

Redmine - Patch #35721

Unlink files after they're closed

2021-08-06 17:33 - Pavel Rosický

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Attachments	Estimated time:	0.00 hour
Target version:			

Description

Tests are trying to unlink the file inside the block while the file descriptor is still active.

they were introduced by #35539

fixes *Errno::EACCES: Permission denied*

```
diff --git a/test/unit/attachment_test.rb b/test/unit/attachment_test.rb
index 2aee87964..a870e3697 100644
--- a/test/unit/attachment_test.rb
+++ b/test/unit/attachment_test.rb
@@ -276,27 +276,31 @@ class AttachmentTest < ActiveSupport::TestCase
  end

  def test_create_diskfile
+   path = nil
    Attachment.create_diskfile("test_file.txt") do |f|
      path = f.path
      assert_match(/^d{12}_test_file.txt$/, File.basename(path))
      assert_equal 'test_file.txt', File.basename(path)[13..-1]
-    File.unlink f.path
    end
+   File.unlink path

    Attachment.create_diskfile("test_accentu .txt") do |f|
+   path = f.path
      assert_equal '770c509475505f37c2b8fb6030434d6b.txt', File.basename(f.path)[13..-1]
-    File.unlink f.path
    end
+   File.unlink path

    Attachment.create_diskfile("test_accentu ") do |f|
+   path = f.path
      assert_equal 'f8139524ebb8f32e51976982cd20a85d', File.basename(f.path)[13..-1]
-    File.unlink f.path
    end
+   File.unlink path

    Attachment.create_diskfile("test_accentu . a") do |f|
+   path = f.path
      assert_equal 'cbb5b0f30978ba03731d61f9f6d10011', File.basename(f.path)[13..-1]
-    File.unlink f.path
    end
  end
```

```
+ File.unlink path
end

def test_title
```

Related issues:

Related to Redmine - Patch # 35539: Race condition (possible filename collisi...

Closed

Associated revisions**Revision 21194 - 2021-08-18 04:47 - Go MAEDA**

Don't unlink files before closing them (#35539, #35721).

Patch by Pavel Rosický.

History**#1 - 2021-08-08 03:47 - Go MAEDA**

- *Related to Patch #35539: Race condition (possible filename collision) in Attachment.disk_filename added*

#2 - 2021-08-09 16:59 - Holger Just

The tests were specifically written to simulate the race condition. On Filesystems following the usual POSIX semantics, it is allowed to unlink files while they are still opened by any process. They will only actually be removed from the filesystem when the last filehandle is closed by the kernel.

On Windows / NTFS, this is stricter. Here, the filesystem denies moving / deleting any file for which there are open file handles (that is, unless all file handles have the ::File::SHARE_DELETE flag set). Thus, on windows the test might fail.

In any case, I think this patch of the tests is safe and can be applied.

#3 - 2021-08-09 19:40 - Mischa The Evil

- *Description updated*

#4 - 2021-08-18 04:51 - Go MAEDA

- *Status changed from New to Closed*

Committed the fix as a part of #35539. Thank you.