

Re: Location of linked table username/password

Source:

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.externaldata/2004-02/0243.html>

From: Joe Fallon (*jfallon1_at_nospamtwcny.rr.com*)

Date: 02/07/04

Date: Fri, 6 Feb 2004 23:40:58 -0500

Not sure I get your question.

If you want to avoid a log-on screen then you need to connect to Oracle with a hardcoded account that has the privileges you need. This code uses variables but you can tweak it.

=====

I use this procedure to re-create links to Oracle.

There is a local Access table (tblODBCTables) that contains the table names and primary key fields I want to link to on the Server.

Note: the source table name needs the Schema User prefix which is in the code. The linked table name usually omits this.

```
Public Function LinkOracleTables(strDSN As String, strSchema As Variant,
strSchemaPwd As Variant) As Boolean
On Error GoTo Err_LinkOracleTables
```

```
Dim db As Database, rs As Recordset, tdfAccess As TableDef, qdf As
QueryDef
Dim dbODBC As Database, strConnect As String, strSQL As String
```

```
If strDSN = "" Then
MsgBox "You must supply a DSN in order to link tables."
Exit Sub
Else
strConnect = "ODBC;DSN=" & strDSN & ";UID=xyz" & ";PWD=abc;"
End If
```

```
SysCmd acSysCmdSetStatus, "Connecting to Oracle..."
```

```
Call DeleteODBCTableNames
```

```
Set db = CurrentDb
Set rs = db.OpenRecordset("tblODBCTables")
Set dbODBC = OpenDatabase("", False, False, strConnect)
DoCmd.SetWarnings False
```

```
Do While Not rs.EOF
    Set tdfAccess = db.CreateTableDef(rs![LinkTablename], dbAttachSavePWD)
    tdfAccess.Connect = dbODBC.Connect
    tdfAccess.SourceTableName = strSchema & "." & rs![LinkTablename]
    db.TableDefs.Append tdfAccess
    'run pseudo index queries here. If the table does not exist then this
gets skipped.
    strSQL = "CREATE INDEX " & rs![LinkTablename] & "Idx ON " &
rs![LinkTablename] & " (" & rs![IndexFields] & ");"
    DoCmd.RunSQL strSQL
TableNotInCollection:
    rs.MoveNext
Loop

LinkOracleTables = True

Exit_LinkOracleTables:
    On Error Resume Next
    DoCmd.SetWarnings True
    rs.Close
    Set rs = Nothing
    Set dbODBC = Nothing
    Set db = Nothing
    SysCmd acSysCmdClearStatus
    Exit Function

Err_LinkOracleTables:
    Select Case Err.Number
        Case 3151
            MsgBox ("There is an ODBC datasource problem." & vbCrLf & "Please
verify the DSN and database are spelled correctly." & vbCrLf & "Note: They
can be case sensitive.")
            Case 3265, 3011, 7874 'item not in collection – table does not exist, or
can't find object
                Resume TableNotInCollection
        Case Else
            MsgBox "Error # " & Err.Number & " was generated by " & Err.Source &
vbCrLf & Err.Description, , "LogOnCode – LinkOracleTables"
    End Select
    LinkOracleTables = False
    Resume Exit_LinkOracleTables

End Function
```

'This procedure deletes all linked ODBC table names in an mdb.

```
Public Sub DeleteODBCTableNames()
On Error GoTo Err_DeleteODBCTableNames

    Dim dbs As Database, tdf As TableDef, I As Integer
    Set dbs = CurrentDb
    For I = dbs.TableDefs.Count – 1 To 0 Step –1
```

microsoft.public.access.externaldata: Re: Location of linked table username/password

```
Set tdf = dbs.TableDefs(I)
If (tdf.Attributes And dbAttachedODBC) Then
    dbs.TableDefs.Delete (tdf.Name)
End If
Next I
```

```
dbs.Close
Set dbs = Nothing
```

```
Exit_DeleteODBCTableNames:
Exit Sub
```

```
Err_DeleteODBCTableNames:
MsgBox ("Error # " & Str(Err.Number) & " was generated by " & Err.Source
& Chr(13) & Err.Description)
Resume Exit_DeleteODBCTableNames
```

```
End Sub
```

```
--
Joe Fallon
Access MVP
"Jason" <pottsja@omu.org> wrote in message
news:bfd501c3ecd3$058d0b50$a501280a@phx.gbl...
> Greetings,
>
> When you link a table from a password-protected data
> source and select save the password, MS Access stores the
> username and password in the string contained in
> MSysObjects.Connect. However, if you do not make this
> selection, where does Access store this information?
>
> I am trying to avoid creating yet another password for our
> users. By having them log into the existing Oracle
> database, then stripping that log-in information out, I
> can use that information for the functionality I am
> building in this database.
>
> Thank you for the information.
```