

Using Kerberos enabled connections with Sybase

Source: <http://coding.derkeiler.com/Archive/Perl/perl.dbi.users/2006-02/msg00171.html>

- *From:* chuckfox2@xxxxxxx (Chuck Fox)
 - *Date:* Thu, 23 Feb 2006 14:51:38 -0500
-

Hello fellow dbi-users.

I am attempting to connect to a 12.5 Sybase server using kerberos enabled connections. My isql and sqsh both correctly connect (sqsh needed a small fix to load the security). However, I am unable to get DBD::Sybase to load the security modules.

Here are the details about the problem

1. Software

OS – RHEL4, 32 bit ASE-15/OCS-15_0
isql from OCS-15_0
DBI-1.50
DBD-Sybase-1.07
perl 5.8.5

2. Environment variables

PATH=/apps/sybase/ASE15/SYSAM-2_0/bin:/apps/sybase/ASE15/OCS-15_0/bin:/apps/sybase/ASE15/ASE-15_0/bin
LD_LIBRARY_PATH=/lib/i686:/apps/sybase/ASE15/DataAccess/ODBC/lib:/apps/sybase/ASE15/EFTS-15_0/verity
NO PERLLIB PERL5LIB set

3. Build

```
cd DBI-1.50
/usr/bin/perl Makefile.PL
make
make install
cd ../DBD-Sybase-1.07
/usr/bin/perl Makefile.PL
make
make install
```

4. test

```
tested using t/login.t, can connect to my ASE with no problem
tested using isql to connect to my ASE using Kerberos (-V) with NO PROBLEM
modified t/login.t as following
#!/perl
#
# $Id: login.t,v 1.3 2004/12/16 12:06:01 mpeppler Exp $
```

```
use lib 'blib/lib';
```

Using Kerberos enabled connections with Sybase

```
use lib 'blib/arch';

BEGIN {
$ENV{SYBASE} = "/apps/sybase/ASE15";
$ENV{SYBASE_OCS} = "OCS-15_0";
}

use lib 't';
use _test;

use strict;

use Test::More tests => 5;

use vars qw($Pwd $Uid $Srv $Db);

BEGIN { use_ok('DBI');
use_ok('DBD::Sybase');}

($Uid, $Pwd, $Srv, $Db) = _test::get_info();

#DBI->trace(3);
my $dbh = DBI->connect("dbi:Sybase:server=$Srv;database=$Db", 'sa', $Pwd, {PrintError => 1});
#DBI->trace(0);
ok(defined($dbh), 'Connect');

#DBI->trace(3);
my $dbh = DBI->connect("dbi:Sybase:server=$Srv;kerberos=$Srv;database=$Db", "chuckfox2", $Pwd,
{PrintError => 1});
#DBI->trace(0);
ok(defined($dbh), 'Connect');

$dbh->disconnect if $dbh;

$dbh = DBI->connect("dbi:Sybase:server=$Srv;database=$Db", 'ohmygod', 'xzyzzy', {PrintError => 0});

ok(!defined($dbh), 'Connect fail');

$dbh->disconnect if $dbh;

exit(0);

> perl t/login.t
1..5
ok 1 - use DBI;
ok 2 - use DBD::Sybase;
ok 3 - Connect
ct_con_props(CS_SEC_NETWORKAUTH) failed at blib/lib/DBD/Sybase.pm line 94.
DBI connect('server=MYTEST1_SYB;kerberos=MYTEST1_SYB;database=tempdb','chuckfox2',...) failed:
OpenClient message: LAYER = (7) ORIGIN = (9) SEVERITY = (5) NUMBER = (12)
Server MYTEST1_SYB, database
```

Using Kerberos enabled connections with Sybase

```
Message String: ct_con_props(SET): security service layer: internal security control layer error: error string
not available
at t/login.t line 33
not ok 4 - Connect
# Failed test (t/login.t at line 35)
ok 5 - Connect fail
# Looks like you failed 1 tests of 5.
```

Poking around in the DBD code (dbdimp.c) and it looks like that the security driver cannot be loaded.

```
int
syb_db_login(dbh, imp_dbh, dsn, uid, pwd, attribs)

<code omitted for clarity>

imp_dbh->kerberosPrincipal[0] = 0;
imp_dbh->kerbGetTicket = fetchSvAttrib(attribs, "syb_kerberos_serverprincipal");

<more code omitted> if(strchr(dsn, '=') {
<more code omitted> extractFromDsn("kerberos=", dsn, imp_dbh->kerberosPrincipal, 32);
<more code omitted> }
<more code omitted>

if(imp_dbh->kerbGetTicket) {
fetchKerbTicket(imp_dbh);
}
```

It appears that you have to pass the syb_kerberos_serverprincipal through the attributes as opposed to using the DSN. Should the check be against kerberosPrincipal instead of kerbGetTicket ?

Any help would be appreciated.

Regards,

Chuck