

## Ruby master - Feature #12399

### Restricted, safe version of `Kernel#eval`

05/19/2016 03:25 PM - sawa (Tsuyoshi Sawada)

|   |          |
|---|----------|
| <b>Status:</b>  | Feedback |
| <b>Priority:</b>  | Normal   |
| <b>Assignee:</b>  |          |
| <b>Target version:</b>  |          |
| <b>Description</b>  |          |
| Kernel#eval is convenient, but sometimes, it can be a security risk, and often people crazily react against using it even when it is not dangerous.   |          |
| I propose to have a restricted version of eval, which can interpret Ruby literals, but whenever there is constant assignment, variable assignment, method call, or method definition, it raises an error.                           |          |
| It can be used to safely accept parameters given as a string. One example use is, parameter interpretation of command line option parser can be easily be done under the assumption that the parameter is given as Ruby expression. |          |

### History

#### #1 - 05/20/2016 02:25 AM - shyouhei (Shyouhei Urabe)

I doubt if such thing could be more useful than JSON or YAML... Both are more widely adopted to non-ruby CLI users like system admins.

#### #2 - 05/20/2016 10:11 AM - naruse (Yui NARUSE)

It should be done as a gem.

#### #3 - 06/24/2016 06:51 AM - hsbt (Hiroshi SHIBATA)

- Status changed from Open to Feedback

#### #4 - 06/13/2017 12:24 AM - shevegen (Robert A. Heiler)

I am neutral on this; I can see that it can be useful. Not sure if matz wants to have it though, I guess there is a reason that "eval" is just one letter away from "evil". :D

I wanted to add only one thing though - shyouhei gave the valid comment that JSON and YAML are more widely adopted and used, but I wanted to say that although for most people, it may not always be interchangeable. I give you an example that may be rare, and unusual - no problem, I am not saying that it is valid for many, just one example of a slight difference.

YAML files have to be valid UTF-8 I think and perhaps UTF-16 or something. My yaml files are mostly still invalid (I am lazy, I know). I use the old syck gem very happily which works just fine - tenderlove and others make occasional updates to keep syck going, which is very nice. Hiroshi Shibata is also one of the maintainers of the syck gem. :)

Anyway - I sometimes break stuff in very unusual, dumb ways. And then I may have some problem e. g. that my yaml files do not work but I also can not get syck to install, because "gem" requires openssl. I had this problem just today and yesterday when I was experimenting with another openssl version.

Anyway - during the time of when gem was not working, ruby itself would still work, e. g. I could use `irb` and such. So in the above example, and please again, keep in mind, my example is dumb, partially contrived and not typical for many other people :), in that example, the ruby internal eval variant would still work whereas the yaml variant would not work.

This is not a good example, but my main point is just that the two, e. g. eval on ruby core itself, or yaml, are not fully interchangeable.

That does not mean that I am in favour or disfavour of the suggestion by Tsuyoshi Sawada by the way - my main point was just to say that it is not completely the same whether it would be yaml/json/eval. Although it may indeed be that the use case is possibly too limited ... it has been quite a long time since I last used eval() (I mean eval() itself ... I use instance\_eval a lot). Sorry for the length of my reply here.