

## Redmine - Feature #32085

### Allow newline as a separator in "Allowed extensions", "Disallowed extensions", "Exclude attachments by name" field in Administration

2019-09-18 03:32 - Go MAEDA

<b>Status:</b>	New	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>	Administration	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	Candidate for next major release		
<b>Resolution:</b>			
<b>Description</b>			
Currently, those fields are displayed as textarea but only comma is allowed as a values separator.  I think newline should be allowed as well because it is very natural to use newline in order to separate values in textarea.			
<b>Related issues:</b>			
Related to Redmine - Feature #19903: Change textfield to textarea for "Exclud...			<b>Closed</b>

#### History

##### #1 - 2019-09-18 03:32 - Go MAEDA

- Related to Feature #19903: Change textfield to textarea for "Exclude attachments by name" added

##### #2 - 2020-02-21 02:30 - Yuichi HARADA

- File 32085-allow-newline-as-separator.patch added

+1  
I think that allowing a newline as a separator makes it easier to type and easier to read. I attached a patch.

```
diff --git a/app/models/attachment.rb b/app/models/attachment.rb
index 627c1a181..f033affb7 100644
--- a/app/models/attachment.rb
+++ b/app/models/attachment.rb
@@ -417,7 +417,7 @@ class Attachment < ActiveRecord::Base
   extension = extension.downcase.sub(/\A\./, '')

   unless extensions.is_a?(Array)
-   extensions = extensions.to_s.split(",").map(&:strip)
+   extensions = extensions.to_s.split(/\s,+/)
   end
   extensions = extensions.map {|s| s.downcase.sub(/\A\./, '')}.reject(&:blank?)
   extensions.include?(extension)
diff --git a/app/models/mail_handler.rb b/app/models/mail_handler.rb
index eccc93a2a..8ef5ed24b 100644
--- a/app/models/mail_handler.rb
+++ b/app/models/mail_handler.rb
@@ -308,7 +308,7 @@ class MailHandler < ActionMailer::Base

   # Returns false if the +attachment+ of the incoming email should be ignored
   def accept_attachment?(attachment)
-   @excluded ||= Setting.mail_handler_excluded_filenames.to_s.split(',').map(&:strip).reject(&:blank?)
+   @excluded ||= Setting.mail_handler_excluded_filenames.to_s.split(/\s,+/).reject(&:blank?)
   @excluded.each do |pattern|
     if Setting.mail_handler_enable_regex_excluded_filenames?
       regexp = %r{\A#{pattern}\z}i
```

##### #3 - 2020-02-25 03:34 - Kevin Fischer

Looks good to me!

Just a little remark about the test code:  
Maybe it's better to add a new extension separated by a newline to the back instead of replacing the comma completely. In that way both separators can be tested and not just the newline

#### #4 - 2020-03-10 08:26 - Yuichi HARADA

- File 32085-allow-newline-as-separator-v2.patch added

Kevin Fischer wrote:

Just a little remark about the test code:

Maybe it's better to add a new extension separated by a newline to the back instead of replacing the comma completely. In that way both separators can be tested and not just the newline

Thank you for pointing out. I fixed it to test with comma and newline delimiters.

```
diff --git a/test/unit/attachment_test.rb b/test/unit/attachment_test.rb
index 7e12483f5..c0c0bb109 100644
--- a/test/unit/attachment_test.rb
+++ b/test/unit/attachment_test.rb
@@ -139,7 +139,7 @@ class AttachmentTest < ActiveSupport::TestCase
  end

  def test_extension_should_be_validated_against_denied_extensions
-   with_settings :attachment_extensions_denied => "txt, png" do
+   with_settings :attachment_extensions_denied => "txt, png\r\ncsv" do
    a = Attachment.new(:container => Issue.find(1),
                      :file => mock_file_with_options(:original_filename => "test.jpeg"),
                      :author => User.find(1))
@@ -153,11 +153,11 @@ class AttachmentTest < ActiveSupport::TestCase
  end

  def test_valid_extension_should_be_case_insensitive
-   with_settings :attachment_extensions_allowed => "txt, Png" do
+   with_settings :attachment_extensions_allowed => "txt, Png\r\nCSV" do
    assert Attachment.valid_extension?(".png")
    assert !Attachment.valid_extension?(".jpeg")
  end
-   with_settings :attachment_extensions_denied => "txt, Png" do
+   with_settings :attachment_extensions_denied => "txt, Png\r\nCSV" do
    assert !Attachment.valid_extension?(".png")
    assert Attachment.valid_extension?(".jpeg")
  end
end

diff --git a/test/unit/mail_handler_test.rb b/test/unit/mail_handler_test.rb
index 07532d82e..fdef53478 100644
--- a/test/unit/mail_handler_test.rb
+++ b/test/unit/mail_handler_test.rb
@@ -33,6 +33,7 @@ class MailHandlerTest < ActiveSupport::TestCase
  FIXTURES_PATH = File.dirname(__FILE__) + '/../fixtures/mail_handler'

  def setup
+   set_tmp_attachments_directory
    ActionMailer::Base.deliveries.clear
    Setting.notified_events = Redmine::Notifiable.all.collect(&:name)
    User.current = nil
@@ -546,7 +547,6 @@ class MailHandlerTest < ActiveSupport::TestCase
  end

  def test_add_issue_from_apple_mail
-   set_tmp_attachments_directory
    issue = submit_email(
      'apple_mail_with_attachment.eml',
      :issue => {:project => 'ecookbook'})
@@ -563,7 +563,6 @@ class MailHandlerTest < ActiveSupport::TestCase
  end

  def test_thunderbird_with_attachment_ja
-   set_tmp_attachments_directory
    issue = submit_email(
      'thunderbird_with_attachment_ja.eml',
      :issue => {:project => 'ecookbook'})
@@ -588,7 +587,6 @@ class MailHandlerTest < ActiveSupport::TestCase
  end

  def test_gmail_with_attachment_ja
-   set_tmp_attachments_directory
    issue = submit_email(
      'gmail_with_attachment_ja.eml',
```

```

      :issue => {:project => 'ecookbook'}
@@ -604,7 +602,6 @@ class MailHandlerTest < ActiveSupport::TestCase
end

def test_thunderbird_with_attachment_latin1
- set_tmp_attachments_directory
  issue = submit_email(
    'thunderbird_with_attachment_iso-8859-1.eml',
    :issue => {:project => 'ecookbook'})
@@ -623,7 +620,6 @@ class MailHandlerTest < ActiveSupport::TestCase
end

def test_gmail_with_attachment_latin1
- set_tmp_attachments_directory
  issue = submit_email(
    'gmail_with_attachment_iso-8859-1.eml',
    :issue => {:project => 'ecookbook'})
@@ -642,7 +638,6 @@ class MailHandlerTest < ActiveSupport::TestCase
end

def test_mail_with_attachment_latin2
- set_tmp_attachments_directory
  issue = submit_email(
    'ticket_with_text_attachment_iso-8859-2.eml',
    :issue => {:project => 'ecookbook'})
@@ -995,7 +990,6 @@ class MailHandlerTest < ActiveSupport::TestCase
end

def test_reply_to_a_nonexistent_issue
- set_tmp_attachments_directory
  Issue.find(2).destroy
  assert_no_difference 'Issue.count' do
    assert_no_difference 'Journal.count' do
@@ -1157,7 +1151,7 @@ class MailHandlerTest < ActiveSupport::TestCase
end

def test_attachments_that_match_mail_handler_excluded_filenames_should_be_ignored
- with_settings :mail_handler_excluded_filenames => "*.vcf,\n *.jpg" do
+ with_settings :mail_handler_excluded_filenames => "*.vcf,\n *.jpg\r\n*.csv" do
  issue = submit_email('ticket_with_attachment.eml', :issue => {:project => 'onlinestore'})
  assert issue.is_a?(Issue)
  assert !issue.new_record?
@@ -1166,7 +1160,7 @@ class MailHandlerTest < ActiveSupport::TestCase
end

def test_attachments_that_match_mail_handler_excluded_filenames_by_regex_should_be_ignored
- with_settings :mail_handler_excluded_filenames => '.*\.\vcf, (pa|nut)ella\.jpg',
+ with_settings :mail_handler_excluded_filenames => ".*\.\vcf, (pa|nut)ella\.\.jpg\r\n.*\.\.csv",
  :mail_handler_enable_regex_excluded_filenames => 1 do
  issue = submit_email('ticket_with_attachment.eml', :issue => {:project => 'onlinestore'})
  assert issue.is_a?(Issue)

```

#### #5 - 2020-03-24 15:23 - Go MAEDA

- Target version set to Candidate for next major release

#### #6 - 2020-04-25 10:27 - Go MAEDA

- File 32085-allow-newline-as-separator-v3.patch added

- Target version changed from Candidate for next major release to 4.2.0

I have removed some refactorings because they are not directly related to this issue and should be a separate patch.

Setting the target version to 4.2.0.

#### #7 - 2020-05-05 04:25 - Go MAEDA

I noticed that a user cannot configure "Exclude attachments by name" to reject attachments that contain spaces in their name (ex: "untitled file.txt") if the patch is applied.

Maybe it is better to use `split(/[\r\n,]+/)` instead of `split(/[\s,]+/)`, isn't it?

#### #8 - 2020-05-07 04:04 - Yuichi HARADA

Go MAEDA wrote:

I noticed that a user cannot configure "Exclude attachments by name" to reject attachments that contain spaces in their name (ex: "untitled file.txt") if the patch is applied.

Maybe it is better to use `split(/[\r\n,]+/)` instead of `split(/[s,]+/)`, isn't it?

Maybe this regular expression `split(/[\r\n,]+/)` alone can't be separated by comma and space delimiters (ex: "txt, png"). I think it's better to use `split(/[\r\n,]+/).map(&:strip)` instead of `split(/[\r\n,]+/)`.

**#9 - 2021-03-26 08:39 - Go MAEDA**

- Target version changed from 4.2.0 to 5.0.0

Yuichi HARADA wrote:

Go MAEDA wrote:

I noticed that a user cannot configure "Exclude attachments by name" to reject attachments that contain spaces in their name (ex: "untitled file.txt") if the patch is applied.

Maybe it is better to use `split(/[\r\n,]+/)` instead of `split(/[s,]+/)`, isn't it?

Maybe this regular expression `split(/[\r\n,]+/)` alone can't be separated by comma and space delimiters (ex: "txt, png"). I think it's better to use `split(/[\r\n,]+/).map(&:strip)` instead of `split(/[\r\n,]+/)`.

This patch requires a fix and a few test cases. Changing the target version to 5.0.0.

**#10 - 2022-03-21 21:08 - Marius BĂLTEANU**

- Target version changed from 5.0.0 to Candidate for next major release

**Files**

---

32085-allow-newline-as-separator.patch	4.89 KB	2020-02-21	Yuichi HARADA
32085-allow-newline-as-separator-v2.patch	5.92 KB	2020-03-10	Yuichi HARADA
32085-allow-newline-as-separator-v3.patch	3.67 KB	2020-04-25	Go MAEDA