

Setting an Out-of-Range Value for ‘Session Lifetime’ Causes a 500 Internal Server Error and Renders Redmine Unusable

2025-01-07 03:00 - Keiji Aita

Status:	New	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Administration	Estimated time:	0.00 hour
Target version:		Affected version:	
Resolution:			

Description

Envitonment

- Redmine version 5.1.4.stable
- Ruby version 3.2.6-p234 (2024-10-30) [x86_64-linux]
- Rails version 6.1.7.10
- Database adapter Mysql2
- No plugins

Description

Regarding the “Session lifetime” setting under [Administration] > [Settings] > [Authentication] tab in Redmine, if you set a value larger than any of the available choices, Redmine results in a 500 Internal Server Error, making login and any other operations impossible.

In general, this issue does not occur with normal operations in the GUI (WebUI), since values are selected from a dropdown menu. The conditions that trigger this issue are as follows:

Obtain the authenticity_token during login with Redmine administrator privileges.
Send the corresponding setting value via a POST request using the authenticity_token.
Based on these conditions, the access complexity (AC: Access Complexity) is medium, and authentication (Au: Authentication) is single. However, since Redmine itself becomes unusable, this bug is being reported as an issue.

Log

- production.log(session_lifetime):

```
2024-11-11 13:10:22 I, [2024-11-11T04:10:22.213565 #1] INFO -- : Started GET "/" for 172.22.0.1 at 2024-11-11 04:10:22 +0000
2024-11-11 13:10:22 I, [2024-11-11T04:10:22.214153 #1] INFO -- : Processing by WelcomeController#index as HTML
2024-11-11 13:10:22 I, [2024-11-11T04:10:22.217841 #1] INFO -- : Completed 500 Internal Server Error in 4ms (ActiveRecord: 0.7ms | Allocations: 1117)
2024-11-11 13:10:22 F, [2024-11-11T04:10:22.218845 #1] FATAL -- :
2024-11-11 13:10:22 ActiveRecord::StatementInvalid (Mysql2::Error: Incorrect DATETIME value: '-4463005062735503222482655784856603883025098983030448705846414029523342816092144159077548313762843174161435924469338849042616476') :
2024-11-11 13:10:22
2024-11-11 13:10:22 app/models/user.rb:478:in `verify_session_token'
2024-11-11 13:10:22 app/controllers/application_controller.rb:86:in `session_expired?'
2024-11-11 13:10:22 app/controllers/application_controller.rb:76:in `session_expiration'
```

- production.log (session_timeout):

```
2024-11-11 13:06:30 I, [2024-11-11T04:06:30.052984 #1] INFO -- : Started GET "/" for 172.22.0.1 at 2024-11-11 04:06:30 +0000
```

```
2024-11-11 13:06:30 I, [2024-11-11T04:06:30.053969 #1] INFO -- : Processing by WelcomeController#index as HTML
2024-11-11 13:06:30 I, [2024-11-11T04:06:30.058140 #1] INFO -- : Completed 500 Internal Server Error in 4ms (ActiveRecord: 0.9ms | Allocations: 1254)
2024-11-11 13:06:30 F, [2024-11-11T04:06:30.059261 #1] FATAL -- : 2024-11-11 13:06:30 ActiveRecord::StatementInvalid (Mysql2::Error: Incorrect TIMESTAMP value: '-190132431040868585787372620777824170090951749712709896695223-04-18 17:27:30.056272'):
2024-11-11 13:06:30
2024-11-11 13:06:30 app/models/user.rb:478:in `verify_session_token'
2024-11-11 13:06:30 app/controllers/application_controller.rb:86:in `session_expired?'
2024-11-11 13:06:30 app/controllers/application_controller.rb:76:in `session_expiration'
```

Workaround

If this issue occurs, you can restore normal functionality by modifying the corresponding setting values in the database.

Example of setting values:

```
mysql> select * from settings where name LIKE 'session%';
+----+-----+-----+-----+
| id | name                | value | updated_on |
+----+-----+-----+-----+
| 11 | session_lifetime    | 525600 | 2024-11-11 02:34:56 |
| 12 | session_timeout     | 2880   | 2024-11-11 02:34:56 |
+----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Remarks

Likewise, for parameters such as “time_entry_list_defaults” and “issue_list_default_columns” in the administration menu’s settings, it is possible to set invalid values in the same manner. However, unlike the authentication issue described in this report, Redmine itself does not become unusable; the only problem is that a 500 Internal Server Error occurs upon opening the settings menu.

History

#1 - 2025-01-07 10:28 - Go MAEDA

Thank you for reporting the issue.

The range supported by MySQL's DATETIME type is '1000-01-01 00:00:00' to '9999-12-31 23:59:59'. In the User.verify_session_token method, if the result of Setting.session_lifetime.to_i.minutes.ago falls before '1000-01-01 00:00:00', an error occurs as you pointed out.

To address this, it may be necessary to implement a validation in the Setting model or take other corrective measures.

#2 - 2025-01-07 10:58 - Go MAEDA

Here is one possible solution to address this issue. In User.verify_session_token, if Setting.session_lifetime exceeds one year, the value is replaced with one year instead of the actual setting.

```
diff --git a/app/models/user.rb b/app/models/user.rb
index 4ce63f809..3ac5ef1fb 100644
--- a/app/models/user.rb
+++ b/app/models/user.rb
@@ -463,10 +463,10 @@ class User < Principal
   scope = Token.where(:user_id => user_id, :value => token.to_s, :action => 'session')
   if Setting.session_lifetime?
-     scope = scope.where("created_on > ?", Setting.session_lifetime.to_i.minutes.ago)
+     scope = scope.where("created_on > ?", [[0, Setting.session_lifetime.to_i].max.minutes, 1.year].min.ago)
   end
   if Setting.session_timeout?
-     scope = scope.where("updated_on > ?", Setting.session_timeout.to_i.minutes.ago)
+     scope = scope.where("updated_on > ?", [[0, Setting.session_timeout.to_i].max.minutes, 48.hours].min.ago)
   )
   end
   last_updated = scope.maximum(:updated_on)
   if last_updated.nil?
```

A similar approach is also used in Attachment#thumbnail. If Setting.thumbnails_size exceeds 800, it is capped at 800.

```
# app/models/attachment.rb

def thumbnail(options={})
  if thumbnailable? && readable?
    size = options[:size].to_i
    if size > 0
      # Limit the number of thumbnails per image
      size = (size / 50.0).ceil * 50
      # Maximum thumbnail size
      size = 800 if size > 800
    else
      size = Setting.thumbnails_size.to_i
    end
    size = 100 unless size > 0
  end
end
```

#3 - 2025-01-08 01:41 - Keiji Aita

Go MAEDA wrote in [#note-1](#):

To address this, it may be necessary to implement a validation in the Setting model or take other corrective measures.

I appreciate your comment.

Please consider whether a fix is necessary going forward. Personally, I believe the probability of this issue occurring is quite low, as it requires an intentional request to be sent under administrator privileges.