

ステータス:	New	開始日:	2011/04/07
優先度:	通常	期日:	
担当者:		進捗率:	0%
カテゴリ:	Email receiving_29	予定工数:	0.00時間
対象バージョン:		作業時間:	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/8093	status_id:	1
category_id:	29	tracker_id:	1
version_id:	0	plus1:	0
issue_org_id:	8093	affected_version:	
author_id:	31669	closed_on:	
assigned_to_id:	0	affected_version_id:	31
comments:	20		

**説明**

Our remine system checks email account regularly for new emails. If we send any email to special address system processes it very good, but if we send an email with other email inside (we use MS Outlook feature "Send as attachment" for emails) then the email will be processed incorrectly and there won't be any attachments inside created issue. Obviously this is some special case in processing.

journals

Any thoughts how to fix or check it?

Could you please post the most simple / light test mail which fail to be processed by Redmine so that the issue can be reproduced it easily ?

Sure. This are 2 msg files of the message: one just in Unicode format.

Unfortunately, outlook doesn't have built-in capabilities for saving emails as emls.

I've converted the email to eml format with some external tool. I'm not sure if it is the same content as processed by the receiver (We use "rake redmine:email:receive\_pop3" approach). Is there any way to catch exactly the same content? (Say, extended information in log or something else)

Hi, any update about the issue?

Can I help with it?

Actually, it's a duplicate of #5994 but I keep it open since there is more details.

I spent half the day learning ruby so I could fix this today. So I have uploaded the patch that I came up with for Redmine 1.3.3 but I made this patch to work with the latest in the git repo.

It's really simple because Tmail has already converted the attachment object to EML and saves it to Redmine as "noname" so windows cannot open the file, I tried manually opening it with outlook and adding .msg to the file but no go. Then I looked up the code where it is saving the attachments and made it write it directly to the file system, then I found that it is already an EML. So all this patch does is if the attachment meets the 'message/rfc822' content type requirement then I name the file attachment.eml before it saves it to Redmine. I have also tried sending attachments inside an email attached as an attachment and it all comes through fine.

The patch is for app/models/mail\_handler.rb.

Unfortunately, that doesn't help in any way. After some tests we discovered that the internal email in this case doesn't look like attachment at least system think so.

## We used this code for check (mail\_handler.rb)

```
def add_attachments(obj)
  if email.has_attachments?
    email.attachments.each_with_index do |attachment, i|
      print "attachment.content_type = "
      print attachment.content_type
    end
  end
end
```

---

and there wasn't any output like `attachment.content_type = smth`

I should also say that in our case there is no noname file in Redmine after email was processed.

To sum that, I think the problem is that ruby mechanisms can't understand the structure of such kind of emails. And that is very sad.

## Looking forward to hearing from you, Joel.

Hmm, interesting.

I'm afraid I cannot help with that one, as the attachments are coming through but Redmine just didn't know how to handle them.

Keep in mind I am using 1.3.3 here, and Redmine uses "Tmail": <http://tmail.rubyforge.org/> for the mail handling.

## So you are using an older version which cannot handle the Outlook .msg files at all?

As I said we are using Redmine 1.1.2. May be there is a simple script which we can use for discovering our issues nature?

This is the list of gems we have:

- actionmailer (3.2.8, 2.3.14, 2.3.5, 2.3.2)
- actionpack (3.2.8, 2.3.14, 2.3.5, 2.3.2)
- activemodel (3.2.8)
- activerecord (3.2.8, 2.3.14, 2.3.5, 2.3.2)
- activerecord-mysql2-adapter (0.0.3)
- activeresource (3.2.8, 2.3.14, 2.3.5, 2.3.2)
- activesupport (3.2.8, 2.3.14, 2.3.5, 2.3.2)
- allison (2.0.3)
- arel (3.0.2)
- builder (3.0.0)
- bundler (1.1.5)
- cgi\_multipart\_eof\_fix (2.5.0)
- coderay (1.0.7)
- daemons (1.1.9, 1.0.10)
- echoe (3.1.1)
- erubis (2.7.0)
- fastercsv (1.5.5)
- fastthread (1.0.1 i386-mswin32)
- fiveruns\_tuneup (0.8.20)
- gem\_plugin (0.2.3)
- gemcutter (0.7.1)
- highline (1.6.13, 1.5.1)
- hike (1.2.1)
- hoe (3.0.8)
- i18n (0.6.0, 0.5.0, 0.4.2)
- journey (1.0.4)
- json\_pure (1.7.5)
- mail (2.4.4)
- mime-types (1.19)
- mongrel (1.1.5 x86-mswin32-60)
- mongrel\_service (0.3.4 i386-mswin32)
- multi\_json (1.3.6)
- mysql (2.8.1 x86-mswin32, 2.7.3 mswin32)
- mysql2 (0.3.11 x86-mswin32-60)

net-ldap (0.3.1)  
pik (0.2.8)  
polyglot (0.3.3)  
rack (1.4.1, 1.1.3, 1.0.1)  
rack-cache (1.2)  
rack-ssl (1.3.2)  
rack-test (0.6.1)  
rails (2.3.14, 2.3.5, 2.3.2)  
rake (0.9.2.2, 0.8.7)  
rmagick (2.10.0 mswin32)  
ruby-openid (2.2.0, 2.1.8)  
rubyforge (2.0.4, 1.0.3)  
sprockets (2.1.3)  
sqlite3 (1.3.6 x86-mswin32-60)  
sqlite3-ruby (1.2.3 mswin32)  
thor (0.16.0)  
tilt (1.3.3)  
treetop (1.4.10)  
tzinfo (0.3.33)  
win32-service (0.5.2 mswin32)

## Any ideas, Joel?

Hi again. We've just migrated to latest redmine version (2.2.0 to the moment). The issue is still here. So obviously that wasn't related with old redmine version we used before update.

Current list of gems:

actionmailer (3.2.11)  
actionpack (3.2.11)  
activemodel (3.2.11)  
activerecord (3.2.11)  
activeresource (3.2.11)  
activesupport (3.2.11)  
ansi (1.4.3)  
arel (3.0.2)  
aws-sdk (1.6.9)  
bcrypt-ruby (3.0.1 x86-mingw32)  
bigdecimal (1.1.0)  
bluecloth (2.1.0 x86-mingw32)  
builder (3.0.0)  
bundler (1.2.3, 1.0.21)  
coderay (1.0.6)  
coffee-rails (3.2.2)  
coffee-script (2.2.0)  
coffee-script-source (1.3.3)  
crack (0.3.1)  
daemons (1.1.8)  
diff-lcs (1.1.3)  
erubis (2.7.0)  
eventmachine (1.0.0 x86-mingw32)  
execjs (1.4.0)  
fastercsv (1.5.4)  
hike (1.2.1)  
hoe (3.0.6)  
httparty (0.8.3)  
i18n (0.6.0)  
io-console (0.3)  
journey (1.0.4)  
jquery-rails (2.1.3, 2.0.3)  
json (1.5.4)  
mail (2.4.4)  
metaclass (0.0.1)  
mime-types (1.19, 1.18)  
mini\_magick (3.4)  
mini\_portile (0.3.0)  
minitest (2.5.1)  
mocha (0.12.3)

multi\_json (1.3.6)  
multi\_xml (0.5.1)  
mustache (0.99.4)  
mysql2 (0.3.11 x86-mingw32, 0.2.18)  
net-ldap (0.3.1)  
nokogiri (1.5.5 x86-mingw32, 1.4.7 x86-mingw32)  
polyglot (0.3.3)  
prototype-rails (3.2.1)  
rack (1.4.1)  
rack-cache (1.2)  
rack-mount (0.8.3)  
rack-openid (1.3.1)  
rack-ssl (1.3.2)  
rack-test (0.6.1)  
rails (3.2.11)  
railties (3.2.11)  
rake (0.9.2.2)  
rake-compiler (0.8.1)  
rdoc (3.12, 3.9.4)  
RedCloth (4.2.9 x86-mingw32)  
rmagick (2.13.1)  
rspec (2.11.0)  
rspec-core (2.11.1)  
rspec-expectations (2.11.2)  
rspec-mocks (2.11.1)  
ruby-openid (2.1.8)  
rubygems-update (1.8.24)  
rvm (1.11.3.5)  
sass (3.1.20)  
sass-rails (3.2.5)  
sprockets (2.4.5, 2.2.1, 2.1.3, 2.1.2)  
sqlite3 (1.3.6 x86-mingw32)  
sqlite3-ruby (1.3.2 x86-mingw32)  
subexec (0.2.2)  
test-unit (2.5.2)  
thin (1.3.1)  
thor (0.15.4, 0.14.6)  
tilt (1.3.3)  
treetop (1.4.10)  
turn (0.9.6)  
tzinfo (0.3.33)  
uglifier (1.2.6)  
uuidtools (2.1.3)

## yard (0.8.1)

Hi! Finally we've been able to deal with this issue. The problem was this kind of attachments in emails generated by outlook was stored not like attachment but as part of multipart message with the following header

```
-----=_NextPart_000_01C9_01CDF973.EB5E8D50  
Content-Type: message/rfc822  
Content-Transfer-Encoding: 7bit  
Content-Disposition: attachment
```

We've created special patch for this case (attached)

I'm not an expert in ruby so that is obviously not a state of the art piece of code and I would be honored if someone would rewrite this to follow ruby, RoR, and Redmine guidelines.

Also that would be great if this code after revision will be added in Redmine core.

---

Bug still exists, checked on 2.3.1 stable, solution above doesn't work, any ideas how to fix it ?

Thank you.

---

I've just made a quick test with the attached sample email @This\_is\_an\_issue\_title.eml@ and it works fine. The issue is created with the test email as attachment, see below. Could you attach the content of your email for which it doesn't work?

!issue.png!

I just ran in the same error. After analyzing:

It is not a question which attachment you attach. The Question is how.

Look at the raw text of your Mails. If you attach with this, the attachment is ignored by redmine.

```
--GPJrCs/72TxItFYR
Content-Type: message/rfc822
Content-Disposition: inline
Content-Transfer-Encoding: 8bit
```

```
Return-path:
[...some mail...]
```

If you use this (with a name option), the attachment will be attached to your ticket.

```
--GPJrCs/72TxItFYR
Content-Type: message/rfc822
  name="withname.eml"
Content-Disposition: inline
Content-Transfer-Encoding: 8bit
```

```
Return-path:
[...some mail...]
```

```
my Details:
Redmine version      2.5.2.stable
Ruby version         1.9.3-p194 (2012-04-20) [x86_64-linux]
```

---

Sergeys patch worked for me with a minor change, because part.content\_disposition does not just contain "attachment" but a longer string...

Example:  
@#, , >@

This is the modified block from Sergeys patch:

```
logger.info "Checking multipart"
if email.parts
  email.parts.each do |part|
    p part.content_disposition
    if part.content_disposition =~ /attachment/i
      logger.info "Found email attachment in parts section"
      contents = part.decoded.gsub(0x0a.chr, 0x0d.chr+0x0a.chr) # this is to change newline type in attachment
      logger.info "Trying to parse it as email"
      attachment_email = Mail.new(contents)
      p attachment_email
      obj.attachments << Attachment.create(container => obj,
                                           :file => contents,
                                           :filename => attachment_email.subject + ".eml",
                                           :author => user,
                                           :content_type => part.content_type)
      logger.info "Attachment added"
    end
  end
end
logger.info "Attachments added"
```

---

Sergey Prosin wrote:

Hi! Finally we've been able to deal with this issue. The problem was this kind of attachments in emails generated by outlook was stored not like attachment but as part of multipart message with the following header

```
-----=_NextPart_000_01C9_01CDF973.EB5E8D50
Content-Type: message/rfc822
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment
```

We've created special patch for this case (attached)

I'm not an expert in ruby so that is obviously not a state of the art piece of code and I would be honored if someone would rewrite this to follow ruby, RoR, and Redmine guidelines.  
Also that would be great if this code after revision will be added in Redmine core.

I think it should be fixed by mail gem.

<https://github.com/mikel/mail/issues/851>

For what it is worth, I don't think this has ever worked for us.

Every 'Forward as Attachment' email that Redmine has touched only had the body text included in the resulting issue, not the attachments. We use Microsoft Exchange as the mail server and Microsoft Outlook as the email client. The oldest version we have installed is Outlook 2010, but later installations are all 2016. We've tested with Outlook 2010, 2013 and 2016, all with the same results.

We generally run the latest Redmine release version. Details for our installation:

@bin/about@

```
sh: 1: darcs: not found
sh: 1: hg: not found
sh: 1: cvs: not found
sh: 1: bzi: not found
sh: 1: git: not found
```

Environment:

```
Redmine version      3.3.3.stable.16539
Ruby version         2.3.1-p112 (2016-04-26) [x86_64-linux-gnu]
Rails version        4.2.7.1
Environment          production
Database adapter     Mysql2
```

SCM:

```
Subversion           1.9.3
Filesystem
```

Redmine plugins:

```
no plugin installed
```

@bundle list@

Gems included by the bundle:

```
* actionmailer (4.2.7.1)
* actionpack (4.2.7.1)
* actionpack-action_caching (1.1.1)
* actionpack-xml_parser (1.0.2)
* actionview (4.2.7.1)
* activejob (4.2.7.1)
* activemodel (4.2.7.1)
* activerecord (4.2.7.1)
* activesupport (4.2.7.1)
* addressable (2.4.0)
* arel (6.0.3)
* builder (3.2.2)
* bundler (1.11.2)
* coderay (1.1.1)
* concurrent-ruby (1.0.2)
* css_parser (1.4.5)
```

- \* erubis (2.7.0)
- \* globalid (0.3.7)
- \* htmlentities (4.3.4)
- \* i18n (0.7.0)
- \* jquery-rails (3.1.4)
- \* json (1.8.3)
- \* loofah (2.0.3)
- \* mail (2.6.4)
- \* mime-types (3.1)
- \* mime-types-data (3.2016.0521)
- \* mimemagic (0.3.2)
- \* mini\_portile2 (2.1.0)
- \* minitest (5.9.1)
- \* mysql2 (0.3.21)
- \* net-ldap (0.12.1)
- \* nokogiri (1.6.8.1)
- \* protected\_attributes (1.1.3)
- \* rack (1.6.4)
- \* rack-openid (1.4.2)
- \* rack-test (0.6.3)
- \* rails (4.2.7.1)
- \* rails-deprecated\_sanitizer (1.0.3)
- \* rails-dom-testing (1.0.7)
- \* rails-html-sanitizer (1.0.3)
- \* railties (4.2.7.1)
- \* rake (11.3.0)
- \* rbpdf (1.19.2)
- \* rbpdf-font (1.19.1)
- \* rdoc (4.3.0)
- \* redcarpet (3.3.4)
- \* request\_store (1.0.5)
- \* rmagick (2.16.0)
- \* roadie (3.2.0)
- \* roadie-rails (1.1.1)
- \* ruby-openid (2.3.0)
- \* sprockets (3.7.0)
- \* sprockets-rails (3.2.0)
- \* thor (0.19.1)
- \* thread\_safe (0.3.5)
- \* tzinfo (1.2.2)
- \* yard (0.9.5)

I tested this on 4.1.1 stable and still is a problem. In my situation Redmine reject incoming email. No new issue, no new issue answer.

related\_issues

relates,Closed,15793,Problems with receiving emails with attached .msg files  
duplicates,New,5994,Creating new issue via email submission strips attached emails

## 履歴

#1 - 2022/05/10 17:19 - Admin Redmine

- カテゴリをEmail receiving\_29 にセット