Ruby - Feature #19630

[RFC] Deprecate `Kernel#open("|command-here")` due to frequent security issues

05/05/2023 11:12 PM - postmodern (Hal Brodigan)

Status:	Closed	
Priority:	Normal	
Assignee:		
Target version:		

Description

Kernel.open() is the source of numerous 1 security 2 issues 3, due to the fact that it can be used to execute commands if given a String argument of the form "|command-here". However, in most uses of Kernel.open() the developer appears to either want to open a local file, or if 'open-uri' was explicitly required open a remote URI. We should deprecate calling Kernel.open() with a "|command-here" style arguments, with a warning message instructing the developer to use IO.popen() instead. Eventually, support for Kernel.open("|command-here") could be removed completely, in favor of having the developer explicitly call IO.popen() or URI.open().

Related issues:

Related to Ruby - Misc #15893: open-uri: URI.open status	**Closed**
Related to Ruby - Feature #19723: [RFC] Deprecate/disallow passing `"|command...	**Closed**

Associated revisions

Revision d2343368ab7e270118ea6baa9c6418bfed83135c - 08/10/2023 12:38 AM - mdalessio (Mike Dalessio)

Deprecate Kernel#open and IO support for subprocess creation/forking

Deprecate Kernel#open and IO support for subprocess creation and
forking. This deprecates subprocess creation and forking in

- Kernel#open
- URI.open
- IO.binread
- IO.foreach
- IO.readlines
- IO.read
- IO.write

This behavior is slated to be removed in Ruby 4.0

[Feature #19630]

Revision d2343368ab7e270118ea6baa9c6418bfed83135c - 08/10/2023 12:38 AM - mdalessio (Mike Dalessio)

Deprecate Kernel#open and IO support for subprocess creation/forking

Deprecate Kernel#open and IO support for subprocess creation and
forking. This deprecates subprocess creation and forking in

- Kernel#open
- URI.open
- IO.binread
- IO.foreach
- IO.readlines
- IO.read
- IO.write

This behavior is slated to be removed in Ruby 4.0

[Feature #19630]

Revision d2343368 - 08/10/2023 12:38 AM - mdalessio (Mike Dalessio)

Deprecate Kernel#open and IO support for subprocess creation/forking

Deprecate Kernel#open and IO support for subprocess creation and
forking. This deprecates subprocess creation and forking in

- Kernel#open

- URI.open
- IO.binread
- IO.foreach
- IO.readlines
- IO.read
- IO.write

This behavior is slated to be removed in Ruby 4.0

[Feature #19630]

Revision e1b65e5f918744a2f2615feaa4ae39e1fb485651 - 11/30/2023 10:05 AM - mame (Yusuke Endoh)

Add NEWS entry for the deprecation of subprocess creation/forking

[Feature #19630]

Revision e1b65e5f918744a2f2615feaa4ae39e1fb485651 - 11/30/2023 10:05 AM - mame (Yusuke Endoh)

Add NEWS entry for the deprecation of subprocess creation/forking

[Feature #19630]

Revision e1b65e5f - 11/30/2023 10:05 AM - mame (Yusuke Endoh)

Add NEWS entry for the deprecation of subprocess creation/forking

[Feature #19630]

History

#1 - 05/06/2023 12:14 AM - postmodern (Hal Brodigan)

A more complete list of the CVEs related to Kernel.open:

- CVE-2017-17405 (ruby, net-ftp)
- CVE-2017-17790 (ruby, resolv)
- CVE-2019-10780 (bibtex-ruby)
- CVE-2021-21289 (mechanize)
- CVE-2019-5477 (nokogiri)
- CVE-2021-31799 (rdoc)
- CVE-2019-5477 (rexical)

#2 - 05/06/2023 01:40 PM - mdalessio (Mike Dalessio)

I enthusiastically support this suggestion. This is something that even experienced Ruby developers frequently forget about. I think it would be wise to evolve this API towards being "secure by default", even if that means forcing users to be explicit about the class or module.

#3 - 05/06/2023 09:41 PM - byroot (Jean Boussier)

- Related to Misc #15893: open-uri: URI.open status added

#4 - 05/24/2023 06:52 AM - hsbt (Hiroshi SHIBATA)

This proposal make sense to me. But I'm not sure how impact existing code for this incompatibility.

Do you have any deprecated process for this?

#5 - 06/07/2023 01:39 PM - mdalessio (Mike Dalessio)

@hsbt (Hiroshi SHIBATA) Because this functionality has existed in Ruby for such a long time, maybe we should target the next major release for removal of this functionality, and for now just print a deprecation warning.

@postmodern Just to clarify, you're only suggesting deprecating this in Kernel#open. It's also possible for commands to be injected into:

- IO.binread
- IO.foreach
- IO.readlines
- IO.read
- IO.write

but my understanding is that you're proposing to leave these methods alone, is that correct?

If noone has objections, I'll create a pull request so we have something concrete to discuss.

#6 - 06/07/2023 01:45 PM - byroot (Jean Boussier)

> for now just print a deprecation warning.

My worry is that since deprecation warnings are disabled by default, many people might not notice.

Recent examples show that things like File.exists? was deprecated for a decade, and some people were still surprised by its removal.

I know it's a distinct issue, but it impacts this one.

#7 - 06/07/2023 02:09 PM - mdalessio (Mike Dalessio)

I've created https://github.com/ruby/ruby/pull/7915 for review.

#8 - 06/08/2023 03:54 AM - postmodern (Hal Brodigan)

@mdalessio (Mike Dalessio) (Mike Dalessio) wrote in #note-5:

> @hsbt (Hiroshi SHIBATA) Because this functionality has existed in Ruby for such a long time, maybe we should target the next major release for removal of this functionality, and for now just print a deprecation warning.
>
> @postmodern Just to clarify, you're only suggesting deprecating this in Kernel#open. It's also possible for commands to be injected into:
>
> - IO.binread
> - IO.foreach
> - IO.readlines
> - IO.read
> - IO.write
>
> but my understanding is that you're proposing to leave these methods alone, is that correct?
>
> If noone has objections, I'll create a pull request so we have something concrete to discuss.

I was unaware that these methods can accept |command style inputs. Based on the stdlib documentation, the first argument is called name and the examples show reading from testfile, which implies to me they should only read from files. We could at first deprecate Kernel.open and see how much impact it has on users complaining about deprecation warnings, or we could start with the other IO methods?

#9 - 06/08/2023 12:17 PM - Eregon (Benoit Daloze)

IIRC IO methods all have an equivalent under File, and those do not accept pipes.
So e.g. RuboCop warns about them and suggest to use File.some_method instead:
https://www.rubydoc.info/gems/rubocop/RuboCop/Cop/Security/IoMethods
And there is already a cop too for Kernel#open it seems: https://www.rubydoc.info/gems/rubocop/RuboCop/Cop/Security/Open

But I agree for security reasons I think it makes sense to deprecate them in Ruby too, not everyone uses RuboCop or these cops in particular.

#10 - 06/09/2023 11:58 AM - nobu (Nobuyoshi Nakada)

postmodern (Hal Brodigan) wrote in #note-8:

> I was unaware that these methods can accept |command style inputs. Based on the stdlib documentation, the first argument is called name and the examples show reading from testfile, which implies to me they should only read from files. We could at first deprecate Kernel.open and see how much impact it has on users complaining about deprecation warnings, or we could start with the other IO methods?

I'm against deprecating pipe in IO methods.
It is intentionally kept quiet, unlike File.

#11 - 06/11/2023 04:35 PM - mdalessio (Mike Dalessio)

If we all agree that deprecating this behavior in Kernel#open is a good idea, is there any objection to something like https://github.com/ruby/ruby/pull/7915 ?

@byroot (Jean Boussier) I agree with your concerns about people ignoring deprecation warnings, but I don't think that's a good reason to stop deprecating behavior that we all agree should be deprecated.

#12 - 06/12/2023 06:47 AM - kosaki (Motohiro KOSAKI)

- Related to Feature #19723: [RFC] Deprecate/disallow passing `"|command..." values to open-uri's URI.open() method added

#13 - 06/12/2023 07:36 AM - byroot (Jean Boussier)

> is there any objection

Not from me, we should add this ticket to the next dev meeting.

However I feel like other IO methods (IO.binread, etc) should do the same otherwise it's a bit inconsistent.

#14 - 07/13/2023 08:47 AM - matz (Yukihiro Matsumoto)

OK. Probably we should remove pipe notation from all open methods, with deprecation process.

Matz.

#15 - 07/14/2023 12:04 AM - hsbt (Hiroshi SHIBATA)

@mdalessio (Mike Dalessio) Could you also deprecate the following methods in your pull request?

```
IO.binread
IO.foreach
IO.readlines
IO.read
IO.write
```

#16 - 07/14/2023 01:47 AM - mdalessio (Mike Dalessio)

@hsbt (Hiroshi SHIBATA) Yes, I'll update the pull request.

#17 - 07/14/2023 01:51 AM - mdalessio (Mike Dalessio)

@hsbt (Hiroshi SHIBATA) Do you think I should also deprecate pipe commends in URI.open as suggested in https://bugs.ruby-lang.org/issues/19723 ?

It seems like @matz (Yukihiro Matsumoto) may be encouraging this by saying "all open methods" above.

#18 - 07/14/2023 02:23 AM - hsbt (Hiroshi SHIBATA)

@mdalessio (Mike Dalessio) Deprecated URI.open is also accepted. We should deprecate it in same time.

#19 - 07/14/2023 05:45 AM - ko1 (Koichi Sasada)

memo:

> rb_warn_deprecated_to_remove("4.0", "Calling Kernel#open with a leading '|'", "IO.popen");

- warning until 4.0
- delete at 4.0

#20 - 07/14/2023 05:51 AM - nobu (Nobuyoshi Nakada)

- Subject changed from [RFC] Deprecate `Kernel.open("|command-here")` due to frequent security issues to [RFC] Deprecate `Kernel#open("|command-here")` due to frequent security issues

#21 - 07/24/2023 08:26 PM - mdalessio (Mike Dalessio)

The pull request is ready for review: https://github.com/ruby/ruby/pull/7915

#22 - 08/10/2023 12:38 AM - mdalessio (Mike Dalessio)

- Status changed from Open to Closed

Applied in changeset git|d2343368ab7e270118ea6baa9c6418bfed83135c.

Deprecate Kernel#open and IO support for subprocess creation/forking

Deprecate Kernel#open and IO support for subprocess creation and
forking. This deprecates subprocess creation and forking in

- Kernel#open
- URI.open
- IO.binread
- IO.foreach

- IO.readlines
- IO.read
- IO.write

This behavior is slated to be removed in Ruby 4.0

[Feature #19630]