Redmine - Defect #42875

"Page not found" error when saving workflows with many statuses on Rack >= 3.1.14

2025-06-16 04:22 - Go MAEDA

Status:	New		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:	Issues workflow		**Estimated time:**	0.00 hour
Target version:	5.0.13			
Resolution:			**Affected version:**	

Description

When using Rack 3.1.14 or later, saving a workflow that contains many statuses results in a "Page not found" error, and the workflow is not saved.

Steps to reproduce:

1. Update Rack to the latest version:

```
bundle update rack
```

2. Load default fixtures:

```
bin/rake db:fixtures:load
```

3. Add 40 issue statuses:

```
bin/rails r '40.times {|i| IssueStatus.create!(name: i.to_s)}'
```

4. Open the workflow edit page (/workflows/edit) in a browser.

5. Uncheck "Only display statuses that are used by this tracker", and click the "Edit" button.

6. Without making any changes, click the "Save" button.

-> A 404 Page Not Found error occurs, and the workflow is not saved.

Error message in the log:

```
Started POST "/workflows/update" for 127.0.0.1 at 2025-06-16 10:47:24 +0900

ActionController::RoutingError (No route matches [POST] "/workflows/update"):
```

Cause:

Starting in Rack 3.1.14, a security fix for [CVE-2025-46727](#) added a limit on the number of query parameters allowed in a request ([default: 4096](#)).

The workflow edit page generates a large number of parameters depending on the number of statuses. The number of parameters grows quadratically with the number of statuses.

For example, in Redmine 6.0, the number of parameters when all checkboxes are shown and checked can be calculated as:

```
6 * IssueStatus.count ** 2 - 4 * IssueStatus.count + 5
```

- With 27 statuses: 4271 parameters (exceeds 4096 limit)
- With 26 statuses: 3957 parameters (just under the limit)

So, the form data is silently discarded by Rack, and the routing fails.

In practice, not all checkboxes are checked, so the limit may be reached with more than 27 statuses — but 40 is enough to reproduce the issue reliably.

Possible workaround:

Increase the parameter limit by setting the RACK_QUERY_PARSER_PARAMS_LIMIT environment variable in config/boot.rb.

For example, the following change increases the limit to 65536, which allows handling workflows with up to 104 issue statuses (6 * 104 ** 2 - 4 * 104 + 5 => 64485) when all transitions are enabled.

```
diff --git a/config/boot.rb b/config/boot.rb
index 7479b5aff..570830c34 100644
--- a/config/boot.rb
+++ b/config/boot.rb
@@ -1,5 +1,7 @@
 # frozen_string_literal: true

+ENV['RACK_QUERY_PARSER_PARAMS_LIMIT'] = '65536'
+
 # Set up gems listed in the Gemfile.
 ENV['BUNDLE_GEMFILE'] ||= File.expand_path('../../Gemfile', __FILE__)
```

History

#1 - 2025-06-16 04:23 - Go MAEDA

- Description updated

#2 - 2025-06-16 12:05 - Go MAEDA

- File 42875.patch added

Here is the patch to fix this issue.

#3 - 2025-06-17 01:30 - Go MAEDA

- Target version changed from Candidate for next minor release to 5.0.13

Setting the target version to 5.0.13.

Files

| 42875.patch | 2.35 KB | 2025-06-16 | Go MAEDA |