redmineorg-copy202205 - Vote #78385

unchecked use of params[:query]

2022/05/09 18:15 - Admin Redmine

ステータス:	New	**開始日:**	2022/05/09
優先度:	通常	**期日:**	
担当者:		**進捗率:**	0%
カテゴリ:	Code cleanup/refactoring_30	**予定工数:**	0.00時間
対象バージョン:		**作業時間:**	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/25570	**status_id:**	1
category_id:	30	**tracker_id:**	3
version_id:	0	**plus1:**	0
issue_org_id:	25570	**affected_version:**	
author_id:	40856	**closed_on:**	
assigned_to_id:	0	**affected_version_id:**	
comments:	2		

説明

Requests to URLs like @/issues?query=12@ result in an application error because @query.rb@ expects @params[:query]@ to be a hash, which it isn't in this case.

I know that there are a more instances in the code base where we assume @params[:foo]@ to be a hash without explicit type checking. This patch does not attempt to address this problem on a global level. For whatever reason it happens to us quite often in this particular place, causing false alerts in server monitoring.

This patch adds an explicit type check to resolve this @params[:query]@ case. As a bonus it makes the following 4 lines where values are taken out of the hash a bit nicer.

In general I think it would be preferably to raise an error of the 4xx class in such cases. Are there any plans to make use of Rails' Strong Parameters feature in the future? I think using these permit / require calls on parameters in controllers would catch such wrong types early and would lead to responses with a more appropriate error code automatically.

journals

wherever I wrote 'to be a hash', I mean 'accessible like a hash'. It's an ActionController::Paramaters instance that we want.

As it turns out the patch causes problems when saving queries and thus I wouldn't recommend applying this.

履歴

#1 - 2022/05/10 17:06 - Admin Redmine

- カテゴリ を Code cleanup/refactoring_30 にセット