Ruby - Misc #19178

How does CRuby handle CVE issues in stdlib gems which get patched?

12/03/2022 09:19 PM - Segaja (Andreas Schleifer)

Status:	Closed	
Priority:	Normal	
Assignee:	hsbt (Hiroshi SHIBATA)	

Description

If there is a CVE issue in one of the stdlibs (https://stdgems.org/) which gets patched, what is CRubys approach on how to push this critical fix to the users?

As far as I know stdlibs get only updated for the users if CRuby releases a new version. So will CRuby always release a new version if there is a critical fix an stdlib "needs" to be updated?

Related issues:

Related to Ruby - Feature #17684: Remove `--disable-gems` from release versio...	**Assigned**

History

#1 - 12/03/2022 09:53 PM - hsbt (Hiroshi SHIBATA)

> As far as I know stdlibs get only updated for the users if CRuby releases a new version. So will CRuby always release a new version if there is a critical fix an stdlib "needs" to be updated?

The all of stdlibs are maintained CRuby committers includes me. If the vulnerability is found and assign CVEs, We will release the new version of stdlibs at first. After that, we may release the new version of Ruby.

#2 - 12/03/2022 09:55 PM - Segaja (Andreas Schleifer)

hsbt (Hiroshi SHIBATA) wrote in #note-1:

> > As far as I know stdlibs get only updated for the users if CRuby releases a new version. So will CRuby always release a new version if there is a critical fix an stdlib "needs" to be updated?
>
> The all of stdlibs are maintained CRuby committers includes me. If the vulnerability is found and assign CVEs, We will release the new version of stdlibs at first. After that, we may release the new version of Ruby.

"may"? This sounds like sometimes CVEs are not considered "important" enough and do not warrant a new CRuby release. Or do I misunderstand this?

#3 - 12/03/2022 10:11 PM - austin (Austin Ziegler)

Segaja (Andreas Schleifer) wrote in #note-2:

> hsbt (Hiroshi SHIBATA) wrote in #note-1:
>
> > > As far as I know stdlibs get only updated for the users if CRuby releases a new version. So will CRuby always release a new version if there is a critical fix an stdlib "needs" to be updated?
> >
> > The all of stdlibs are maintained CRuby committers includes me. If the vulnerability is found and assign CVEs, We will release the new version of stdlibs at first. After that, we may release the new version of Ruby.
>
> "may"? This sounds like sometimes CVEs are not considered "important" enough and do not warrant a new CRuby release. Or do I misunderstand this?

Since the stdlib gems are able to be upgraded independently of Ruby, the need for *immediate* CRuby releases (or other Ruby release versions) is reduced.

#4 - 12/03/2022 10:14 PM - Segaja (Andreas Schleifer)

austin (Austin Ziegler) wrote in #note-3:

> Segaja (Andreas Schleifer) wrote in #note-2:
>
> > hsbt (Hiroshi SHIBATA) wrote in #note-1:
> >
> > > As far as I know stdlibs get only updated for the users if CRuby releases a new version. So will CRuby always release a new version if there is a critical fix an stdlib "needs" to be updated?
> >
> > The all of stdlibs are maintained CRuby committers includes me. If the vulnerability is found and assign CVEs, We will release the new version of stdlibs at first. After that, we may release the new version of Ruby.
>
> "may"? This sounds like sometimes CVEs are not considered "important" enough and do not warrant a new CRuby release. Or do I misunderstand this?

Since the stdlib gems are able to be upgraded independently of Ruby, the need for *immediate* CRuby releases (or other Ruby release versions) is reduced.

I think we have a naming difference here. I'm talking about the "default gems" as listed on https://stdgems.org/3.0.4/ for example for CRuby version 3.0.4. From all I understood these "default gems" are shipped with the main ruby version and can not be updated independently. So my question is how CVEs in those (for example the json default gem) will be handled.

#5 - 12/03/2022 10:20 PM - austin (Austin Ziegler)

Segaja (Andreas Schleifer) wrote in #note-4:

> austin (Austin Ziegler) wrote in #note-3:
>
> > "may"? This sounds like sometimes CVEs are not considered "important" enough and do not warrant a new CRuby release. Or do I misunderstand this?
>
> Since the stdlib gems are able to be upgraded independently of Ruby, the need for *immediate* CRuby releases (or other Ruby release versions) is reduced.
>
> I think we have a naming difference here. I'm talking about the "default gems" as listed on https://stdgems.org/3.0.4/ for example for CRuby version 3.0.4. From all I understood these "default gems" are shipped with the main ruby version and can not be updated independently. So my question is how CVEs in those (for example the json default gem) will be handled.

No, they can be upgraded independently.

```
$ ruby -rjson -e 'puts "JSON: #{JSON::VERSION}"'
JSON: 2.6.1
$ gem search '^json$'
*** REMOTE GEMS ***

json (2.6.2 ruby java, 1.1.5 x86-linux, 1.1.1 mswin32)
$ gem install json
Fetching json-2.6.2.gem
Building native extensions. This could take a while...
Successfully installed json-2.6.2
Parsing documentation for json-2.6.2
Installing ri documentation for json-2.6.2
Done installing documentation for json after 0 seconds
1 gem installed
$ ruby -rjson -e 'puts "JSON: #{JSON::VERSION}"'
JSON: 2.6.2
```

I'm currently using Ruby 3.1.

#6 - 12/03/2022 10:55 PM - hsbt (Hiroshi SHIBATA)

Austin, Thanks for your explanation for details.

We will update the all of bundled stdlibs(=default gems) at the release time of Ruby.

#7 - 12/03/2022 11:03 PM - Segaja (Andreas Schleifer)

austin (Austin Ziegler) wrote in #note-5:

> No, they can be upgraded independently.

That is interesting. The second sentence from https://rubyreferences.github.io/rubyref/stdlib/bundled.html says "Unlike standard library, these gems can be updated independently from Ruby itself."

But your way of updating "json" as a normal gem over the default gem means that whenever ruby is used with --disable-gems then the updated version is not used and thus a CVE could still be exposed.

Also doing such updates with a major version could break a lot of software which for example breaks with psych version 4.x as far as I know.

But I think my question remains: If I (as Arch maintainer) don't update (package the gem as new package) the gem, then how long will it take for a CVE to be fixed in the default ruby release?

#8 - 12/03/2022 11:08 PM - hsbt (Hiroshi SHIBATA)

> But your way of updating "json" as a normal gem over the default gem means that whenever ruby is used with --disable-gems then the updated version is not used and thus a CVE could still be exposed.

--disable-gems is only development option for debugging the Ruby binary. Do not use it for application or software development.

#9 - 12/04/2022 12:03 AM - ioquatix (Samuel Williams)

I've created an initial document, trying to distill some of the discussions here into a single place that downstream package maintainers can use as guidance.

https://github.com/ruby/ruby/pull/6856

Please help expand this document to clarify various points about how Ruby itself should be distributed and the process around it.

#10 - 12/04/2022 03:10 PM - graywolf (Gray Wolf)

hsbt (Hiroshi SHIBATA) wrote in #note-8:

> > But your way of updating "json" as a normal gem over the default gem means that whenever ruby is used with --disable-gems then the updated version is not used and thus a CVE could still be exposed.
>
> --disable-gems is only development option for debugging the Ruby binary. Do not use it for application or software development.

That is interesting. I know that I do use it in few places, usually for startup time reduction:

```
+$ time -p ruby -e 'puts 1'
1
real 0.06
user 0.04
sys 0.01
+$ time -p ruby --disable-all -e 'puts 1'
1
real 0.01
user 0.00
sys 0.01
```

Since that (based on you comment) does not seems like a right thing to do, are there other options to make ruby start up faster that are actually supported?

#11 - 12/13/2022 02:28 AM - hsbt (Hiroshi SHIBATA)

- Related to Feature #17684: Remove `--disable-gems` from release version of Ruby added

#12 - 12/13/2022 04:35 AM - hsbt (Hiroshi SHIBATA)

- Status changed from Open to Closed

@Segaja I'll close this because your first question was resolved now.

#13 - 12/13/2022 04:35 AM - hsbt (Hiroshi SHIBATA)

- Assignee set to hsbt (Hiroshi SHIBATA)

#14 - 12/13/2022 04:36 AM - nobu (Nobuyoshi Nakada)

Segaja (Andreas Schleifer) wrote in #note-7:

> That is interesting. The second sentence from https://rubyreferences.github.io/rubyref/stdlib/bundled.html says "Unlike standard library, these gems can be updated independently from Ruby itself."

This site seems pretty outdated.