# *Encrypted Split No Strings Database DEMO*

This application has been created in response to several forum questions about methods of preventing data being stolen from Access databases. For example, this AWF thread: [Prevent Importing ODBC tables from ACCDE]

It is intended to show how the data in an Access database can be made reasonably secure against hackers whilst still allowing full functionality to authorised users.

This DEMO should behave exactly as any split database …
… BUT there are no linked tables and therefore no connection strings visible in the MSysObjects system table.
In addition, the data is protected using a RC4 encryption cipher.

There are THREE versions of this DEMO - ACCDB or ACCDE for 32-bit or 64-bit Access
Each zip file contains a split database with 2 files. Save both files in the same folder

a) 32-bit ACCDE frontend FEX32.accde & BEX.accdb
b) 64-bit ACCDE frontend FEX64.accde & BEX.accdb
c) FE.accdb & BE.accdb

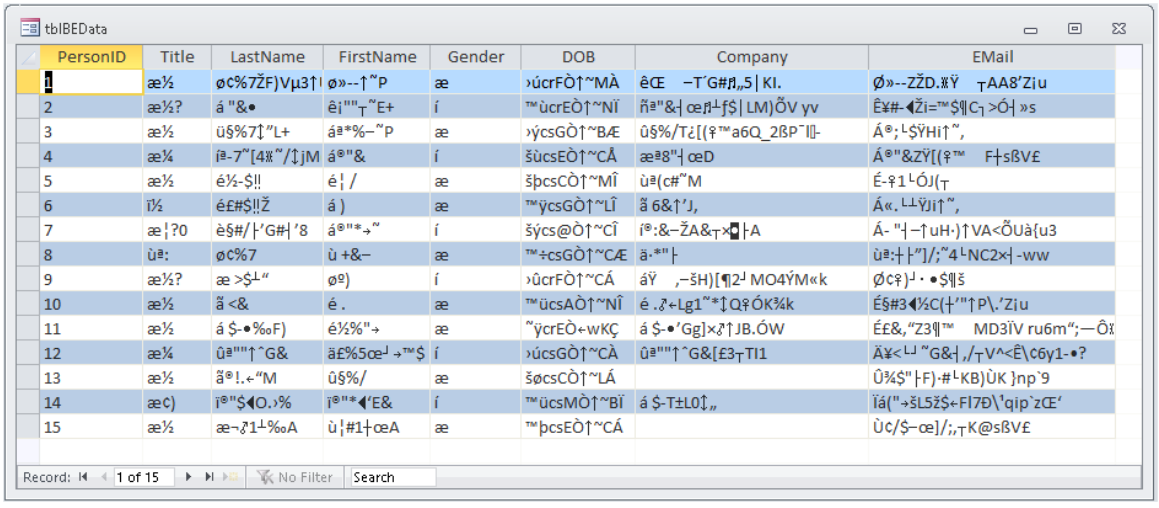
Both ACCDE versions have been locked down with the navigation pane & ribbon removed
The ACCDB FE file is not password protected so you can view the code if you are interested in knowing how this works.
However, I recommend you try out the appropriate ACCDE version first.

All BE files and the ACCDE FE files are encrypted with password '***isladogs***'
The BE contains 1 'deep hidden' table, **tblBEData** (though it would work equally well with a standard table)
The FE contains 2 forms and a report

Both of the forms and the report have no saved record source.
Instead the record source is created using code when the object is loaded and destroyed when it is closed

As the forms/reports have no record source, there is no link to the BE table.
Therefore, there is no connection string accessible to that table (other than in the code)

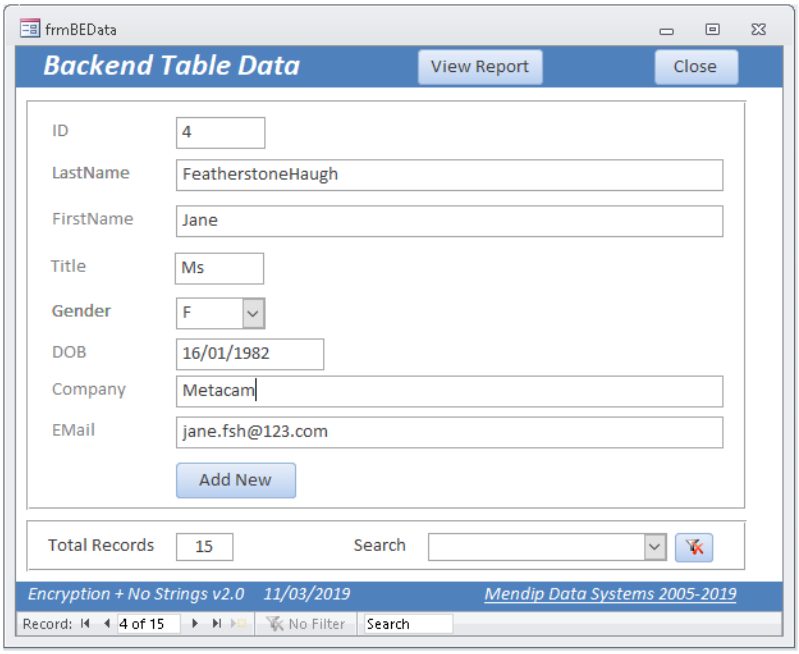
The BE table contains 8 fields of which all except the PK field have been encrypted

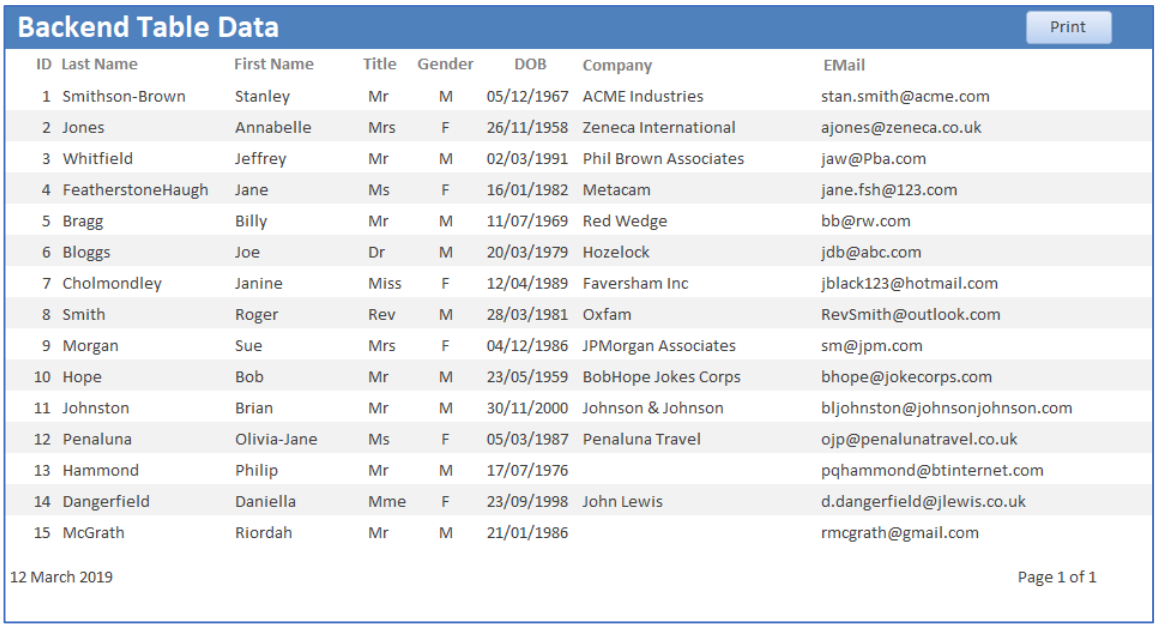



You need to be aware that encrypting date or number fields is problematic as the encryption cipher converts data into 'random' text strings. Hence, I have used a text field for the date of birth (DOB) field above

There is clearly little point encrypting fields with limited values e.g. Gender (M/F) or Title (Mr/Mrs/Ms/Miss etc)
However, I have done so here for completeness.

The data is visible, unencrypted in the main form and the report
The main form is fully editable and new records can be added – any changes will automatically be encrypted



**Backend Table Data** Print

| ID | Last Name | First Name | Title | Gender | DOB | Company | EMail |
|---|---|---|---|---|---|---|---|
| 1 | Smithson-Brown | Stanley | Mr | M | 05/12/1967 | ACME Industries | stan.smith@acme.com |
| 2 | Jones | Annabelle | Mrs | F | 26/11/1958 | Zeneca International | ajones@zeneca.co.uk |
| 3 | Whitfield | Jeffrey | Mr | M | 02/03/1991 | Phil Brown Associates | jaw@Pba.com |
| 4 | FeatherstoneHaugh | Jane | Ms | F | 16/01/1982 | Metacam | jane.fsh@123.com |
| 5 | Bragg | Billy | Mr | M | 11/07/1969 | Red Wedge | bb@rw.com |
| 6 | Bloggs | Joe | Dr | M | 20/03/1979 | Hozelock | jdb@abc.com |
| 7 | Cholmondley | Janine | Miss | F | 12/04/1989 | Faversham Inc | jblack123@hotmail.com |
| 8 | Smith | Roger | Rev | M | 28/03/1981 | Oxfam | RevSmith@outlook.com |
| 9 | Morgan | Sue | Mrs | F | 04/12/1986 | JPMorgan Associates | sm@jpm.com |
| 10 | Hope | Bob | Mr | M | 23/05/1959 | BobHope Jokes Corps | bhope@jokecorps.com |
| 11 | Johnston | Brian | Mr | M | 30/11/2000 | Johnson & Johnson | bljohnston@johnsonjohnson.com |
| 12 | Penaluna | Olivia-Jane | Ms | F | 05/03/1987 | Penaluna Travel | ojp@penalunatravel.co.uk |
| 13 | Hammond | Philip | Mr | M | 17/07/1976 | | pqhammond@btinternet.com |
| 14 | Dangerfield | Daniella | Mme | F | 23/09/1998 | John Lewis | d.dangerfield@jlewis.co.uk |
| 15 | McGrath | Riordah | Mr | M | 21/01/1986 | | rmcgrath@gmail.com |

12 March 2019 Page 1 of 1

NOTE:

1. In the unlocked BE version of this DEMO, I have deliberately left the table so it can be viewed and can therefore also be EDITED (NOT recommended!)
   As it contains encrypted data, editing those fields will lead to partly encrypted data being visible in the form

2. This approach is only worth considering if your data is **highly sensitive**
   Be aware that creating the forms will take much longer than usual as unbound controls must be used.

   The editable form contains 2 sets of each control in order to allow editing of the encrypted data.
   Unbound controls are used to display the decrypted data
   Doing this also means code needs to be added to each control to encrypt the entered data

3. If anyone manages to directly access the data tables, all they will see is encrypted data
   However, it is of course still possible for anyone with authorised access to the FE to print the data using reports or just take screenshots of the data.

4. If you decide to use this approach with **highly sensitive data** of your own, I would recommend that:
   - The BE database is given a different password to the FE.
     End users do NOT need to know the BE password
   - All data is stored in the BE. There should be no data tables in the FE
   - The Access BE file is stored securely on the server to which end users have no access
     Much better still - use SQL Server or similar for the BE database
   - Both the ribbon and navigation pane are removed from the FE. All interaction via forms ONLY
   - A strong 128-bit encryption method is used such as RC4 (or any other **secure** cipher)
     XOR encoding is NOT recommended as it is too easy to decode
   - Different encryption keys are used for each table.
     You could even use a different key for each field if it seems worth the additional coding effort needed
   - OPTIONAL - for additional security, the BE tables can be 'deep hidden' as in the locked versions of this DEMO. However, it isn't essential to do so if you ensure users have no means of accessing the BE

5. See the next page for some examples of the code used in this DEMO

6. If you do find any security issues in the ACCDE versions, please let me know.
   I do have several additional security measures that I use to deter unauthorised hacking . . .
   . . . but I can't give away all my secrets!

7. I decided not to encode the encryption key & BE password for this DEMO as it would just add confusion.

   In any case, as the code will be used in an encrypted ACCDE file, it is impossible for end users to retrieve this information. That is unless a specialist company is employed to reverse engineer the encryption of the ACCDE file itself. Reputable firms will only do that provided proof of ownership can be established which will not normally be possible.

8. Finally let me repeat a comment I have written many times previously in relation to security in Access:

   ***Access databases can NEVER be made 100% secure***
   ***A capable and determined hacker can break any Access database given sufficient time and motivation. However, by erecting various barriers, it is certainly possible to make the process so difficult and time consuming that it isn't normally worth attempting.***

   ***Access apps (or any applications) are only as secure as the weakest part of the security used***

I hope this idea will be interesting for others to use / adapt / improve.

I've used both encryption and the 'no strings' approach for particularly sensitive data but have never felt it necessary to do so for a whole database. I'll leave others to decide how practical this would be for an entire application.

If you have any questions about this approach, please email me at info@mendipdatasystems.co.uk

*Colin Riddington* *Mendip Data Systems* *12/03/2019*

**Example Code:**
When the form frmBEData or the report are opened, the record source is set using

```
Me.RecordSource = "SELECT tblBEData.PersonID, RC4([Title],'" & RMP_RC4_Key & "') AS XTitle," & _
    " RC4([LastName],'" & RMP_RC4_Key & "') AS XLastName, RC4([FirstName],'" & RMP_RC4_Key & "') AS XFirstName," & _
    " RC4([Gender],'" & RMP_RC4_Key & "') AS XGender, RC4([DOB],'" & RMP_RC4_Key & "') AS XDOB, " & _
    " RC4([Company],'" & RMP_RC4_Key & "') AS XCompany, RC4([EMail],'" & RMP_RC4_Key & "') AS XEMail" & _
    " FROM tblBEData IN '' [MS Access;PWD=isladogs;DATABASE=" & CurrentProject.Path & "\BE.accdb];"
```

For example, the code `RC4([LastName],'" & RMP_RC4_Key & "') AS XLastName` is DECRYPTING the LastName field and it is referenced here as XLastName to help clarify the data source.
Similarly, for each of the other encrypted fields

The encryption key constant is saved in the header section of the modEncryption module.
For additional security it could itself be encrypted . . . but using a different method!

The connection string to the BE table is the line:

```
FROM tblBEData IN '' [MS Access;PWD=isladogs;DATABASE=" & CurrentProject.Path & "\BE.accdb];"
```

Once again, the password itself could be encrypted … again using a different method.

When a field is updated in an unbound control, code like this is used to update the encrypted field in the BE table:

```
Db.Execute "UPDATE tblBEData IN '' [MS Access;PWD=isladogs;DATABASE=" & CurrentProject.Path & "\BE.accdb] " & _
        " SET tblBEData.FirstName = RC4('" & strText & "','" & RMP_RC4_Key & "')" & _
        " WHERE (((tblBEData.PersonID)= " & [Forms]![frmBEData].[PersonID] & "));"
```

Encryption and decryption are done using the same RC4 cipher and key. In other words, the encryption is reversible. This feature makes coding much easier to manage but it is also a weakness of RC4.
Other ciphers exist where the encryption is 'one-way', but coding will therefore be more complex still

When a new record is added, the following code is used:

```
    'get encrypted data for saving to table
    strLN = "RC4('" & Forms!frmBEData.txtLastName & "','" & RMP_RC4_Key & "')"
    strFN = "RC4('" & Forms!frmBEData.txtFirstName & "','" & RMP_RC4_Key & "')"
    strC = "RC4('" & Forms!frmBEData.txtCompany & "','" & RMP_RC4_Key & "')"
    strE = "RC4('" & Forms!frmBEData.txtEMail & "','" & RMP_RC4_Key & "')"
    strDB = "RC4('" & Forms!frmBEData.txtDOB & "','" & RMP_RC4_Key & "')"
    strG = "RC4('" & Forms!frmBEData.cboGender & "','" & RMP_RC4_Key & "')"
    strT = "RC4('" & Forms!frmBEData.txtTitle & "','" & RMP_RC4_Key & "')"

    'append new record
    CurrentDb.Execute "INSERT INTO tblBEData ( Title, LastName, FirstName, Gender,DOB, Company, EMail ) " & _
          "  IN '' [MS Access;PWD=isladogs;DATABASE=" & CurrentProject.Path & "\BE.accdb]" & _
          " SELECT " & strT & " AS Title, " & strLN & " AS LastName," & _
          " " & strFN & " AS FirstName, " & strG & " AS Gender," & _
          " " & strDB & " AS DOB, " & strC & " AS Company, " & strE & " AS EMail;"
```

When the form/report is closed the record source is destroyed using `Me.RecordSource = ""`