Redmine - Defect #6355

Redmine notification mailer.rb creates email messages with RFC-violating Message-ID: headers containing invalid ".." (doubled periods) sub-strings

2010-09-10 10:46 - Keith Farrar

Status:	Closed	**Start date:**	2010-09-10
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Email notifications	**Estimated time:**	0.00 hour
Target version:			
Resolution:	Cant reproduce	**Affected version:**	

Description

Redmine notification mailer.rb creates email messages with RFC-violating Message-ID: headers containing invalid ".." (doubled periods) sub-strings. The invalid messages trigger our anti-spam filters.

Issue appears on version 0.8.5 and in version 0.9.4. The message-id generating code appears unchanged in 1.0.1.

It might be a bug in mailer.rb code - messages for items with an associated object (bug status update messages to watchers) seem valid, but account related messages (new user registration and password reset) create bogus message-ids.

No header format complaints are generated for substituting underscore for dot as separator in left hand side of message-id header. So long as there's no code depending on parsing dot-separated contents in that string, a simple substitution should fix the header format.

The RFC 5322 spec disallows consecutive periods in a mail address or a Message-ID header.
See http://tools.ietf.org/html/rfc5322

Probably needs this code fixed in app/models/mailer.rb, replacing underscore for period as the separator.

```
 # Returns a predictable Message-Id for the given object
  def self.message_id_for(object)
    # id + timestamp should reduce the odds of a collision
    # as far as we don't send multiple emails for the same object
    timestamp = object.send(object.respond_to?(:created_on) ? :created_on : :updated_on)
    hash = "redmine.#{object.class.name.demodulize.underscore}-#{object.id}.#{timestamp.strftime("%Y%m%d%H%M%S")}"
    host = Setting.mail_from.to_s.gsub(%r{^.*@}, '')
    host = "#{::Socket.gethostname}.redmine" if host.empty?
    "<#{hash}@#{host}>"
  end
```

History

#1 - 2010-09-10 12:07 - Felix Schäfer

Could you give an example of what one of the malformed strings looks like? I'd wager this happens only for specific objects, and those would need mending, not the mailer.

#2 - 2010-09-10 17:06 - Keith Farrar

Note: I suggested changing the period separator because it's a special character in Message-ID headers. Unless there is a Message-ID header parser used within Redmine, there is no need to use period as the generated field separator.

I'm mostly ignoring the issue with 0.8.* (which also generated bogus Message-ID headers for bug status updates).

What I've observed so far with version 0.9.4: messages for items with an associated object (bug status update messages to watchers) have valid generated Message-ID headers, but account related notification messages (new user registration and password reset) create bogus Message-ID headers.

Here is an example invalid Message-ID header from a new user account activation request:

```
Syntax error in Message-Id: value found on alpha.xerox.com:
    Message-Id:    <4c7819d616aff_452..fdba2b3801fb@extwebwiki.tmail>
                       ^-illegal word in localpart
```

#3 - 2010-09-10 18:21 - Felix Schäfer

Mmh, Message-IDs should be of the form redmine.${OBJECT_TYPE}-${OBJECT_ID}.${OBJECT_TIMESTAMP_OR_NOW}${AT}${REDMINE_HOST}, what you have there doesn't fit into that scheme. Do you have any plugin installed that could affect that? Could you try to list the (reproducible) actions that cause those mails?

#4 - 2010-09-10 18:31 - Keith Farrar

Apparently, Redmine supplied no value for the field "${OBJECT_TYPE}-${OBJECT_ID}" for new account requests nor for password reset requests. That empty/NULL value causes invalid Message-ID headers in the notification messages.

#5 - 2010-09-10 18:35 - Felix Schäfer

No, the Message-ID you pasted neither begins with redmine. (that part is immutable) nor does it end with @${REDMINE_HOST}, in fact the .tmail TLD would suggest that it's generated by the rails mailing facilities.

#6 - 2010-09-10 19:22 - Keith Farrar

Okay, I was looking in the wrong place. I'm certainly not a Rails expert. :)

Looking at the Github 1.0-stable branch.

What's the default template for Message-ID headers in Rails tmail?

These methods in app/models/mailer.rb create tmail objects without calling message_id:

- reminder
- document_added
- attachments_added
- account_information
- account_activation_request
- account_activated
- lost_password
- register
- test

#7 - 2010-09-10 20:08 - Felix Schäfer

Keith Farrar wrote:

> What's the default template for Message-ID headers in Rails tmail?

TMail's MessageIDs are generated here: http://github.com/mikel/tmail/blob/master/lib/tmail/utils.rb#L71 , and they should never contain any period.

#8 - 2019-02-10 07:37 - Go MAEDA

- Status changed from New to Closed

- Resolution set to Cant reproduce

A Message-id is never generated in such a format. Please reopen this issue if you still experiencing the problem.

source:tags/4.0.1/app/models/mailer.rb#L711

```
  def self.token_for(object, rand=true)
    timestamp = object.send(object.respond_to?(:created_on) ? :created_on : :updated_on)
    hash = [
      "redmine",
      "#{object.class.name.demodulize.underscore}-#{object.id}",
      timestamp.strftime("%Y%m%d%H%M%S")
    ]
    if rand
      hash << Redmine::Utils.random_hex(8)
    end
    host = Setting.mail_from.to_s.strip.gsub(%r{^.*@|>}, '')
    host = "#{::Socket.gethostname}.redmine" if host.empty?
    "#{hash.join('.')}@#{host}"
  end
```