Redmine - Defect #10202

Access to svn may not be granted by redmine.pm if user is authenticated by an external LDAP server

2012-02-10 19:29 - Tiemo Vorschuetz

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	LDAP	**Estimated time:**	0.00 hour
Target version:			
Resolution:		**Affected version:**	1.3.1

Description

If a user is part of two roles, one that granted access and one that does not allow repository browsing, and if the user is authenticated through an external LDAP server (e.g. MS AD) the access to the svn repository may fail. This depends on the order the permissions are calculated inside the while loop below.

The is_member function inside the redmine.pm file should be modified as shown below.

...
my $ret;
while (my ($hashed_password, $salt, $auth_source_id, $permissions) = $sth->fetchrow_array) {
...

should be changed to

...
my $ret = 0;
while ((my ($hashed_password, $salt, $auth_source_id, $permissions) = $sth->fetchrow_array) and not $ret) {
...

Regards,
Tiemo

History

#1 - 2012-02-14 10:26 - Jean-Philippe Lang

I'm not able to reproduce although I made sure that a role without the permission was returned first.
The while loop in Redmine.pm tests all roles so I can't see how it could happen. Can you post the entire code of your sub is_member function in Redmine.pm?

#2 - 2012-02-14 13:52 - Tiemo Vorschuetz

Hi Jean,

this is the sub that is working for me:

```
sub is_member {
  my $redmine_user = shift;
  my $redmine_pass = shift;
  my $project_id   = shift;
  my $r = shift;

  my $dbh          = connect_database($r);

  my $pass_digest = Digest::SHA1::sha1_hex($redmine_pass);

  my $usrprojpass;
  if ($cfg->{RedmineCacheCredsMax}) {
    $usrprojpass = $cfg->{RedmineCacheCreds}->get($redmine_user.":".$project_id);
    return 1 if (defined $usrprojpass and ($usrprojpass eq $pass_digest));
  }
  my $query = $cfg->{RedmineQuery};
  my $sth = $dbh->prepare($query);
  $sth->execute($redmine_user, $project_id);

  my $ret;
```

```
  my $user = $r->user;
  $ret = 0;

  while ((my ($hashed_password, $auth_source_id, $permissions) = $sth->fetchrow_array) and not $ret){

      $ret = is_admin( $r->user, $r );
      unless ($auth_source_id) {
          my $method = $r->method;
          if ($hashed_password eq $pass_digest && ((defined $read_only_methods{$method} && is_admin( $r->user,
 $r ) || $permissions =~ /:browse_repository/) || $permissions =~ /:commit_access/) ) {
              $ret = 1;
              last;
          }
      } elsif ($CanUseLDAPAuth) {
          #printlog("LDAP user");
          my $sthldap = $dbh->prepare(
              "SELECT host,port,tls,account,account_password,base_dn,attr_login from auth_sources WHERE id = ?
;"
          );
          $sthldap->execute($auth_source_id);
          while (my @rowldap = $sthldap->fetchrow_array) {
            my $ldap = Authen::Simple::LDAP->new(
                host    =>      ($rowldap[2] eq "1" || $rowldap[2] eq "t") ? "ldaps://$rowldap[0]:$rowldap[1]"
 : $rowldap[0],
                port    =>      $rowldap[1],
                basedn  =>      $rowldap[5],
                binddn  =>      $rowldap[3] ? $rowldap[3] : "",
                bindpw  =>      $rowldap[4] ? $rowldap[4] : "",
                filter  =>      "(".$rowldap[6]."=%s)"
            );
            my $method = $r->method;
            $ret = 1 if ($ldap->authenticate($redmine_user, $redmine_pass) && ((defined $read_only_methods{$me
thod} && is_admin( $r->user, $r ) || $permissions =~ /:browse_repository/) || $permissions =~ /:commit_access/
));
          }
          $sthldap->finish();
          undef $sthldap;
      }
      close File;
  }
  $sth->finish();
  undef $sth;
  $dbh->disconnect();
  undef $dbh;

  if ($cfg->{RedmineCacheCredsMax} and $ret) {
    if (defined $usrprojpass) {
      $cfg->{RedmineCacheCreds}->set($redmine_user.":".$project_id, $pass_digest);
    } else {
      if ($cfg->{RedmineCacheCredsCount} < $cfg->{RedmineCacheCredsMax}) {
        $cfg->{RedmineCacheCreds}->set($redmine_user.":".$project_id, $pass_digest);
        $cfg->{RedmineCacheCredsCount}++;
      } else {
        $cfg->{RedmineCacheCreds}->clear();
        $cfg->{RedmineCacheCredsCount} = 0;
      }
    }
  }

  $ret;
}
```

Regards,
Tiemo

#3 - 2012-02-14 14:21 - Tiemo Vorschuetz

We added the use case, that any admin is also able to browse any repository. Maybe this is the case?