Checking Access File Security MDB / MDE vs ACCDB / ADDCE (Version 2)

This article describes a simple way of comparing the security of the **MDB** & **ACCDB** file formats together with the compiled versions **MDE & ACCDE.**

First of all, I created three very simple test **MDB** files in **Access 2003** format.

1. A backend file **BE.mdb** with one table **Table1.**
   The table had 4 fields (ID, Surname, FirstName, YearGroup) and 2 records Ann Jones / Ben Smith
   The BE file was encrypted with password ***dinsdale***.
2. A frontend file **FE.mdb.** I then linked this to **Table1** in the **BE** file
3. The frontend was copied, encrypted with the password ***MDS*** and saved as **FE - PWD.mdb**

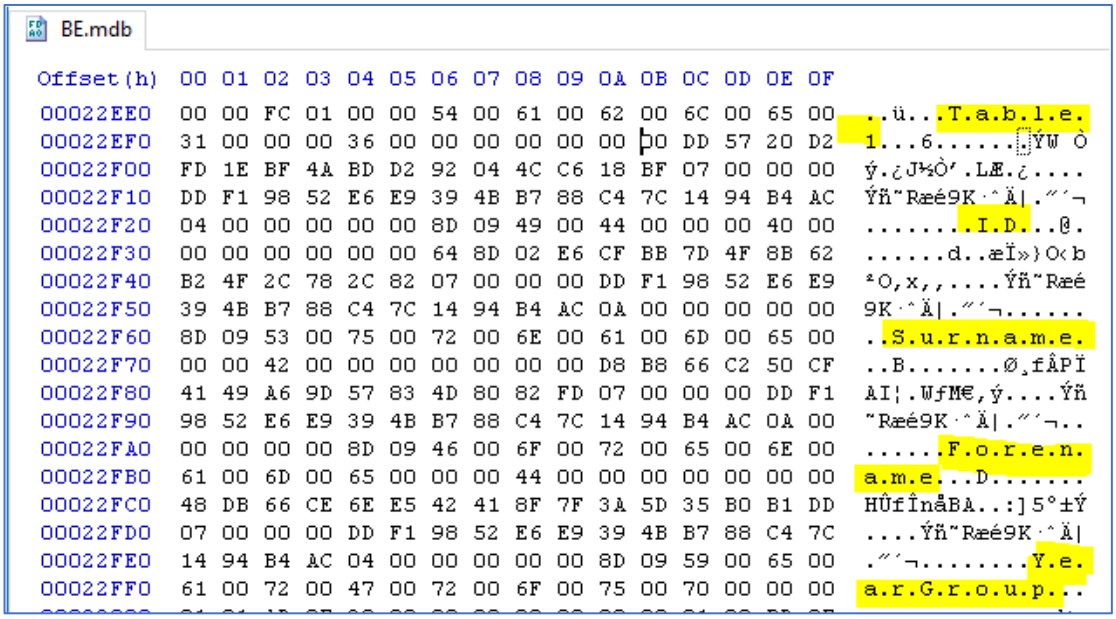
Both versions of the frontend were then saved as **MDE** files: **FE.mde** & **FE - PWD.mde**

In early versions such as **Access 97**, **password strings** could be directly read using a text or hex editor. This appalling security lapse was fixed in later versions.

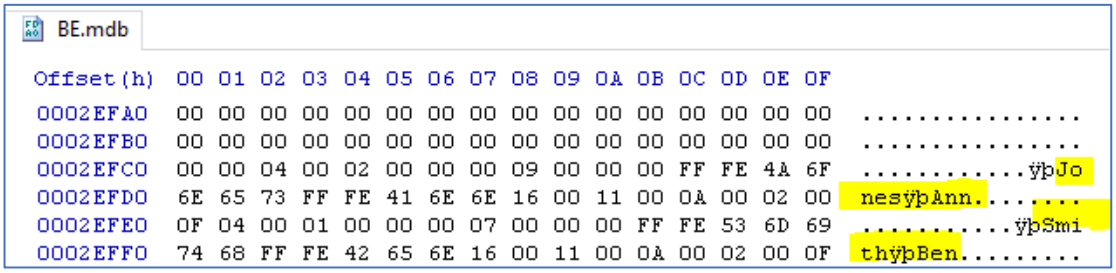
I checked both the frontend and backend files using a **hex editor** and searched for the two password strings. Although password data is stored in the file, the password information **CANNOT** be directly read using a hex editor in **Access 2003 MDB** files.

However, when I viewed the **BE** file using the hex editor, the details of **Table1** were found unencrypted in several locations. Both field names and the actual data were visible even though the file is password protected.

Password protected MDB BE file
Field names

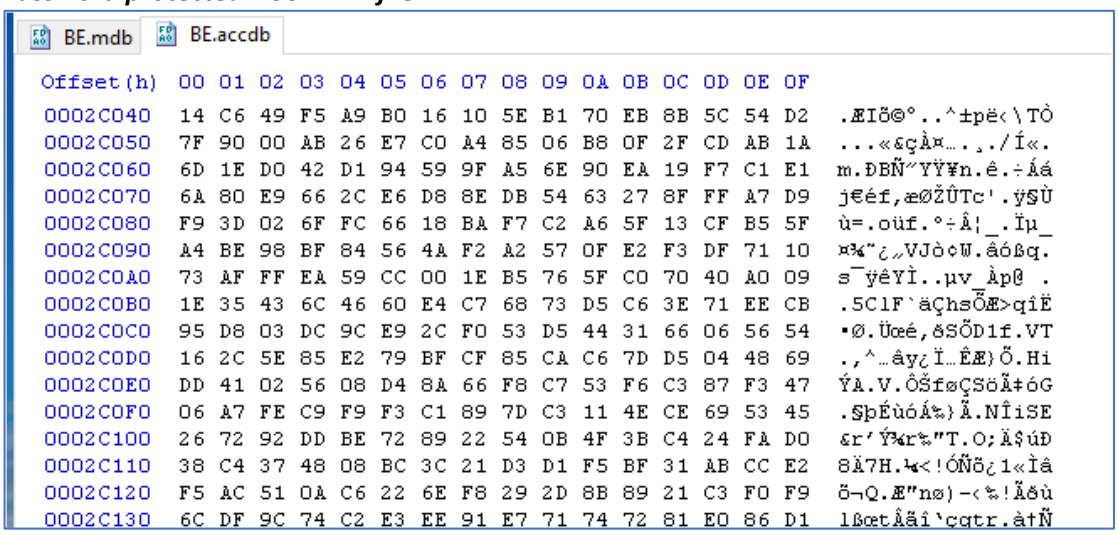


Data



For comparison, the **entire file** is **encrypted** in a **password protected ACCDB** file

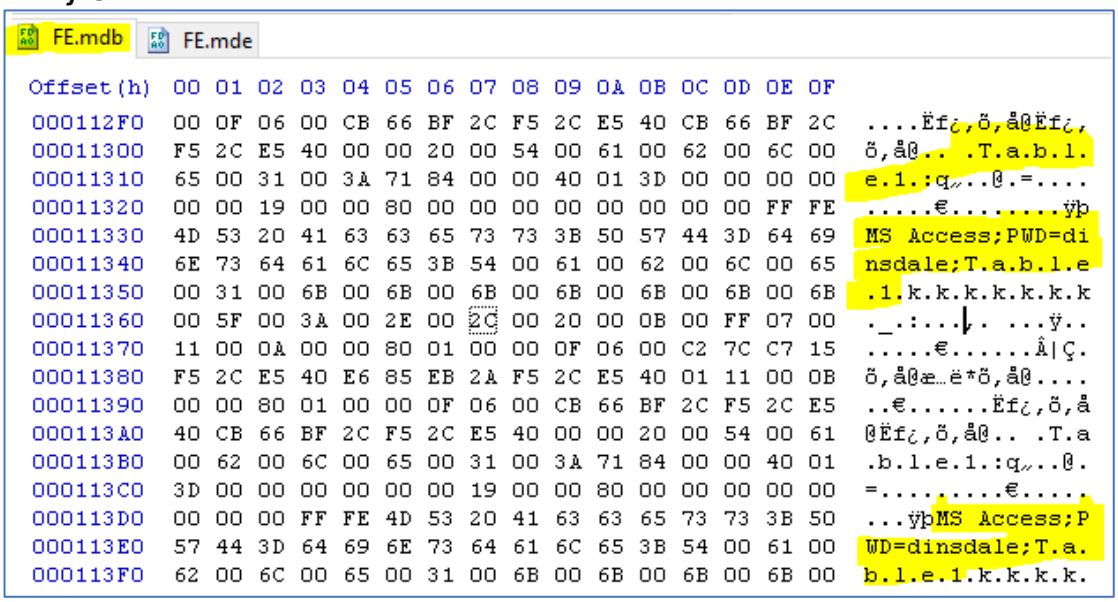
Password protected ACCDB BE file


BE.mdb BE.accdb

```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
0002C040   14 C6 49 F5 A9 B0 16 10 5E B1 70 EB 8B 5C 54 D2   .ÆIõ©°..^±pë‹\TÒ
0002C050   7F 90 00 AB 26 E7 C0 A4 85 06 B8 0F 2F CD AB 1A   ...«&çÀ¤….¸./Í«.
0002C060   6D 1E D0 42 D1 94 59 9F A5 6E 90 EA 19 F7 C1 E1   m.ÐBÑ"YŸ¥n.ê.÷Áá
0002C070   6A 80 E9 66 2C E6 D8 8E DB 54 63 27 8F FF A7 D9   j€éf,æØŽÛTc'.ÿ§Ù
0002C080   F9 3D 02 6F FC 66 18 BA F7 C2 A6 5F 13 CF B5 5F   ù=.oüf.°÷Â¦_.Ïµ_
0002C090   A4 BE 98 BF 84 56 4A F2 A2 57 0F E2 F3 DF 71 10   ¤¾˜¿„VJò¢W.âóßq.
0002C0A0   73 AF FF EA 59 CC 00 1E B5 76 5F C0 70 40 A0 09   s¯ÿêYÌ..µv_Àp@ .
0002C0B0   1E 35 43 6C 46 60 E4 C7 68 73 D5 C6 3E 71 EE CB   .5ClF`äÇhsÕÆ>qîË
0002C0C0   95 D8 03 DC 9C E9 2C F0 53 D5 44 31 66 06 56 54   •Ø.Üœé,ðSÕD1f.VT
0002C0D0   16 2C 5E 85 E2 79 BF CF 85 CA C6 7D D5 04 48 69   .,^…ây¿Ï…ÊÆ}Õ.Hi
0002C0E0   DD 41 02 56 08 D4 8A 66 F8 C7 53 F6 C3 87 F3 47   ÝA.V.ÔŠføÇSöÃ‡óG
0002C0F0   06 A7 FE C9 F9 F3 C1 89 7D C3 11 4E CE 69 53 45   .§þÉùóÁ‰}Ã.NÎiSE
0002C100   26 72 92 DD BE 72 89 22 54 0B 4F 3B C4 24 FA D0   &r'Ý¾r‰"T.O;Ä$úÐ
0002C110   38 C4 37 48 08 BC 3C 21 D3 D1 F5 BF 31 AB CC E2   8Ä7H.¼<!ÓÑõ¿1«Ìâ
0002C120   F5 AC 51 0A C6 22 6E F8 29 2D 8B 89 21 C3 F0 F9   õ¬Q.Æ"nø)-‹‰!Ãðù
0002C130   6C DF 9C 74 C2 E3 EE 91 E7 71 74 72 81 E0 86 D1   lßœtÂãî‘çqtr.à†Ñ
```

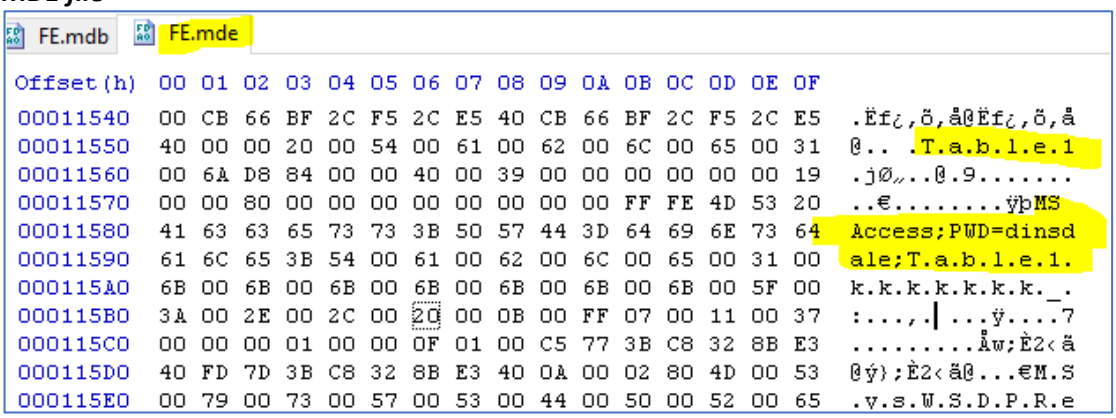
Next, I wanted to see how secure the **backend** data would be from someone who didn't know the **BE** password. To do so, each **frontend** file was then checked using a **hex editor** looking specifically for the BE password text strings ***PWD*** and ***dinsdale.*** These were found in several locations:

MDB file


FE.mdb FE.mde

```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
000112F0   00 0F 06 00 CB 66 BF 2C F5 2C E5 40 CB 66 BF 2C   ....Ëf¿,õ,å@Ëf¿,
00011300   F5 2C E5 40 00 00 20 00 54 00 61 00 62 00 6C 00   õ,å@.. .T.a.b.l.
00011310   65 00 31 00 3A 71 84 00 00 40 01 3D 00 00 00 00   e.1.:q„..@.=....
00011320   00 00 19 00 00 80 00 00 00 00 00 00 00 00 FF FE   .....€........ÿþ
00011330   4D 53 20 41 63 63 65 73 73 3B 50 57 44 3D 64 69   MS Access;PWD=di
00011340   6E 73 64 61 6C 65 3B 54 00 61 00 62 00 6C 00 65   nsdale;T.a.b.l.e
00011350   00 31 00 6B 00 6B 00 6B 00 6B 00 6B 00 6B 00 6B   .1.k.k.k.k.k.k.k
00011360   00 5F 00 3A 00 2E 00 2C 00 20 00 0B 00 FF 07 00   ._.:...,. ...ÿ..
00011370   11 00 0A 00 00 80 01 00 00 0F 06 00 C2 7C C7 15   .....€.......Â|Ç.
00011380   F5 2C E5 40 E6 85 EB 2A F5 2C E5 40 01 11 00 0B   õ,å@æ…ë*õ,å@....
00011390   00 00 80 01 00 00 0F 06 00 CB 66 BF 2C F5 2C E5   ..€.......Ëf¿,õ,å
000113A0   40 CB 66 BF 2C F5 2C E5 40 00 00 20 00 54 00 61   @Ëf¿,õ,å@.. .T.a
000113B0   00 62 00 6C 00 65 00 31 00 3A 71 84 00 00 40 01   .b.l.e.1.:q„..@.
000113C0   3D 00 00 00 00 00 00 19 00 00 80 00 00 00 00 00   =.........€.....
000113D0   00 00 00 FF FE 4D 53 20 41 63 63 65 73 73 3B 50   ...ÿþMS Access;P
000113E0   57 44 3D 64 69 6E 73 64 61 6C 65 3B 54 00 61 00   WD=dinsdale;T.a.
000113F0   62 00 6C 00 65 00 31 00 6B 00 6B 00 6B 00 6B 00   b.l.e.1.k.k.k.k.
```

MDE file

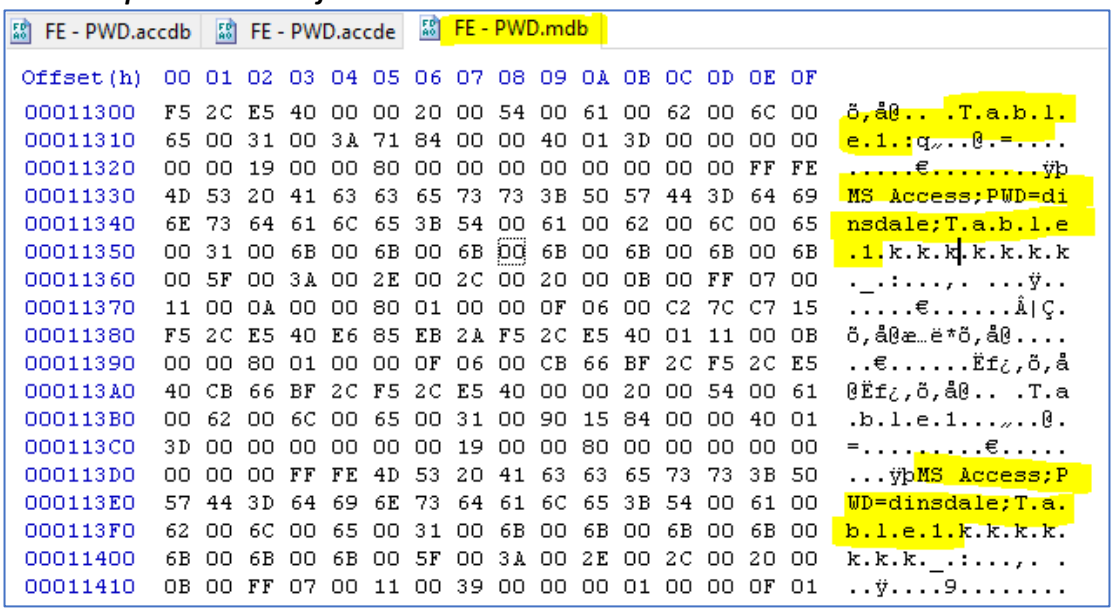

FE.mdb FE.mde

```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00011540   00 CB 66 BF 2C F5 2C E5 40 CB 66 BF 2C F5 2C E5   .Ëf¿,õ,å@Ëf¿,õ,å
00011550   40 00 00 20 00 54 00 61 00 62 00 6C 00 65 00 31   @.. .T.a.b.l.e.1
00011560   00 6A D8 84 00 00 40 00 39 00 00 00 00 00 00 19   .jØ„..@.9.......
00011570   00 00 80 00 00 00 00 00 00 00 00 00 FF FE 4D 53   ..€.........ÿþMS
00011580   41 63 63 65 73 73 3B 50 57 44 3D 64 69 6E 73 64   Access;PWD=dinsd
00011590   61 6C 65 3B 54 00 61 00 62 00 6C 00 65 00 31 00   ale;T.a.b.l.e.1.
000115A0   6B 00 6B 00 6B 00 6B 00 6B 00 6B 00 6B 00 5F 00   k.k.k.k.k.k.k._.
000115B0   3A 00 2E 00 2C 00 20 00 0B 00 FF 07 00 11 00 37   :...,. ...ÿ....7
000115C0   00 00 00 01 00 00 0F 01 00 C5 77 3B C8 32 8B E3   .........Åw;È2‹ã
000115D0   40 FD 7D 3B C8 32 8B E3 40 0A 00 02 80 4D 00 53   @ý};È2‹ã@...€M.S
000115E0   00 79 00 73 00 57 00 53 00 44 00 50 00 52 00 65   .y.s.W.S.D.P.R.e
```

As you can see, ***converting the file to MDE format does not encrypt the linked table name or the password needed to open the BE file. The BE file is completely insecure***

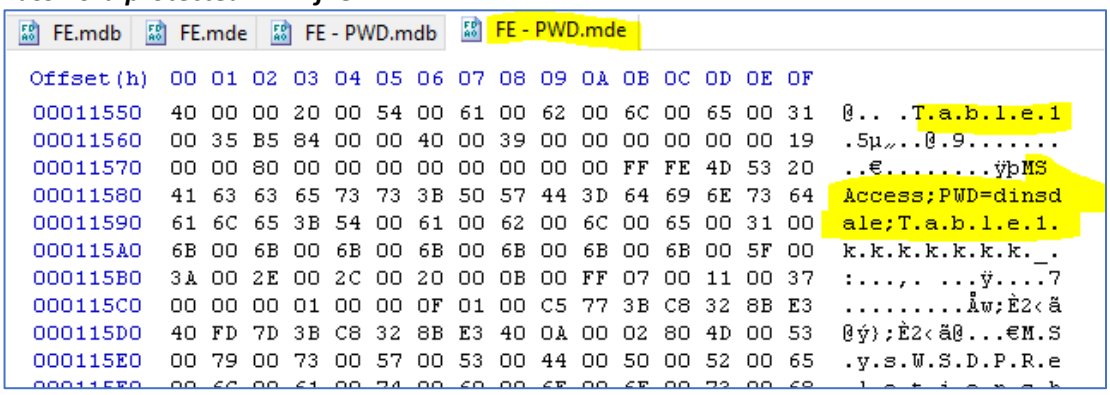
Next, I checked whether adding a password to the FE had any effect

Password protected MDB file


FE - PWD.accdb | FE - PWD.accde | FE - PWD.mdb

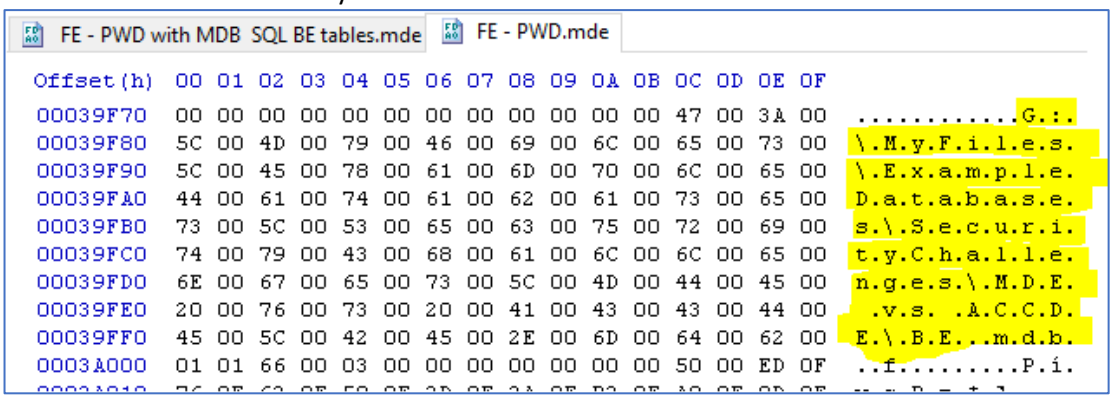
Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00011300	F5	2C	E5	40	00	00	20	00	54	00	61	00	62	00	6C	00	õ,å@.. .T.a.b.l.
00011310	65	00	31	00	3A	71	84	00	00	40	01	3D	00	00	00	00	e.1.:q„..@.=....
00011320	00	00	19	00	00	80	00	00	00	00	00	00	00	00	FF	FE	.....€.....ÿþ
00011330	4D	53	20	41	63	63	65	73	73	3B	50	57	44	3D	64	69	MS Access;PWD=di
00011340	6E	73	64	61	6C	65	3B	54	00	61	00	62	00	6C	00	65	nsdale;T.a.b.l.e
00011350	00	31	00	6B	00	6B	00	6B	00	6B	00	6B	00	6B	00	6B	.1.k.k.k.k.k.k.k
00011360	00	5F	00	3A	00	2E	00	2C	00	20	00	0B	00	FF	07	00	._.:...,. ...ÿ..
00011370	11	00	0A	00	00	80	01	00	00	0F	06	00	C2	7C	C7	15	.....€.....Â|Ç.
00011380	F5	2C	E5	40	E6	85	EB	2A	F5	2C	E5	40	01	11	00	0B	õ,å@æ…ë*õ,å@....
00011390	00	00	80	01	00	00	0F	06	00	CB	66	BF	2C	F5	2C	E5	..€.....Ëf¿,õ,å
000113A0	40	CB	66	BF	2C	F5	2C	E5	40	00	00	20	00	54	00	61	@Ëf¿,õ,å@.. .T.a
000113B0	00	62	00	6C	00	65	00	31	00	90	15	84	00	00	40	01	.b.l.e.1...„..@.
000113C0	3D	00	00	00	00	00	00	19	00	00	80	00	00	00	00	00	=.....€.....
000113D0	00	00	00	FF	FE	4D	53	20	41	63	63	65	73	73	3B	50	...ÿþMS Access;P
000113E0	57	44	3D	64	69	6E	73	64	61	6C	65	3B	54	00	61	00	WD=dinsdale;T.a.
000113F0	62	00	6C	00	65	00	31	00	6B	00	6B	00	6B	00	6B	00	b.l.e.1.k.k.k.k.
00011400	6B	00	6B	00	6B	00	5F	00	3A	00	2E	00	2C	00	20	00	k.k.k._.:...,. .
00011410	0B	00	FF	07	00	11	00	39	00	00	00	01	00	00	0F	01	..ÿ....9........

Password protected MDE file


FE.mdb | FE.mde | FE - PWD.mdb | FE - PWD.mde

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00011550	40	00	00	20	00	54	00	61	00	62	00	6C	00	65	00	31	@.. .T.a.b.l.e.1
00011560	00	35	B5	84	00	00	40	00	39	00	00	00	00	00	00	19	.5µ„..@.9.......
00011570	00	00	80	00	00	00	00	00	00	00	00	FF	FE	4D	53	20	..€.....ÿþMS
00011580	41	63	63	65	73	73	3B	50	57	44	3D	64	69	6E	73	64	Access;PWD=dinsd
00011590	61	6C	65	3B	54	00	61	00	62	00	6C	00	65	00	31	00	ale;T.a.b.l.e.1.
000115A0	6B	00	6B	00	6B	00	6B	00	6B	00	6B	00	6B	00	5F	00	k.k.k.k.k.k.k._.
000115B0	3A	00	2E	00	2C	00	20	00	0B	00	FF	07	00	11	00	37	:...,. ...ÿ....7
000115C0	00	00	00	01	00	00	0F	01	00	C5	77	3B	C8	32	8B	E3	.........Åw;È2‹ã
000115D0	40	FD	7D	3B	C8	32	8B	E3	40	0A	00	02	80	4D	00	53	@ý};È2‹ã@...€M.S
000115E0	00	79	00	73	00	57	00	53	00	44	00	50	00	52	00	65	.y.s.W.S.D.P.R.e

The **file location** is also easily found:


FE - PWD with MDB SQL BE tables.mde | FE - PWD.mde

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00039F70	00	00	00	00	00	00	00	00	00	00	00	00	47	00	3A	00	..........G.:.
00039F80	5C	00	4D	00	79	00	46	00	69	00	6C	00	65	00	73	00	\.M.y.F.i.l.e.s.
00039F90	5C	00	45	00	78	00	61	00	6D	00	70	00	6C	00	65	00	\.E.x.a.m.p.l.e.
00039FA0	44	00	61	00	74	00	61	00	62	00	61	00	73	00	65	00	D.a.t.a.b.a.s.e.
00039FB0	73	00	5C	00	53	00	65	00	63	00	75	00	72	00	69	00	s.\.S.e.c.u.r.i.
00039FC0	74	00	79	00	43	00	68	00	61	00	6C	00	6C	00	65	00	t.y.C.h.a.l.l.e.
00039FD0	6E	00	67	00	65	00	73	00	5C	00	4D	00	44	00	45	00	n.g.e.s.\.M.D.E.
00039FE0	20	00	76	00	73	00	20	00	41	00	43	00	43	00	44	00	.v.s. .A.C.C.D.
00039FF0	45	00	5C	00	42	00	45	00	2E	00	6D	00	64	00	62	00	E.\.B.E...m.d.b.
0003A000	01	01	66	00	03	00	00	00	00	00	00	00	50	00	ED	0F	..f.........P.í.

The above screenshots show that ***adding a password to the frontend does not encrypt the linked table name or the password needed to open the BE file. The BE file remains completely insecure***

For comparison, I converted all the test files to **ACCDB** format but otherwise **identical to those above**

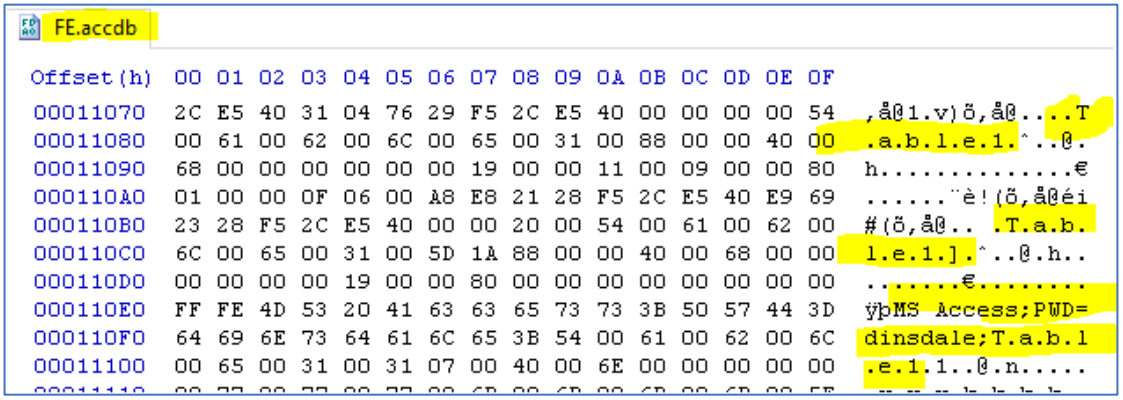
1. A backend file **BE.accdb** with one table **Table1**. The file was encrypted with password ***dinsdale***.
2. A frontend file **FE.accdb.** I then linked this to **Table1** in the **BE** file
3. The frontend was copied, encrypted with the password ***MDS*** and saved as **FE - PWD.accdb**
   In **ACCDB** files, much stronger 128-bit encryption is used which encrypts the **entire file**

   NOTE:
   For that reason, zipping an encrypted ACCDB/ACCDE file will have little effect on file size
   By comparison zipping a password protected MDB/MDE file does reduce file size

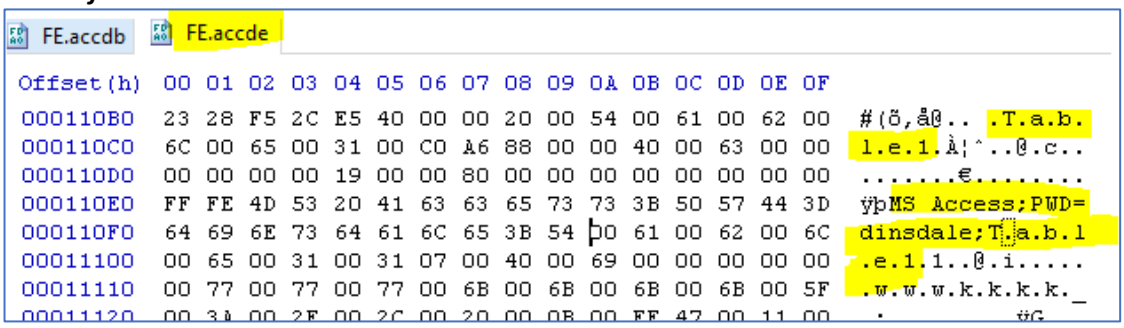
Both versions of the frontend were then saved as **ACCDE** files: **FE.accde** & **FE - PWD.accde**

The same tests were run as for the MDB / MDE files

ACCDB file



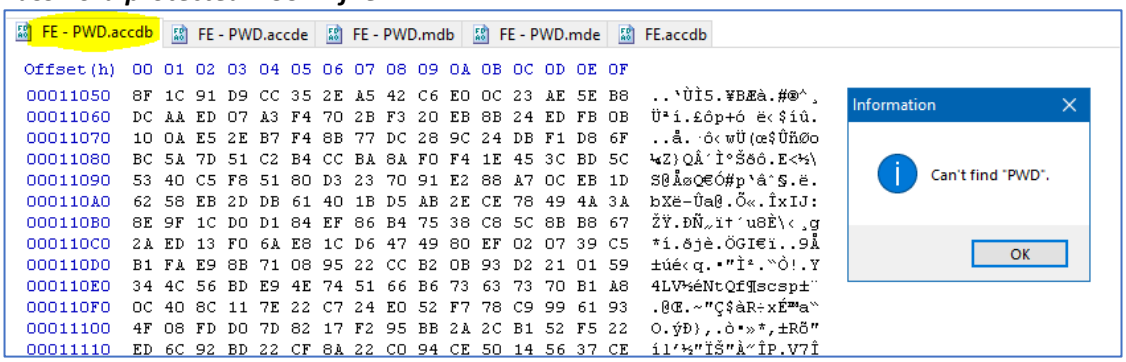
ACCDE file



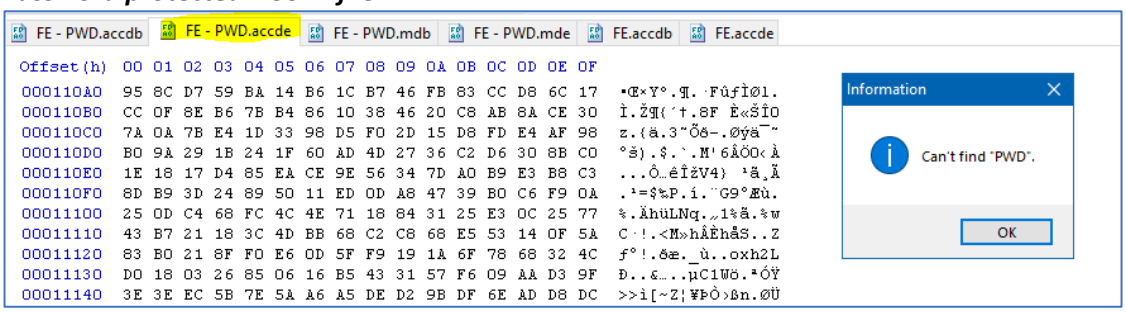
Neither ACCDB nor ACCDE files offer any additional security in terms of the BE file.
As a result, this remains insecure

For comparison, here are the results using password protected (encrypted) **ACCDB / ACCDE** files

Password protected ACCDB file



Password protected ACCDE file



The password protected ACCDB & ACCDE files encrypt the entire file including BOTH the linked table name and password information

Therefore, to keep the data in a BE file secure, the FE file needs to be in ACCDB/ACCDE format and encrypted using password protection.

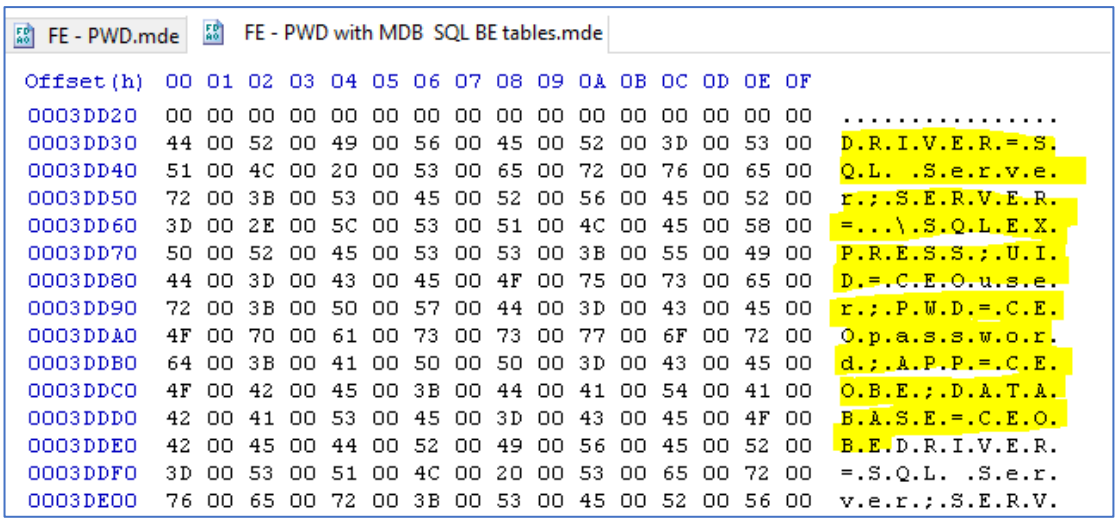
Linking to other sources

ACCDB/ACCDE files can also link to other data sources including **MDB** & **SQL Server** tables.
Providing the **FE is encrypted**, the **details of these link tables are also secure**.

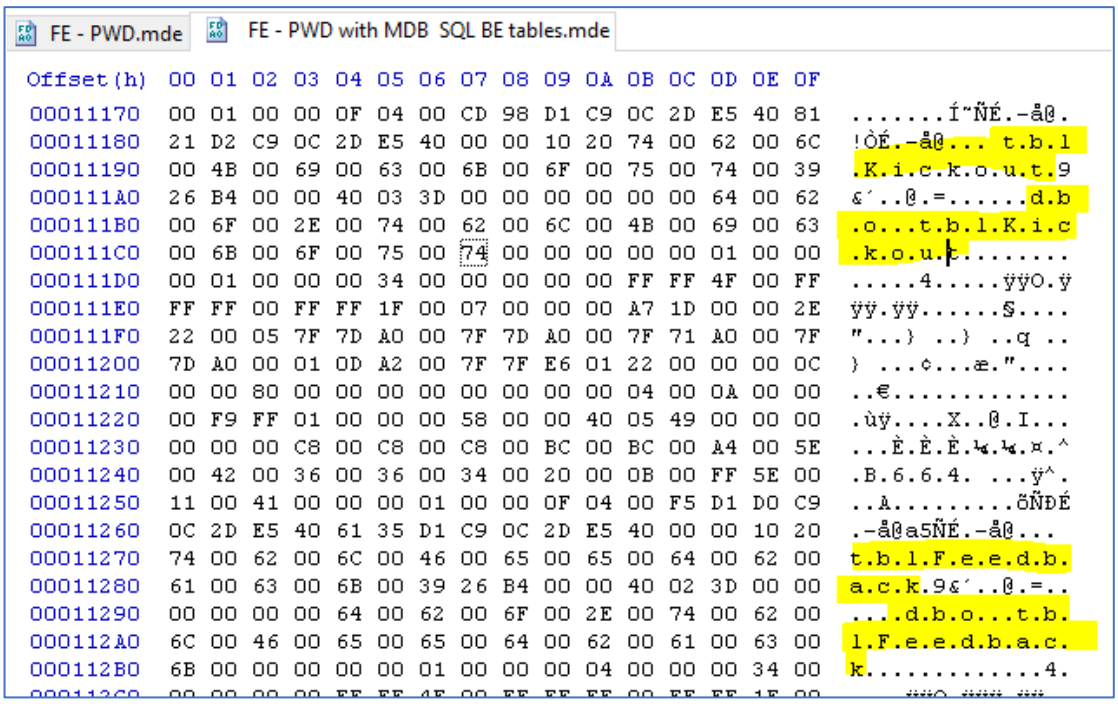
MDB files cannot link to **ACCDB** files. However, they can of course be linked to **SQL** tables including those created in later versions of **SQL Server** providing the datatypes can be recognised by the **MDB** file.

SQL Server is a much more secure environment so does that mean **linked SQL Server** tables are also secure in an **MDB** file? Sadly, the answer is, not really!
I added links to several **SQL** tables in a **password protected MDE**. As with any other linked tables in **MDB/MDE or ACCDB/ACCDE** files, the **connection strings are visible** though the **password is NOT shown**.

However, using a hex editor, the **full SQL connection string including the password is shown** for each table

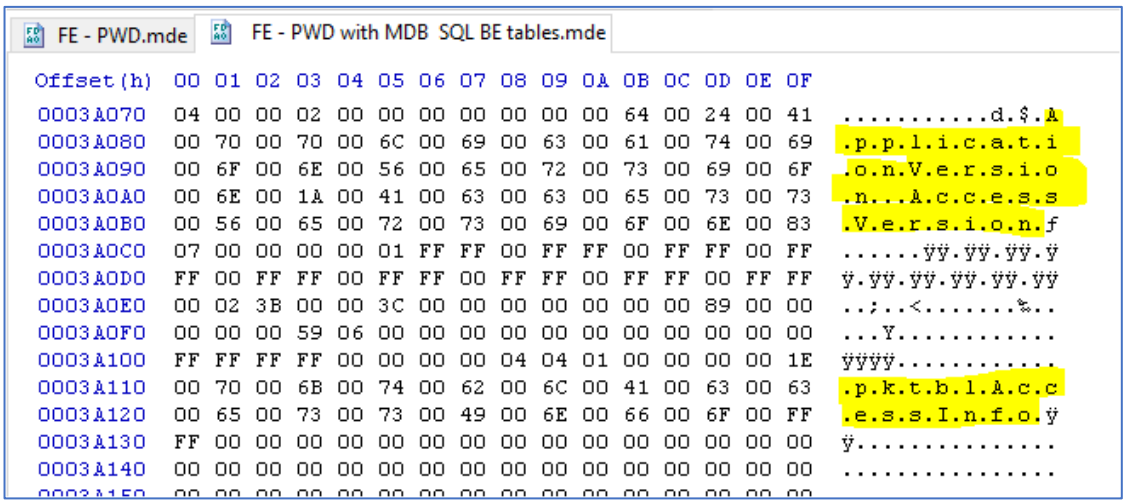


The **SQL table names** are also easily found. For example:



Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00011170	00	01	00	00	0F	04	00	CD	98	D1	C9	0C	2D	E5	40	81	.....Í˜ÑÉ.-å@.
00011180	21	D2	C9	0C	2D	E5	40	00	00	10	20	74	00	62	00	6C	!ÒÉ.-å@... t.b.l
00011190	00	4B	00	69	00	63	00	6B	00	6F	00	75	00	74	00	39	.K.i.c.k.o.u.t.9
000111A0	26	B4	00	00	40	03	3D	00	00	00	00	00	00	64	00	62	&´..@.=.....d.b
000111B0	00	6F	00	2E	00	74	00	62	00	6C	00	4B	00	69	00	63	.o...t.b.l.K.i.c
000111C0	00	6B	00	6F	00	75	00	74	00	00	00	00	00	01	00	00	.k.o.u.t........
000111D0	00	01	00	00	00	34	00	00	00	00	00	FF	FF	4F	00	FF	.....4.....ÿÿO.ÿ
000111E0	FF	FF	00	FF	FF	1F	00	07	00	00	00	A7	1D	00	00	2E	ÿÿ.ÿÿ.....§....
000111F0	22	00	05	7F	7D	A0	00	7F	7D	A0	00	7F	71	A0	00	7F	"...} ..} ..q ..
00011200	7D	A0	00	01	0D	A2	00	7F	7F	E6	01	22	00	00	00	0C	} ...¢...æ."....
00011210	00	00	80	00	00	00	00	00	00	00	00	04	00	0A	00	00	..€.............
00011220	00	F9	FF	01	00	00	00	58	00	00	40	05	49	00	00	00	.ùÿ....X..@.I...
00011230	00	00	00	C8	00	C8	00	C8	00	BC	00	BC	00	A4	00	5E	...È.È.È.¼.¼.¤.^
00011240	00	42	00	36	00	36	00	34	00	20	00	0B	00	FF	5E	00	.B.6.6.4. ...ÿ^.
00011250	11	00	41	00	00	00	01	00	00	0F	04	00	F5	D1	D0	C9	..A.........õÑÐÉ
00011260	0C	2D	E5	40	61	35	D1	C9	0C	2D	E5	40	00	00	10	20	.-å@a5ÑÉ.-å@...
00011270	74	00	62	00	6C	00	46	00	65	00	65	00	64	00	62	00	t.b.l.F.e.e.d.b.
00011280	61	00	63	00	6B	00	39	26	B4	00	00	40	02	3D	00	00	a.c.k.9&´..@.=..
00011290	00	00	00	00	64	00	62	00	6F	00	2E	00	74	00	62	00	....d.b.o...t.b.
000112A0	6C	00	46	00	65	00	65	00	64	00	62	00	61	00	63	00	l.F.e.e.d.b.a.c.
000112B0	6B	00	00	00	00	00	01	00	00	00	04	00	00	00	34	00	k.............4.

The **SQL field names** are also listed. For example:



Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
0003A070	04	00	00	02	00	00	00	00	00	00	00	64	00	24	00	41	..........d.$.A
0003A080	00	70	00	70	00	6C	00	69	00	63	00	61	00	74	00	69	.p.p.l.i.c.a.t.i
0003A090	00	6F	00	6E	00	56	00	65	00	72	00	73	00	69	00	6F	.o.n.V.e.r.s.i.o
0003A0A0	00	6E	00	1A	00	41	00	63	00	63	00	65	00	73	00	73	.n...A.c.c.e.s.s
0003A0B0	00	56	00	65	00	72	00	73	00	69	00	6F	00	6E	00	83	.V.e.r.s.i.o.n.ƒ
0003A0C0	07	00	00	00	00	01	FF	FF	00	FF	FF	00	FF	FF	00	FF	......ÿÿ.ÿÿ.ÿÿ.ÿ
0003A0D0	FF	00	FF	FF	00	FF	FF	00	FF	FF	00	FF	FF	00	FF	FF	ÿ.ÿÿ.ÿÿ.ÿÿ.ÿÿ.ÿÿ
0003A0E0	00	02	3B	00	00	3C	00	00	00	00	00	00	00	89	00	00	..;..<.......‰..
0003A0F0	00	00	00	59	06	00	00	00	00	00	00	00	00	00	00	00	...Y............
0003A100	FF	FF	FF	FF	00	00	00	00	04	04	01	00	00	00	00	1E	ÿÿÿÿ............
0003A110	00	70	00	6B	00	74	00	62	00	6C	00	41	00	63	00	63	.p.k.t.b.l.A.c.c
0003A120	00	65	00	73	00	73	00	49	00	6E	00	66	00	6F	00	FF	.e.s.s.I.n.f.o.ÿ
0003A130	FF	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	ÿ..............
0003A140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	....

It is somewhat of a relief to state that, the **linked SQL table data** DOES appear to be encrypted.
However, as the **full SQL connection string and table names** have been exposed, it may not be that difficult for users to gain access to the **SQL datafile** itself.

Password Recovery utilities

Password recovery utilities can be useful if you forget the password to your own applications but can just as easily be used for other applications that are not your own work.

Running a quick online check will soon find several utilities (mostly free) that will crack passwords on MDB files. This is usually very quick as the level of encryption is relatively weak

By comparison, I am only aware of one commercial utility that can crack ACCDB passwords.
This is done by 'brute force' trying every possible combination of characters until one is successful.
If a strong password is used, the process can take MANY hours and is therefore unlikely to be worth the effort in most cases.

Obviously, there are many other factors involved in making databases as secure as possible including:

a) The use of workgroup security for MDB/MDE files
b) Protecting the location of the BE file by saving on a secure network and limiting user permissions to the folder
c) Using a SQL server backend database
d) Using strong passwords that cannot easily be hacked

However, an Access database can NEVER be made 100% secure. A capable and determined hacker can break any Access database given sufficient time and determination

---

I would welcome feedback on the information provided in this article

Colin Riddington *Mendip Data Systems* *Updated 24/09/2018*