Mizu VOIP Server Database Interface

Table of Contents

About

Mizu VOIP server natively supports MS SQL and most of the installations are using the free MS SQL Express or the full versions of the database engine. You can work with the database using the SQL Server Management application or connecting to it with ODBC or ADO from any programming language like C++, C# or PHP.

Mostly you will work with the user table (tb_users) and with the cdr table (tb_cdrs) to load statistics.
Please consult the Admin Guide and the database layout to be familiar with these tables.

ADO Connection String

Provider=SQLOLEDB.1;Password=PASSWORD;Persist Security Info=True;User ID=USERNAME;Initial Catalog=VOIPSERVERDBNAME;Data Source=127.0.0.1,PORT;

Default values:
PORT: 1433 (or 2223)
VOIPSERVERDBNAME: mserver
USERNAME: sa
PASSWORD: srEgtknj34f

You can find the values for your server in the "mizuserver.ini" file found in the server application directory (the default database password must always be changed during the server install)

Do not hardcode the connection string in any file or application that can be accessed/downloaded by the users!
Make sure that your code cannot be affected with SQL injection!
For better security you might use a middle layer between your application and the database and don't allow direct connection to the database from the public internet.

Examples:

User authentication:

query->SQL->Text = "select id from tb_users with(nolock) where username = "+QuotedStr(USERNAME)+" and password = "+QuotedStr(PASSWORD)+" and enabled <> 0 and temporarydisabled <> 1";

```
query->Open();
if(!query->Eof && !query->FieldByName("id")->IsNull)
{
 //successfull login
}
else
{
 //invalid username/password
}
query->Close();
```

Note: instead of using quoted strings it is safer to use query parameters!

Add enduser record

insert into tb_users (username, password,credit, name, email, phone, address,type,isoperator,Postpaid,credit,needauth,comment)
values ('USERNAME','PASSWORD',0,'FULL NAME','EMAIL ADDR','PHONENUMBER','ADDRESS',0,2,0,0,4,'new user');

Add traffic sender record

insert into tb_users (username, password, authip, type,Postpaid,credit, needauth, comment)
values ('NAME','RANDOMPASSWORD','ORIGINATINGIP',5,1,0,1,'new tr');

Add credit for endusers

> Short way:

update tb_users set credit = credit + VALUE where id = USERID (or where username = 'USERNAME')

> Better way:

1. Check if user exists (reject if the user is missing, not enabled or is a postpaid account)
select top 1 id,credit,postpaid from tb_users with (nolock) where username = 'USERNAME' and enabled <> 0
2. Check if user is not in a call (and reject the transaction if yes)
select top 1 id from tb_currcalls with(nolock) where callerid = USERID
3. Add credit
update tb_users set credit = credit + VALUE where id = USERID
4. Increase account elapse timeout (This is optional if timeouts are enabled. User records can be automatically disabled after a time of inactivity)
update tb_users set creditelapsedt = DATETIME, accelapsedt = DATETIME where id = USERID
DATETIME must be a value in the future based on your calculations (getdate()+days)
5. Log the transaction (good for later statistics)
insert into tb_invoices (type,userid,paymentreceived,paymentprice,comment,payment_type,currentcredit) values (8, USERID,getdate(),VALUE,'web recharge','WEB',NEWCREDIT)

Make sure that the bank transfer (or pin code) credit decrease is in the same transaction with the voip account credit increase so if any error occurs you can rollback the whole transaction.

Set call forwarding

update tb_users set forwardonbusy = 'PHONENUME' where id = USERID

Disable enduser

update tb_users set temporarydisabled = 1 where id = USERID

Delete enduser

update tb_users set enabled = 0 where id = USERID
OR
delete from tb_users where id = USERID (this is not recommended)

Get user status

For this you can write a direct query and check the "status" and the "laststatusdate" for the user.
Or you can use the following stored procedure:

RETURN VALUES:
 null: user not found or disabled
 0: offline (deregistered or last seen is more than 10 minute ago)
 1: online (registered)
 2: in call (calling or speaking)

Add a speed dial entry for a user

insert into tb_speeddial (userid,shortnum,num) values ('USERID', 'SPEEDNUMBER','PHONENUMBER')

Display rate list

First you must determine which listing (ID tb_billentries) do you wish to display. The ID's can be found if you execute the following SQL:
SELECT TOP 100 id,name FROM tb_billentries with(nolock) ORDER BY name
Then you can display the listing from tb_billingtimes:
SELECT prefix AS 'Direction', origprice AS 'Cost'
FROM tb_billingtimes with(nolock) WHERE entryid = X
ORDER BY prefix

If you wish to display also the country/operator, then join it with the tb_directions table:
SELECT prefix AS 'Prefix',
(
            select top 1 tb_directions.name
    from tb_directions with(nolock) where
    tb_billingtimes.prefix + 'xxxxxx' like tb_directions.prefix +'%'
            order by LEN(tb_directions.prefix) desc
) as 'Direction',
origprice AS 'Cost'
FROM tb_billingtimes with(nolock) WHERE entryid = X
ORDER BY Prefix

For other pricing options like billingstep, currency, etc, please load the corresponding value from tb_billentries or tb_billingtimes.

Initiate callback or p2p call

-for callback and phone to phone calls the recommended method is the http or tcp interface. By using the database interface there is no way to get a direct feedback about the results, however it is very easy to implement by simply inserting a new record to tb_conferences.
-for these to work the "checkdbconf" global setting parameter has to be set to a positive value used for database polling interval. (For example 5000). The default value is 0 (functionality disabled)

Callback:
Insert into tb_conferences (type,createdby, createdbynumber, number) values (2,"BILLEDUSERID","BILLEDUSERNUMBER","NUMBERTOCALL")

P2P:
Insert into tb_conferences (type,createdby, createdbynumber, number1, number2) values (3,"BILLEDUSERID","BILLEDUSERNUMBER","NUMBERTOCALL1","NUMBERTOCALL2")

Statistics examples

```
        case 0: //Today data
            {
                        sql = "select \
count(*) as 'Calls', \
sum(realduration)/60 as 'Minutes', \
cast(sum(costenduser) as numeric(19,2)) AS 'Cost' \
from tb_cdrs WITH(NOLOCK) \
WHERE \
datum > GETDATE() - 1 and \
callerid = "+IntToStr(UserSession()->userid)+" AND \
DATEPART(YEAR,datum) = DATEPART(YEAR,GETDATE())  AND \
DATEPART(month,datum) = DATEPART(month,GETDATE())  AND \
DATEPART(day,datum) = DATEPART(day,GETDATE())";
            } break;
            case 1: //This month data
            {
                        sql = "select \
```

```
count(*) as 'Calls', \
sum(realduration)/60 as 'Minutes', \
cast(sum(costenduser) as numeric(19,2)) AS 'Cost' \
from tb_cdrs WITH(NOLOCK) \
WHERE \
datum > GETDATE() - 33 and \
callerid = "+IntToStr(UserSession()->userid)+" AND \
DATEPART(YEAR,datum) = DATEPART(YEAR,GETDATE())  AND \
DATEPART(month,datum) = DATEPART(month,GETDATE())";
                } break;
                case 2: //Last month data
                {
                            sql = "select \
count(*) as 'Calls', \
sum(realduration)/60 as 'Minutes', \
cast(sum(costenduser) as numeric(19,2)) AS 'Cost' \
from tb_cdrs WITH(NOLOCK) \
WHERE \
datum > GETDATE() - 63 and \
callerid = "+IntToStr(UserSession()->userid)+" AND \
DATEPART(YEAR,datum) = DATEPART(YEAR,GETDATE())  AND \
DATEPART(month,datum) = DATEPART(month,GETDATE())-1";
                } break;
                case 3: //Calls by day
                {
                            sql = "SELECT \
DATEPART(year,b.datum) as 'Year', \
DATEPART(month,b.datum) as 'Month', \
DATEPART(Day,b.datum) as 'Day', \
count(b.id) as 'Count', \
sum(b.realduration)/60 as 'Duration', \
sum(b.costenduser) as 'Cost' \
FROM tb_cdrs b WITH(NOLOCK) \
WHERE \
b.datum >= GETDATE()-62 AND \
callerid = "+IntToStr(UserSession()->userid)+"  AND \
realduration > 0 \
GROUP BY DATEPART(year,b.datum), datepart(month,b.datum), datepart(day,b.datum) \
ORDER BY DATEPART(year,b.datum), datepart(month,b.datum), datepart(day,b.datum)";
                } break;
                case 4: //List last calls
                {
                            sql = "SELECT TOP 100 \
datum as 'Time', Connecttime,realduration AS 'Duration (sec)', callernumber AS 'Caller', origcallednumber AS 'Called', cast(costenduser as numeric(19,2)) AS 'Cost', tb_reasoncodes.text as 'DiscReason' \
FROM tb_cdrs WITH(NOLOCK) \
left join tb_reasoncodes with(nolock) on (discreason = tb_reasoncodes.code) \
WHERE \
datum > GETDATE() - 63 and \
callerid = "+IntToStr(UserSession()->userid)+" \
ORDER BY datum desc";
                } break;
```

.NET C# example

```
SqlConnection connection = new SqlConnection(Utils.GetConnStr());
SqlCommand command;
string sqlstring;
connection.Open();
sqlstring = "SELECT credit from tb_users WHERE ID = "+Convert.ToString(Session["userid"])+" OR BillingUserID = " +Convert.ToString(Session["userid"]);
command = new SqlCommand(sqlstring, connection);
tcredit.Text = "Credit: " + command.ExecuteScalar().ToString();
tcredit.Visible = true;
connection.Close();
```

PHP example

The mssql module must be enabled in the PHP configuration file.
http://www.php.net/mssql

```
<?

 include_once("functions/func.php");

try{

 $tridx = "unknown";
 $devizax = "unknown";
```

```php
  $amountx = "-1";
  $resultx = "unknown";

  //load values for tridx, devizax, etc here

  $tridx = str_quote($tridx);
  $devizax = str_quote($devizax);
  $amountx = str_quote($amountx);
  $resultx = str_quote($resultx);
  //todo: store the results in tb_banktransfers and tb_logs and modify user credit on success
  //tb_banktransfers: id, datum,userid, product, price, currency, trid, result, state, enddate,comment    state: 0=unknown,1=initok, 2=initfailed,3=finishedok,4=transfailed
  $con = mssql_connect("127.0.0.1", "username", "password");
  if(!$con)
  {
    echo "ERROR: cannot connect to database!";
    exit;
  }
  if(!mssql_select_db ("voipserverdb", $con))
  {
    echo "ERROR: db cannot sel!";
    mssql_close ($con);
    exit;
  }
  $sql= "select top 1 id, userid from tb_banktransfers where trid = '" . $tridx . "' and price = " . $amountx . " and datediff(hour,datum,getdate()) < 3 and state = 1 order by datum desc";
  $rs= mssql_query ($sql, $con);
  if(!$rs || $rs == FALSE)
  {
    echo "ERROR: cannot select transaction!";
    $sql= "insert into tb_logs (ip, source, msgtype, message) values ('127.0.0.1','WEB','ERROR','bankpayment: error loading transaction when returned from xxx. trid =" .$tridx. ", price = " .$amountx. ", result =
" . $resultx . " ')";
    mssql_query ($sql, $con);
  }
  elseif(mssql_num_rows($rs) != 1)
  {
    echo "ERROR: cannot found transaction!";
    $sql= "insert into tb_logs (ip, source, msgtype, message) values ('127.0.0.1','WEB','ERROR','bankpayment: cannot find transaction when returned from xxx. trid =" .$tridx. ", price = " .$amountx. ", result =  "
. $resultx . " ')";
    mssql_query ($sql, $con);
  }
  else
  {
                  $transactionid = mssql_result ($rs, 0, 0);
                  $userid = mssql_result ($rs, 0, 1);
                  if(!$transactionid || $transactionid < 1 || !$userid || $userid < 1 || strlen($transactionid) < 1)
                  {
                    $sql= "insert into tb_logs (ip, source, msgtype, message) values ('127.0.0.1','WEB','ERROR','bankpayment: invalid trid returned from xxx. trid =" . $tridx . ", price = " . $amountx . ", result =  " .
$resultx . ", id = ". $transactionid ." ')";
                    mssql_query ($sql, $con);
                  }
                  else
                  {
                    if($resultx == "OK")
                          {
                                      $sql= "update tb_users set credit = credit + " . ($amountx * 100) . " where id = " . $userid . " and datedeiff(hour,lastweblogintime,getdate()) < 6  and enabled = 1 and
temporarydisabled = 0";
                                      mssql_query ($sql, $con);

                                      $sql= "update tb_banktransfers set state = 3,enddate = getdate(), comment = comment + ' finalret is OK' where id = " . $transactionid . " ";
                                      mssql_query ($sql, $con);

                                      $sql= "insert into tb_logs (ip, source, msgtype, message) values ('127.0.0.1','WEB','ERROR','bankpayment: " . $transactionid . " successfully finished for user " . $userid ."
')";
                                      mssql_query ($sql, $con);

                                      echo "RESULT: transfer finished successfully received OK.";
                          }
                          else
                          {
                                      $sql= "update tb_banktransfers set state = 4,enddate = getdate(), comment = comment + ' finalret is " .  $resultx  . "' where id = " . $transactionid . " ";
                                      mssql_query ($sql, $con);

                                      $sql= "insert into tb_logs (ip, source, msgtype, message) values ('127.0.0.1','WEB','ERROR','bankpayment: " . $transactionid . " failed for user " . $userid ." with result " .
$resultx  . " ')";
                                      mssql_query ($sql, $con);

                                      echo "RESULT: transfer failed received OK.";
                          }
                  }
  }

  if($con) { mssql_close ($con);  $con = null; }
  exit;
```

```php
} catch (Exception $e) {  }
try{
    echo "ERROR: fatal processing error";
                if($con)
                {
                    mssql_close ($con);
                }
} catch (Exception $e) {  }
?>
```

C++ example

Recharge user credit with recharge card (PIN number):

```cpp
TMyADOQuery *query = dbobj->GetQuery(); //get query will prepare a connected query object from DAL
if(!query) return;
try{
query->OpenEx("select top 1 * from tb_prepaidcodes where code = "+QuotedStrEx(common.params[1])+" and enabled <> 0 and (usedby is null or usedby < 1) and credit > 0 ");
if(query->EofEx())
{
  query->CloseEx();
  processingthread->SetResult( new TSocketData(socketdata, "ERROR,invalid PIN code\r\n"));
}
else
{
  float credit = query->FieldByNameEx("credit").AsFloat();
  query->EditEx();
  query->FieldByName("usedby")->AsInteger = data->userid;
  query->FieldByName("used")->AsDateTime = Now();  //this should be done in transaction (clear pin only on successful charge)
  query->PostEx();
  query->CloseEx();

  query->OpenEx("select id,credit,creditelapsedt,accelapsedt from tb_users with (nolock) where id = "+IntToStrEx(data->userid));
  if(query->EofEx() || query->FieldByNameEx("id").IsNull())
  {
    query->CloseEx();
    processingthread->SetResult( new TSocketData(socketdata, "ERROR,no user\r\n"));
  }
  else
  {
    float fdays = 99999;
    if(config.creditelapseunit > 0)
    {
      fdays = query->FieldByNameEx("credit").AsFloat()/config.creditelapseunit;
    }

    int days = fdays;
    if(days > config.maxcreditelapsedays && config.maxcreditelapsedays > 0) days = config.maxcreditelapsedays;

    query->EditEx();
    query->FieldByName("credit")->AsFloat = query->FieldByNameEx("credit").AsFloat() + credit;
    query->FieldByName("creditelapsedt")->AsDateTime = Now()+days;
    query->FieldByName("accelapsedt")->AsDateTime = Now()+days+config.accelapsedays;
    query->PostEx();
    query->CloseEx();

    float userdebt = 0;
    query->OpenEx("select top 1 debt as 'ccc' from tb_invoices where userid = "+IntToStrEx(data->userid)+" order by datum desc, id desc");
    if(!query->EofEx() && !query->FieldByNameEx("ccc").IsNull() )
    {
      userdebt = query->FieldByNameEx("ccc").AsFloat();
    }

    query->ExecSQLEx("insert into tb_invoices (type,userid,paymentreceived,paymentprice,debt,comment,payment_type) values (8,"+IntToStrEx(data-
>userid)+",getdate(),"+FloatToStr(credit)+","+FloatToStr(userdebt - credit)+",'console payment','PIN')");

    query->OpenEx("select credit from tb_users with (nolock) where id = "+IntToStrEx(data->userid));

    if(query->EofEx() || query->FieldByNameEx("credit").IsNull())
    {
      query->CloseEx();
      processingthread->SetResult( new TSocketData(socketdata, "ERROR,err loading credit\r\n"));
    }
    else
    {
      processingthread->SetResult( new TSocketData(socketdata,  (StringEx)"OK,new credit is "+query->FieldByNameEx("credit").AsString()+"\r\n"));
      query->CloseEx();
    }
  }
}

}
```

```
}catch(Exception &e) { common.PutToDebugLog2((StringEx)"ERROR,"+ e.Message); }
if(dbobj) dbobj->ReleaseQuery(query); query = NULL;
```

Resources

For more details please consult the Admin Guide and browse the database with the SQL Server Management application.

For help, contact serversupport@mizu-voip.com

http://www.mizu-voip.com

```
}catch(Exception &e) { common.PutToDebugLog2((StringEx)"ERROR,"+ e.Message); }
if(dbobj) dbobj->ReleaseQuery(query); query = NULL;
```