Redmine - Defect #27356

Confusing statements concerning fixed versions on Security Advisories wiki page

2017-11-01 15:35 - Gregor Schmidt

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Gregor Schmidt	**% Done:**	0%
Category:	Website (redmine.org)	**Estimated time:**	0.00 hour
Target version:			
Resolution:	Fixed	**Affected version:**	

Description

The "fixed versions" for two old Rails related vulnerabilities listed on Security Advisories are very confusing.

Here's the relevant part of the table:

Critical	Ruby on Rails vulnerability (announcement)		All releases prior to 2.2.1 and 2.1.6	Fix for 1.4.7
Critical	Ruby on Rails vulnerability (announcement)		All releases prior to 2.2.1 and 2.1.6	1.4.7

I assume the proper 'Fixed Versions' would be:

Critical	Ruby on Rails vulnerability (announcement)		All releases prior to 2.2.1 and 2.1.6	2.2.1, 2.1.6, Fix for 1.4.7
Critical	Ruby on Rails vulnerability (announcement)		All releases prior to 2.2.1 and 2.1.6	2.2.1, 2.1.6, 1.4.7

Though I am not absolutely sure, if this change is correct - due to the confusing-ness of the current version.

History

#1 - 2017-11-24 09:39 - Mischa The Evil

I've spent about an hour and a half digging on this issue, yet I don't have a clear answer yet either. These were pretty messy times...

This involves:

- three to four CVE's:
  - CVE-2013-0155
    - https://groups.google.com/forum/#!topic/rubyonrails-security/t1WFuuQyavI
    - https://groups.google.com/forum/#!msg/rubyonrails-security/c7jT-EeN9eI/L0u4e87zYGMJ (updated to include 2.3.x)
    - CVE-2013-6417
      - https://groups.google.com/forum/#!topic/rubyonrails-security/niK4drpSHT4 (additional fix, never backported to 2.3.x)
  - CVE-2013-0156
    - https://groups.google.com/forum/#!topic/rubyonrails-security/61bkgvnSGTQ
  - CVE-2013-0333
    - https://groups.google.com/forum/#!topic/rubyonrails-security/1h2DR63ViGo
  - ~~CVE-2012-3464~~
    - https://groups.google.com/forum/#!msg/rubyonrails-security/kKGNeMrnmiY/r2yM7xy-G48J
- four Redmine releases:
  - 2.2.1, 2.1.6 and 1.4.6: Redmine 2.2.1, 2.1.6 and 1.4.6 security releases
  - 1.4.7: Redmine 1.4.7 security release
- one Redmine release hot fix
  - 1.4.7 with Rails 2.3.16 (for CVE-2013-0333): New Rails vulnerability affects Redmine 1.4.7
- three Rails updates:
  - 3.2.11
  - 2.3.16

    - 2.3.15
- ~~(possibly)~~ a manually backported fix for ~~CVE-2012-3464~~ CVE-2013-0155 in Redmine 1.4.7 [~~possibly~~ with an error in the code comments referring to CVE-2012-3464]:
    - r11197 and r11208

Updated by Mischa The Evil on 2017-11-28 to reflect latest findings.

#2 - 2017-11-27 02:18 - Mischa The Evil

- Assignee changed from Jean-Philippe Lang to Mischa The Evil

When it wasn't clear yet: I'm researching this issue. Almost done btw. Some last commit-history checks for both Rails and Redmine and wrapping up are remaining. Though, the issues with the current table values begin to be more clearly visible already... ;)

Results so far (~~and sorry upfront for the alignment, I'm copy pasting from temp. *notepad.exe* text file in ANSI; will fix it in the end~~ fixed):

```
ID   Severity   Details                                    Affected versions                       Fixed version
s        Redmine News link

1    Critical   RoR vulnerability (announcement[1])      All releases prior to 2.2.1 and 2.1.6   Fix for 1.4.7
         http://www.redmine.org/news/78 (New Rails vulnerability affects Redmine 1.4.7), 29-01-13
2    Critical   RoR vulnerability (announcement[2])      All releases prior to 2.2.1 and 2.1.6   1.4.7
         http://www.redmine.org/news/76 (Redmine 1.4.7 security release), 20-01-13
3    Critical   RoR vulnerability (announcement[3])      All prior releases                      2.2.1, 2.1.6,
 1.4.6   http://www.redmine.org/news/75 (Redmine 2.2.1, 2.1.6 and 1.4.6 security releases), 09-01-13

Notes:
1. https://groups.google.com/forum/?fromgroups=#!topic/rubyonrails-security/1h2DR63ViGo
   "Vulnerability in JSON Parser in Ruby on Rails 3.0 and 2.3", 28-01-13
   CVE-2013-0333, Affected Rails: 2.3.x, 3.0.x; Not Affected: 3.1.x, 3.2.x, applications using the yajl gem; F
ixed: 3.0.20, 2.3.16
2. https://groups.google.com/forum/#!msg/rubyonrails-security/c7jT-EeN9eI/L0u4e87zYGMJ
   "Updated Advisory: Unsafe Query Generation Risk in Ruby on Rails", 14-01-13
   CVE-2013-0155, Affected Rails: 2.x, 3.x; Not-Affected: None; Fixed: 3.2.11, 3.1.10, 3.0.19, -2.3.15- [+2.3.
16+]
   \-> Update of: https://groups.google.com/forum/#!topic/rubyonrails-security/t1WFuuQyavI
                  "Unsafe Query Generation Risk in Ruby on Rails (CVE-2013-0155)", 08-01-13
                  CVE-2013-0155, Affected Rails: 3.x; Not-Affected: 2.x; Fixed: 3.2.11, 3.1.10, 3.0.19
3. http://weblog.rubyonrails.org/2013/1/8/Rails-3-2-11-3-1-10-3-0-19-and-2-3-15-have-been-released/
   "[SEC][ANN] Rails 3.2.11, 3.1.10, 3.0.19, and 2.3.15 have been released!", 08-01-13
   CVE-2013-0155 & CVE-2013-0156[4]
4. https://groups.google.com/forum/#!topic/rubyonrails-security/61bkgvnSGTQ
   "Multiple vulnerabilities in parameter parsing in Action Pack (CVE-2013-0156)", 08-01-13
   CVE-2013-0156, Affected Rails: All; Not-Affected: None; Fixed: 3.2.11, 3.1.10, 3.0.19, 2.3.15
```

Will pickup & finish another day...

Updated by Mischa The Evil on 2017-11-28 to reflect latest findings.

#3 - 2017-11-28 04:57 - Mischa The Evil

Final results

Here are the final results of my research. I've already modified/updated the earlier posted bits of info.

Course of events:

The course of events in that January month of 2013 can best be represented within a table:

Events/state:	Date:	2.2-stable:	2.1-stable:	1.4-stable:
Then current releases	< 2013-01-08	2.2.0 (3.2.9)	2.1.5 (3.2.8)	1.4.5 (2.3.14)
CVE-2013-015[5|6]	2013-01-08	a	a	a
New releases	2013-01-09	2.2.1 (3.2.11)	2.1.6 (3.2.11)	1.4.6 (2.3.15)
CVE-2013-0155 rep.	2013-01-14..20	n/a	n/a	a
New releases	2013-01-20	-	-	1.4.7 (2.3.15 with sec. fix backport [r11197 & r11208])
CVE-2013-0333	2013-01-28	n/a	n/a	a
Release hot fix	2013-01-29	-	-	1.4.7-HotFix (2.3.16)

CVE-2013-6417	2013-12-03	n/a	n/a	a

Based on that info we can do some observations:

- O1: messy times... ;)
- O2: Jean-Philippe and Toshi responded swiftly with adequate resolutions :thumbsup:
- O3: A misleading (referring to unrelated CVE-2012-3464) code comment crept in along the way
- O4: Rails team left 2.3.x vulnerable to CVE-2013-0155 through CVE-2013-6417 for which the resolution was not backported to 2.3.x (anymore)

Suggestion what table should read:

Based on all the currently available information I'd suggest to modify the *three* related table rules to look like follows:

Severity	Details	External references	Affected versions	Fixed versions
Critical	Ruby on Rails vulnerability (announcement)		All releases prior to and including 1.4.7	Fix for 1.4.7
Critical	Ruby on Rails vulnerability (announcement)		All releases prior to 2.2.1 and 2.1.6, and 1.4.6	1.4.7
Critical	Ruby on Rails vulnerability (announcement)		All prior releases	2.2.1, 2.1.6, 1.4.6

What do you think?

#4 - 2017-11-28 04:58 - Mischa The Evil

- Status changed from New to Needs feedback

- Assignee changed from Mischa The Evil to Gregor Schmidt

#5 - 2017-11-28 11:58 - Gregor Schmidt

Thank you so much for your research. In your proposed update, the third entry convers CVE-2013-0155 and CVE-2013-0156. While the second line covers mainly CVE-2013-0155 for 2.3.x. This follows the time line, but I think it would be more comprehensive to follow the vulnerabilities in this case.

Critical	Ruby on Rails vulnerability (announcement)	CVE-2013-0333	All releases prior to and including 1.4.7	Fix for 1.4.7
Critical	Ruby on Rails vulnerability (announcement)	CVE-2013-0155	All prior releases	2.2.1, 2.1.6, 1.4.7
Critical	Ruby on Rails vulnerability (announcement)	CVE-2013-0156	All prior releases	2.2.1, 2.1.6, 1.4.6

What do you think, is this still accurate?

#6 - 2017-12-05 09:35 - Mischa The Evil

Gregor Schmidt wrote:

> Thank you so much for your research. In your proposed update, the third entry convers CVE-2013-0155 and CVE-2013-0156. While the second line covers mainly CVE-2013-0155 for 2.3.x. This follows the time line, but I think it would be more comprehensive to follow the vulnerabilities in this case.

I'd ok with that, but I always interpret these kind of lists as event lines (adding the date to each line automatically). It also follows the separate news items.

> <table snip>
>
> What do you think, is this still accurate?

It is still accurate enough for me. However, JPL or sec. team may think differently. I'd like to hear their opinion before I'd change the page.

Edit by Mischa The Evil on 2017-12-05: snip quoted table.

#7 - 2024-06-20 17:36 - Holger Just

- Status changed from Needs feedback to Closed

- Resolution set to Fixed

I have update the entries according to the proposed changes by Gregor #note-5 in revision 78 of Security_Advisories.

Thanks for your efforts to ensure accurate information, even though the final update took its sweet time in the end :)