Redmine - Feature #3096

Lock accounts after X failed attempts

2009-04-01 18:04 - Ben Blier

Status:	New	**Start date:**	2009-04-01
Priority:	High	**Due date:**	
Assignee:		**% Done:**	50%
Category:	Accounts / authentication	**Estimated time:**	0.00 hour
Target version:			
Resolution:			

Description

I believe Redmine should have the functionality available to put accounts in to a locked state after so many failed login attempts. The number of failed attempts should be configurable via the Administration panel. Notification to an administrator e-mail address that the account was locked is desired as well.

I am surprised this feature has not made it in to Redmine yet. Could this be something that makes it in to a 0.9 release? I plan on exposing my Redmine instance to more than just internal folk within the next 6mo-1yr. I do not want to give any external entity the ability to brute force my password.

Related issues:

Related to Redmine - Feature #3155: Password policy and secure logon procedure	**New**	**2009-04-10**

History

#1 - 2009-04-01 18:39 - Jens Goldhammer

+1

#2 - 2009-04-01 20:00 - Maxim Krušina

+ function to email admin/user about locking. Also account can be optionally unlocked after some (probably configurable) period, like 1 hour...

#3 - 2009-04-02 09:58 - Adam Kubica

+1 (failed attempts number should be configurable)

Automatic unlocking after some period might be security problem.

#4 - 2009-04-05 04:59 - Alexander J. Murmann

+1
I also think this might be very useful.
I just started working on a patch for this.

#5 - 2009-04-06 20:59 - Alexander J. Murmann

- % Done changed from 0 to 50

I am almost done with the patch but was wondering how accounts should be unlocked.
I can see the following alternatives:

1. After a timeout, as suggested earlier
2. Notification email contains a link that will unlock the account again
3. You have to deal with an admin outside the system and he has to manually unlock it
4. Go through "forgot password" and reset the password and when the password is reset the account will be unlocked.

I personally think that 2. would be best.

Any thoughts about this or other suggestions?

#6 - 2009-04-06 22:24 - Ben Blier

#4 isn't a viable option since my LDAP is read-only and I don't even know if "Forgot password" works with LDAP (probably not).

It would be best if the admin is given the option to configure #1, #2, or #3, but I'll take either #2 or #3.

#7 - 2009-04-13 05:25 - Alexander J. Murmann

- File login_attempts.diff added

I implemented solution 2. Although if there is need it should be very easy to add an option to use 3. in addition.

Attached is a patch that should allow the admin to define a number of allowed login attempts and the address of an admin.
If a user fails to login the flash-message will show how many logins are left. If none are left the flash tells so and the account gets locked. A mail informing the provided admin address will be send. The suer will also receive a mail telling him what happened and providing a link to reactivate the account.

However since I am a bad boy I didn't write unit tests yet. So there still might be something wrong. I will provide another patch which will include tests later this week.

#8 - 2009-04-15 01:10 - Eric Davis

Thanks Alexander. Once you add some unit tests I'll be able to take a closer look at applying this patch. From a quick glance User#authentication_failed could be cleaned up a bit. I see two calls to self.save! and no handling of their failure cases.

#9 - 2009-04-27 04:26 - Alexander J. Murmann

- File login_attempts.diff added

I added a unit test and changed the two 'save!'s to 'save' since I could not come up with a useful way to catch a failed save.

Please let me know if and how I can improve the patch further!

#10 - 2009-05-01 18:54 - Michael Litton

Great, I really need this.

#11 - 2009-06-15 23:40 - Ben Blier

I'm curious if anybody has been running this patch in their environment... What are your thoughts? Anything that could be improved?

#12 - 2011-04-02 14:00 - S Reid

Is this still the only method to lock accounts after failed retries ? Does it work with the current version of redmine ?

#13 - 2011-04-19 00:49 - Nuno Duarte

I believe this feature would improve a lot redmine security. Giving more confidence to me and my clients.

#14 - 2015-01-20 21:56 - @ go2null

duplicate of #3155

#15 - 2015-01-21 06:46 - Mischa The Evil

@ go2null wrote:

> duplicate of #3155

Not completely. #3155 is older than this issue and it is much more generic, while this issue is specific to one requested change. So I'll add a relation, but won't close this one as *duplicate*.

#16 - 2015-01-21 06:46 - Mischa The Evil

- Related to Feature #3155: Password policy and secure logon procedure added

#17 - 2015-03-11 09:48 - François Bélingard

+1

#18 - 2023-08-28 04:58 - Mizuki ISHIKAWA

- File login_attempts_v2.patch added

A patch for this feature is attached.

Features/Changes provided in this patch:

- Administrator set max_login_attempts (int) in the admin settings panel (default is nil).

- If a user exceeds this number of failed login attempts, their account will be locked.
- The timing for displaying l(:notice_account_locked) has been changed to prevent attackers from identifying the correct password.
- An email will be sent to the user when their account is locked.
- A security notification email will be sent to the admin when a user's account is locked.
- The lock can be lifted in one of the following ways:
  - An email sent to the locked user will contain a URL to unlock their account.
  - Admins can unlock the user from the user list.

Features not included in this patch:

- Auto-unlock feature after a certain period has elapsed (After this patch has been merged, it is better to address this in a other issue if necessary.).

I am not a security expert, so I would appreciate it if you could confirm that there are no issues with these specifications.

#19 - 2023-08-28 22:37 - Pavel Rosický

wow 14 years :) Unfortunately from my experience, there are two issues with this approach:
1/ someone could easily lock the whole system if he knows the login (which is usually public or simple to guess). You'll notice the attack, but it's a nightmare for the administrator.
2/ without required twofa for all users, it's not sufficient protection against spraying attacks
https://www.hypr.com/security-encyclopedia/password-spraying-attack

#20 - 2023-08-29 01:48 - Mizuki ISHIKAWA

Pavel Rosický wrote in #note-19:

> wow 14 years :) Unfortunately from my experience, there are two issues with this approach:
> 1/ someone could easily lock the whole system if he knows the login (which is usually public or simple to guess). You'll notice the attack, but it's a nightmare for the administrator.
> 2/ without required twofa for all users, it's not sufficient protection against spraying attacks
> https://www.hypr.com/security-encyclopedia/password-spraying-attack

Thank you for your feedback.
1 /
You're absolutely right that this approach could be exploited to lock out users, causing a headache for administrators.
In anticipation of this, the feature is disabled by default, allowing administrators to opt-in only if they find it suitable for their specific use-case.
If an account is locked, users can unlock themselves via a URL sent to their registered email. Additionally, the IP address of the user attempting the lockout is logged and sent to administrators, who could then potentially block the IP at the server level using tools like Apache.

2 /
The patch I submitted does not include a "reset login attempts over time" feature. This means that even if an attacker uses a password spraying attack technique with long intervals between attempts, they will still be limited by the maximum number of allowed attempts. Even if a "reset login attempts over time" feature were to be implemented, it would not make the situation worse for instances of Redmine where two-factor authentication is not enabled, as attackers could already make unlimited attempts under the current system.

Many Thanks,

#21 - 2023-08-29 17:19 - Pavel Rosický

Redmine should have some kind of default protection against these types of attacks and your implementation is fine, at least it protects against cracking passwords in the background without any notification...

but in my opinion, a rate limiter + captcha would be a better option. Administrators could still be informed about the attack, but in reality, it takes some time for administrators to take action. A malicious actor could make the system unusable without much effort (locking users over and over) until the administrator bans the IP and then he can start again from a different IP. The feature protects against stealing passwords, but on the other hand, it opens a new opportunity, to shut the application down / block regular users from using it.

I'm also not a security expert, so I'm open to other opinions :)

Files

login_attempts.diff	12.9 KB	2009-04-13	Alexander J. Murmann
login_attempts.diff	13.8 KB	2009-04-27	Alexander J. Murmann
login_attempts_v2.patch	16.1 KB	2023-08-28	Mizuki ISHIKAWA