usr/lib/ruby/2.1.0/net/pop.rb:665:in `each'
/usr/lib/ruby/2.1.0/net/pop.rb:665:in `each_mail'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/lib/redmine/pop3.rb:48:in `block in check'
/usr/lib/ruby/2.1.0/net/pop.rb:531:in `start'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/lib/redmine/pop3.rb:43:in `check'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/lib/tasks/email.rake:161:in `block (4 levels) in <top (required)>'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/app/models/mailer.rb:383:in `with_synced_deliveries'
/home/huanghanzhen/rubyworkspace/function_tracker/redmine-3.0.3/lib/tasks/email.rake:160:in `block (3 levels) in <top (required)>'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:240:in `call'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:240:in `block in execute'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:235:in `each'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:235:in `execute'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:179:in `block in invoke_with_call_chain'
/usr/lib/ruby/2.1.0/monitor.rb:211:in `mon_synchronize'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:172:in `invoke_with_call_chain'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:165:in `invoke'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:150:in `invoke_task'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:106:in `block (2 levels) in top_level'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:106:in `each'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:106:in `block in top_level'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:115:in `run_with_threads'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:100:in `top_level'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:78:in `block in run'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:176:in `standard_exception_handling'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:75:in `run'
/var/lib/gems/2.1.0/gems/rake-10.4.2/bin/rake:33:in `<top (required)>'
```

```
/usr/local/bin/rake:23:in `load'  
/usr/local/bin/rake:23:in `'
```

### #30 - 2015-07-17 20:09 - Alexander Lyzhenkov

+1

### #31 - 2017-01-23 08:11 - Igor Antonevich

+1. Any ideas for fix this issue?

### #32 - 2018-10-01 11:43 - Erich Balan

+1. Is there an update about this issue?

### #33 - 2019-05-05 11:44 - Sunding Wei

With slight modification to @Robert Hailey, the fix works for Redmine v3.4.4-stable.

```
--- attachments_controller.rb.old 2019-05-05 17:33:49.050841458 +0800  
+++ attachments_controller.rb 2019-05-05 17:39:45.647950163 +0800  
@@ -18,7 +18,7 @@  
class AttachmentsController < ApplicationController  
  before_action :find_attachment, :only => [:show, :download, :thumbnail, :update, :destroy]  
  before_action :find_editable_attachments, :only => [:edit_all, :update_all]  
- before_action :file_readable, :read_authorize, :only => [:show, :download, :thumbnail]  
+ before_action :file_readable, :read_authorize, :only => [:show, :thumbnail]  
  before_action :update_authorize, :only => :update  
  before_action :delete_authorize, :only => :destroy  
  before_action :authorize_global, :only => :upload  
@@ -27,6 +27,9 @@  
  # MIME type text/javascript.  
  skip_after_action :verify_same_origin_request, :only => :download  
  
+ # INSECURE: download attachments without login, fix for showing images in email  
+ skip_before_action :check_if_login_required, :only => :download  
+  
  accept_api_auth :show, :download, :thumbnail, :upload, :update, :destroy  
  
  def show
```

### #34 - 2019-12-25 13:41 - Nick Nick

guys, what kind of plugin works on version 4.0.2?

### #35 - 2020-02-18 15:09 - Robert Röttger



Nick Nick wrote:

| *guys, what kind of plugin works on version 4.0.2?*

I'm interested too.

**#36 - 2020-02-24 15:18 - Robert Korulczyk**

Since now each user receives its own email message, it could be solved using tokens in URLs (similar as for atom feeds). All images in email message could use URLs like `https://example.com/attachments/download/123/image.png?key=somerandomtoken` - server could identify user by token and check if it has permissions to see this image.

**#37 - 2020-05-28 09:48 - Edouard COMTET**

If someone know how to fix that (allow everyone to access to attachment to display them in mails), I'm interested too.

**#38 - 2020-07-29 19:55 - David Gessel**

yes, me too. I'm very sad v4 broke the plugin that was making this work. It seems like a good core feature.

**Files**

---

ticket_with_inline_image.png	20.4 KB	2009-02-21	Jean-Philippe Lang
6-30-2011_3-12-19_PM.png	35.1 KB	2011-06-30	Gilbert Flamino
6-30-2011_3-11-28_PM.png	15.5 KB	2011-06-30	Gilbert Flamino
hack_fix_missing_attachments.patch	1.12 KB	2011-07-21	Brian Crowell
attachments_controller.allow_without_auth.patch	56 Bytes	2013-02-14	Al McNicoll
f2770-unauthorized-attachments.patch	467 Bytes	2013-02-14	Robert Hailey