Redmine - Defect #3351

Weak autologin token generation algorithm causes duplicate tokens

2009-05-13 10:56 - Alexander Pavlov

Status:	Closed	**Start date:**	2009-05-13
Priority:	Urgent	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Accounts / authentication	**Estimated time:**	0.00 hour
Target version:	0.8.4		
Resolution:	Fixed	**Affected version:**	

Description

After switching to mod_passenger we got 7 (seven!) duplicated autologin tokens within 2 weeks. It caused some changes have been made under wrong user account!

Looks like due to using of pseudo-random sequence generator two concurrent Ruby processes may use the same random seed (and as result the same random sequence).

At our instance we made quick fix - prepend random sequence with "#{user.id}_" and substring left 40 chars, however, I guess there may be better solution.

Associated revisions

Revision 2740 - 2009-05-13 18:54 - Jean-Philippe Lang

Use ActiveSupport::SecureRandom to generate tokens (#3351).

Revision 2741 - 2009-05-13 18:55 - Jean-Philippe Lang

Add token value uniqueness validation (#3351).

Revision 2742 - 2009-05-13 18:56 - Jean-Philippe Lang

Do not autologin if more that one token is found (#3351).

History

#1 - 2009-05-13 11:06 - Alexander Pavlov

Also, I suggest to deny login if search by autologin within Token table returned 2 or more records - it allows to prevent and troubleshot possible errors in future.

#2 - 2009-05-13 18:50 - Jean-Philippe Lang

- Status changed from New to Resolved

- Target version set to 0.8.4

- Resolution set to Fixed

I never experienced this issue but I've just committed the following fixes in r2740, r2741, r2742:

- ActiveSupport::SecureRandom is now used to generate tokens
- Added a validation on token uniqueness that will prevent 2 tokens with the same value from being saved
- Autologin is denied if more than one token is found

#3 - 2009-05-14 09:47 - Alexander Pavlov

> I never experienced

We suspect it is due to process forking which leads to random sequence seed **inherited** from parent process so two processes continue working with the same sequence.

> ActiveSupport::SecureRandom is now used to generate tokens

Thanks, it is what we were going to suggest!

#4 - 2009-05-14 09:58 - Alexander Pavlov

Small example from our developers

```
irb(main):004:0> rand(50)
=> 9
irb(main):005:0> fork { puts rand(50) }
37
=> 22831
irb(main):006:0> rand 50
=> 37
```

#5 - 2009-05-14 10:00 - Alexander Pavlov

Also, you could check your DB to ensure you have really never affected by this vulnerability

```
select value, count(*) from tokens group by value having count(*) > 1
```

#6 - 2009-05-14 18:25 - Jean-Philippe Lang

That's what I did when I said that I never experienced this issue.

#7 - 2009-05-15 09:03 - Alexander Pavlov

Jean-Philippe Lang wrote:

> That's what I did when I said that I never experienced this issue.

Probably you are not using mod_passenger. If you started several predefined processes (without mod_passenger) then random sequence had their own seeds and their own random sequences.

mod_passenger do forks and these inherit parent seed (see post #4) - it is key factor to reproduce problem.

#8 - 2009-05-17 11:04 - Jean-Philippe Lang

- Status changed from Resolved to Closed

Indeed, I'm using apache+mod_fcgid.
Fixes are backported in 0.8-stable branch in r2747.