Improve Security in Access Databases

First published 05/11/2018 in response to a question by Utter Access member Frank Ruperto
UA: http://www.utteraccess.com/forum/index.php?s=&showtopic=2049623&view=findpost&p=2701306

Also published at:
Access World Forums: https://www.access-programmers.co.uk/forums/showthread.php?t=302287
Isladogs on Access Website: https://www.isladogs.co.uk/improve-security/

This article is a companion to the following items at the **Isladogs on Access** website:
Access File Security, Purpose of System Tables, Security Challenges

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.

This article summarises some of the main steps that developers should do to improve security
Make sure you make several backups as you add each step as its only too easy to lock yourself out!

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

1. Split your databases

In a **multiuser environment**, Access applications should ALWAYS be split with:

a) **Shared backend file(s) on the server and only containing tables**
Users should connect to the **backend files** on a **LOCAL area network (LAN)** - **NOT on a WIDE area network (WAN)** as it is likely to be slow and cause issues
NEVER use a cloud-based location such as **OneDrive** or **Dropbox** as this increases the risk of corruption significantly
Similarly, users should **NEVER run a split database** when connected to the server **using a WIRELESS connection**

b) All **forms, reports, macros and module code** in a **frontend database**
Each user needs to have their **own copy of the frontend database located on their own workstation**
It is **ESSENTIAL** that users **do NOT share the same copy of the frontend** as **corruption WILL occur**

2. Backend file(s) - BE

- Hide all tables
- Hide navigation pane
- Disable various Access options e.g. UNTICK all the following:
  - allow full menus
  - allow default shortcut menus
  - use Access special keys (this disables the SHIFT bypass – but see below)
- Do ONE of the following:
  - Hide the ribbon
  - Remove Privacy Options from the File menu (otherwise users can undo all the above)
- Encrypt using a strong password & do NOT store in anywhere in the application (FE or BE)

NOTE: There is no benefit in using an ACCDE file if your BEs only contain tables

3. Frontend file (FE)

- Do everything listed above in the FE as well – no ribbon / no nav pane / shift bypass disabled etc
- ALWAYS use an ACCDE so your code is compiled and therefore not available to end users
- Rename as ACCDR and add code to prevent it being used if the file type is changed back again
- Use strong encryption with a different password to those used in the BE file(s)
- A VBA project password can also be used but it is easy to disable these. **Don't rely on this alone!**
- If users login with user name & password
    - if possible don't store the passwords in the db - use Active Directory
    - If you must store passwords in the DB, make sure you use strong 128-bit encryption such as RC4 (rather than just encoding as that is usually very easy to decode) and do NOT
        - store the cipher anywhere it can be viewed
        - include decryption code in the DB
    - if you must store the encryption cipher in the DB, encrypt that as well (using a different method)

4. Other security features include

- Add registry strings that are checked by the FE & if altered, prevent it running
- Using activation type code to 'lock' the application to a particular workstation and prevent copying
- Prevent users having direct access to the folders containing BE files
- Consider removing the taskbar whilst the app is in use but, if so, ensure that is reversed afterwards

IMPORTANT:

It is possible to do ALL the following (**and more**) externally from another Access application or any VBA enabled application such as Excel

- re-enable the **s**hift **b**ypass **key**
- remove or modify the database password (if the original password is known)
- view the connection strings to the BE tables (which exposes the BE password!)
- change the start-up form e.g. to bypass the user login form
- bypass everything on first opening if the location isn't trusted

I am deliberately NOT going to explain how to do any of these things here

5. Security Challenges

All the above features (and more) are included in my various security challenges which are available from https://www.isladogs.co.uk/security-challenges/

Their purpose is partly fun but also to encourage users to improve their own security by looking into weaknesses in other apps

Each challenge is meant to be solvable so there are deliberate built-in weaknesses (as well as a couple of unintentional flaws included by mistake!)

6. RC4 Encryption

This provides very strong 128-bit encryption (good enough for use in Access) but is no longer considered secure enough for commercial databases storing information such as credit card data etc

7. File-based or server-based databases

Access is a file-based application which is fundamentally why it can never be 100% secure
If you have mission critical data, an Access BE is not the right solution.
Using a server-based BE such as SQL Server will significantly improve the security of your data.
However, it does require someone with a knowledge of SQL Server to maintain it.

8. Further reading:

Access File Security: https://www.isladogs.co.uk/compare-access-file-security/

Purpose of System Tables: https://www.isladogs.co.uk/purpose-of-system-tables/

RC4 Encryption: https://en.wikipedia.org/wiki/RC4