Redmine - Feature #1913

LDAP - authenticate as user

2008-09-16 15:02 - Adi Kriegisch

Status:	Closed	**Start date:**	2008-09-16
Priority:	Normal	**Due date:**	
Assignee:	Jean-Philippe Lang	**% Done:**	70%
Category:	LDAP	**Estimated time:**	0.00 hour
Target version:	1.4.0		
Resolution:			

Description

The attached patch allows to bind to the ldap server as the user logging in (instead of either anonymous or specific admin account).
The idea behind the patch is quite simple: When configuring an LDAP source one may or may not specify an "Account". If no account is specified, Redmine will bind anonymously. If the account is specified Redmine binds as that user.
The patch introduces a third state: bind as user. When the account is specified as (for example) "uid=$login,ou=people,dc=example,dc=com" $login is replaced with the login name and the password given by the user is used.
Therefor there is no need to have an LDAP directory that is readable by anonymous bind and there is no need to have a password saved into the database.
I tried to make the patch as unintrusive as possible while completely being compatible to the way things are now.

Related issues:

Related to Redmine - Defect #3253: LDAP Auth : Alias Dereference	**New**	**2009-04-28**
Has duplicate Redmine - Feature #10375: LDAP: Account Name binding should be ...	**Closed**	

Associated revisions

Revision 9241 - 2012-03-17 13:09 - Jean-Philippe Lang

LDAP: adds the ability to bind with user's account (#1913).

Revision 9242 - 2012-03-17 13:18 - Jean-Philippe Lang

Fixed test names (#1913).

Revision 9243 - 2012-03-17 13:19 - Jean-Philippe Lang

Typo (#1913).

History

#1 - 2008-10-07 13:23 - Markus Peter

I applied this patch on a development checkout (revision 1901) and it works fine!
It took me a while to figure out the ldap settings.

In the Account field, I entered
[domain]\$login
and in Base DN
CN=users,DC=people,DC=example,DC=com
This finally worked for me, no password stored anywhere.

#2 - 2008-11-14 19:00 - Martin Bächtold

This patch also worked for me, thank you very much.

#3 - 2009-02-20 09:58 - Jérémie Delaitre

Will this patch be included into the core ?

#4 - 2009-02-20 22:57 - Eric Davis

Could someone update the patch to include a few test cases?

#5 - 2009-05-01 21:24 - Daniel Marczisovszky

I've created a patch for alias dereferencing (#3253) and I included your patch. It also adds options for custom search filtering that was influenced by your patch, it uses the same syntax. Could you please try it out?

#6 - 2009-08-21 15:34 - Adi Kriegisch

- File Redmine-ldap-as-user.diff added

To make this patch usable with Apache authentication against Redmine for SVN access (ie. all the stuff that is located in redmine/extra/svn) it is necessary to patch Redmine.pm to as well understand how to treat $login in config.
btw. Arnaud Martel did a great job in enhancing Redmine.pm. See #3712.

#7 - 2010-02-17 22:40 - Eric Davis

- Category changed from Accounts / authentication to LDAP

Can someone post a simple OpenLDAP configuration that removes the anonymous binding so I can test this? I've been working in the LDAP code recently and would like to apply this.

#8 - 2010-02-18 09:31 - Adi Kriegisch

Eric Davis wrote:

> Can someone post a simple OpenLDAP configuration that removes the anonymous binding so I can test this? I've been working in the LDAP code recently and would like to apply this.

Cool! Looking forward to you patch!

Assuming you've got a working LDAP server adding an access control list is the only thing required (if you already have some watch out for the correct sorting -- they are processed top to bottom and first match stops further processing).
access to attrs=userPassword by dn.base="uid=root,dc=mydomain,dc=com" write
_ _ by self write
_ _ by anonymous auth
(_ _ == just spaces)
should be quite self explaining: in contrary to SQL-queries LDAP just hides attributes if you're not allowed to see them ("read" permission) so you will still be able to get a list of all users. Just the "userPassword" field is hidden. "anonymous" connections may just use this attribute to authenticate whereas the authenticated user himself may read and change his password and root -- as usual -- may do anything!

In real life you might add other ACL as well and you might want to add the "ssf" parameter as well (security strenght factor that specifies if SSL/TLS is required for access to a certain attribute).

Hope this helps! Feel free to contact me if you need any further assistance!

#9 - 2010-02-18 09:35 - Felix Schäfer

This is a "light" version of what we have, so it's kinda untested in this form, but it should give read or read/write permission to some attributes only to some users.

```
to attrs=userPassword,shadowLastChange
  by self write
  by anonymous auth
  by * none

to dn.one="ou=People,dc=example,dc=com" attrs=uid   # This assumes a flat user directory, change one to childr
en to extend it to any child of ou=People at any depth
  by self read
  by users read   # this ensures that any user in the LDAP can search/read for uids to enable the search for a
 cn based on the uid, redmine obviously needs a "user" somewhere in the LDAP tree for that to work
  by * none

to dn.children="ou=People,dc=example,dc=com" attrs=givenName,sn,mail   # We have more attributes here, but tha
t should be the only three redmine needs atm
  by self write
  by users read   # same comment as above
  by * none

to dn.base=""
  by * read

to *
  by users read   # not sure how much this part is needed, as it gives a blanket read access to all users in t
he LDAP, but should be ok for testing the need for user credentials to read the LDAP
  by * none
```

I also removed all our "extra" admin accesses and stuff, catch me on IRC if you need more info/help with that.

#10 - 2010-02-18 09:39 - Felix Schäfer

By the way, you won't need a "full" user for the redmine to access the LDAP, "only" an LDAP object with a password, our "users" for app access export to something like that:

```
dn: cn=Redmine,ou=Apps,ou=System,dc=example,dc=com
cn: Redmine
objectClass: namedObject
objectClass: top
objectClass: simpleSecurityObject
userPassword: {SSHA}somefunnyhash
```

#11 - 2010-02-21 22:14 - Antoine Beaupré

- File Redmine-app-models-auth_source_ldap-0.9.1-2.diff added

- % Done changed from 0 to 50

So here's a patch that applies to the Debian 0.9.1 package that improves a bit on what's already here in that it avoids doing a second bind if we're using the introduced "bind as user" setting.

#12 - 2010-02-21 22:46 - Antoine Beaupré

- File 1913_redmine_bind_as_user.diff added

- % Done changed from 50 to 70

... and here's an untested patch for current head. it refactors get_user_dn() to use an internal ldap_con to avoid binding twice with the ldap server when using user-level bindings.

i think the UI could still need some love to explain the $login hack, but in the meantime this makes more sense in the LDAP world...

#13 - 2010-02-24 14:26 - Bernhard Furtmueller

I had the very same problem, tried to search but didn´t find this solution in the past.
Therefore a did the "reinvent the wheel" approach and came up with my patch
which introduces a new domain field.

See forum entry here:
http://www.redmine.org/boards/1/topics/11119

And the patch (which works at least against active directory):
http://www.redmine.org/attachments/3176/patch0.patch

I´ll try the patch (is it supposed to work against active directory?)
attached to this tracker asap and will report.

br,
bernhard

#14 - 2010-04-21 00:44 - Markus Peter

- File Bind_as_user_LDAP.diff added

Antoine Beaupré wrote:

> ... and here's an untested patch for current head. it refactors get_user_dn() to use an internal ldap_con to avoid binding twice with the ldap server when using user-level bindings.

The patch did throw an error:

```
NoMethodError (undefined method `ldap_con=' for #<AuthSourceLdap:0x8c22f74>):
  app/models/auth_source_ldap.rb:38:in `authenticate'
  app/models/user.rb:105:in `try_to_login'
  app/controllers/account_controller.rb:147:in `password_authentication'
  app/controllers/account_controller.rb:142:in `authenticate_user'
  app/controllers/account_controller.rb:30:in `login'
```

Chances are I messed up something...

I included a patch which works with the current head, though it still calls initialize_ldap_con twice, once in authenticate and once in authenticate_dn.

#15 - 2010-04-21 08:00 - Felix Schäfer

Markus Peter wrote:

> I included a patch which works with the current head, though it still calls initialize_ldap_con twice, once in authenticate and once in authenticate_dn.

There are environments in which you might need to first bind as the "redmine user" to the LDAP to determine the DN corresponding to a certain login as not all LDAP setups support searching for login as anonymous.

#16 - 2010-04-21 09:06 - Markus Peter

Felix Schäfer wrote:

> There are environments in which you might need to first bind as the "redmine user" to the LDAP to determine the DN corresponding to a certain login as not all LDAP setups support searching for login as anonymous.

Our AD does not support serching for logins as anonymous, that's why we use this patch to connect with the login/password supplied by the user and the parameters provided in the LDAP configuration *without* having to enter a user/pwd for each LDAP config.

Not sure whether this works with self-registration, though.

Antoine's patch is certainly better, but unless I messed up something there may be missing something to make it work.

#17 - 2010-04-21 22:20 - Bernhard Furtmueller

Had the same problem, so I´ll post my approach here. This is against trunk @3625
This adds a domain field which will be prefixed to the userid.
It also allows self-registration.
HTH,
bernhard

```
commit c6b87839849899fb2c24fde1533224f60818074e
Author: Bernhard Furtmueller <bernhard.furtmueller@hilti.com>
Date:   Tue Mar 30 13:37:14 2010 +0000

    adding a domain field in order to allow direct active directory
    authentication without requiring a read only ads user.

    forward port of 0b9ee54dafe21140bf694bf968431633d4ec09b5
    only with lang en and de

diff --git a/app/models/auth_source_ldap.rb b/app/models/auth_source_ldap.rb
index d2a7e70..a7bb7ba 100644
--- a/app/models/auth_source_ldap.rb
+++ b/app/models/auth_source_ldap.rb
@@ -21,7 +21,7 @@ require 'iconv'
 class AuthSourceLdap < AuthSource
   validates_presence_of :host, :port, :attr_login
   validates_length_of :name, :host, :account_password, :maximum => 60, :allow_nil => true
-  validates_length_of :account, :base_dn, :maximum => 255, :allow_nil => true
+  validates_length_of :account, :domain, :base_dn, :maximum => 255, :allow_nil => true
   validates_length_of :attr_login, :attr_firstname, :attr_lastname, :attr_mail, :maximum => 30, :allow_nil =>
 true
   validates_numericality_of :port, :only_integer => true

@@ -33,7 +33,7 @@ class AuthSourceLdap < AuthSource

   def authenticate(login, password)
     return nil if login.blank? || password.blank?
-    attrs = get_user_dn(login)
+    attrs = get_user_dn(login,password)

     if attrs && attrs[:dn] && authenticate_dn(attrs[:dn], password)
       logger.debug "Authentication successful for '#{login}'" if logger && logger.debug?
@@ -100,8 +100,10 @@ class AuthSourceLdap < AuthSource
   end

   # Get the user's dn and any attributes for them, given their login
-  def get_user_dn(login)
-    ldap_con = initialize_ldap_con(self.account, self.account_password)
+  def get_user_dn(login,password)
+    #ldap_con = initialize_ldap_con(self.account, self.account_password)
+    domain.blank? ? ldap_con = initialize_ldap_con(self.account, self.account_password) : ldap_con = initiali
ze_ldap_con(domain + "\\" + login, password);
```

```
+
     login_filter = Net::LDAP::Filter.eq( self.attr_login, login )
     object_filter = Net::LDAP::Filter.eq( "objectClass", "*" )
     attrs = {}
diff --git a/app/views/auth_sources/_form.rhtml b/app/views/auth_sources/_form.rhtml
index 9ffffaf..f023bce 100644
--- a/app/views/auth_sources/_form.rhtml
+++ b/app/views/auth_sources/_form.rhtml
@@ -11,6 +11,9 @@
 <p><label for="auth_source_port"><%=l(:field_port)%> <span class="required">*</span></label>
 <%= text_field 'auth_source', 'port', :size => 6 %> <%= check_box 'auth_source', 'tls'  %> LDAPS</p>

+<p><label for="auth_source_domain"><%=l(:field_domain)%></label>
+<%= text_field 'auth_source', 'domain'  %></p>
+
 <p><label for="auth_source_account"><%=l(:field_account)%></label>
 <%= text_field 'auth_source', 'account'  %></p>

diff --git a/config/locales/de.yml b/config/locales/de.yml
index 982452e..d79c20a 100644
--- a/config/locales/de.yml
+++ b/config/locales/de.yml
@@ -268,6 +268,7 @@ de:
   field_port: Port
   field_account: Konto
   field_base_dn: Base DN
+  field_domain: DomÃ¤ne
   field_attr_login: Mitgliedsname-Attribut
   field_attr_firstname: Vorname-Attribut
   field_attr_lastname: Name-Attribut
diff --git a/config/locales/en.yml b/config/locales/en.yml
index 4082670..b155582 100644
--- a/config/locales/en.yml
+++ b/config/locales/en.yml
@@ -243,6 +243,7 @@ en:
   field_port: Port
   field_account: Account
   field_base_dn: Base DN
+  field_domain: Domain
   field_attr_login: Login attribute
   field_attr_firstname: Firstname attribute
   field_attr_lastname: Lastname attribute
diff --git a/db/migrate/20100330124427_add_auth_sources_domain.rb b/db/migrate/20100330124427_add_auth_sources
_domain.rb
new file mode 100644
index 0000000..f9d1de5
--- /dev/null
+++ b/db/migrate/20100330124427_add_auth_sources_domain.rb
@@ -0,0 +1,9 @@
+class AddAuthSourcesDomain < ActiveRecord::Migration
+  def self.up
+    add_column :auth_sources, :domain, :string, :default => 'none', :null => false
+  end
+
+  def self.down
+    remove_column :auth_sources, :domain
+  end
+end
```

#18 - 2010-12-13 16:55 - Antoine Beaupré

- File 1913_redmine_bind_as_user2.diff added

For me the patch in comment #17 was unclear: first, there's commented code in there and second, I don't understand the reason for the 'domain' field. The patch I am providing here is just a port of the one in comment #14, which is itself a port of my patch, which was a port of... well, you see where I'm going. :)

Basically, I believe that using $login in the field is (a) much more powerful and (b) flexible enough to cover for the "domain" case, which is really unclear to me what it does.

Can we get this committed please? This issue has been opened for 2 years and a patch has been available for over a year now. I've been using it in production here for the last 10 months without any problems.

The patch applies to 1.0.1 but it should be trivial to port it to trunk. Besides, i did that earlier and that didn't seem to favor inclusion so I'll just try to port this to new releases as we upgrade, but I'd really like to see this hit the trunk so i don't have to maintain this silly patch.

#19 - 2010-12-13 18:24 - Adi Kriegisch

Thanks, Antoine, for forward porting this patch. (Just a minor correction: this patch is available for two years now -- and it is still working in production ;-)
Probably main developers are lacking something? What do I/we need to do to finally get this included into Redmine?
Eric Davis (comment 4) asked for test cases. How should they look like? Is it enough to proof that this patch generates a correct login dn?
Probably documentation is missing? How should the documentation look like? The very first comment explains how to bind against AD. Comment 8 and 9 provide useful LDAP config snippets for OpenLDAP. Anything more needed?
I'd love to have this upstream: rebuilding packages all the time to get this working again and again is just annoying.

#20 - 2012-02-01 16:22 - Daniel Ritz

+1

Even the minimal patch in 1913_redmine_bind_as_user2.diff is a big improvement. Using that one at work to authenticate against a Windows AD without requiring an extra user (which is not easy to get...corporate politics).

#21 - 2012-02-13 18:18 - Antoine Beaupré

I can confirm this is still working in production, and I painstakingly update this at every release. So far there was no change in the 1.1 release, but we'll see for 1.3.

How **do** we get patches merged into redmine? I have a good lot waiting in the queue and they seem to be getting no attention whatsoever...

#22 - 2012-03-15 19:33 - Antoine Beaupré

Hello? Anyone?

Should I submit this to chiliproject instead?

#23 - 2012-03-16 08:18 - Jean-Philippe Lang

Would you be able to add a test case to the patch ? Or should I take care of it ?

#24 - 2012-03-16 09:08 - Adi Kriegisch

Jean-Philippe, it would be great if you can add test cases. Please let me know if you lack anything for adding this patch. I'd gladly help!
Thank you for considering this patch!

#25 - 2012-03-16 18:05 - Antoine Beaupré

I am sorry I am not familiar enough with Redmine's unit testing to provide a test case here. Besides, I think it would require a running LDAP server, which is not trivial...

It would be awesome if someone else could provide that test case though. Thanks for looking into this patch!

#26 - 2012-03-17 13:17 - Jean-Philippe Lang

- Status changed from New to Closed

- Assignee set to Jean-Philippe Lang

- Target version set to 1.4.0

Feature added in r9241 - r9243 with slight changes and tests. The initial patch was breaking 2 tests and i think it's safer to escape the submitted login.

Antoine Beaupré wrote:

> Besides, I think it would require a running LDAP server, which is not trivial...

This was already required to run the full test suite, so adding tests for this new feature was pretty straightforward.
Thanks for your contribution.

#27 - 2012-05-21 22:38 - Harley Laue

extra/svn/Redmine.pm still needs patched (which was supplied) to work correctly. Without this patch, SVN read and/or write is completely broken while using this feature within Redmine.

#28 - 2012-05-29 20:27 - Jean-Philippe Lang

Indeed, #11046 created.

#29 - 2014-06-05 22:32 - Andrew Kohlsmith

There is a more generic issue with binding to the AD. I ran into this and don't have a good solution, but I'll describe it here so that hopefully others will

find it.

Windows 2003 seems to confuse the notion of binding to the LDAP server and logging in to a workstation. If you have a domain user who is restricted to logging in only from a specific workstation or workstations, the bind as that user will fail because AD sees the bind, validates the password and then notices that the AD server itself is not in the list of allowed workstations. The specific LDAP error is

```
W80090308: LdapErr: DSID-0C090334, comment: AcceptSecurityContext error, data 531, vece
```

where the 531 corresponds to "User not allowed to logon at this computer".

I don't know enough about AD to know if there is a way to say "any valid account can bind the AD server" while maintaining the workstation login restriction. The only way around it that I have found is to add the NetBIOS name of the AD server to the restricted user's allowed workstation list. This is clearly not a great solution.

Files

Redmine-app-models-auth_source_ldap.diff	818 Bytes	2008-09-16	Adi Kriegisch
Redmine-ldap-as-user.diff	1.24 KB	2009-08-21	Adi Kriegisch
Redmine-app-models-auth_source_ldap-0.9.1-2.diff	1.68 KB	2010-02-21	Antoine Beaupré
1913_redmine_bind_as_user.diff	2.1 KB	2010-02-21	Antoine Beaupré
Bind_as_user_LDAP.diff	1.11 KB	2010-04-20	Markus Peter
1913_redmine_bind_as_user2.diff	1.08 KB	2010-12-13	Antoine Beaupré