PawCom Application Note 110

Security Warnings in Office 2003 and Office 2007

Microsoft has been tightening up security in its Office applications starting with Office 2003 to help you from accidentally opening up an application with malicious VBA code. This added security may, by default, cause difficulties when you try to open Multiware tools in MS Access or MS Excel. This application note discusses how you can deal with Office security.

Security Levels

If you get a message "Security Warning: Unsafe expressions are not blocked", then your computer does not have Microsoft Jet 4.0 Service Pack 8 (SP8) or later installed. In that case you should install SP8. The rest of this note assumes that you have SP8 or later installed.

Microsoft has implemented for levels of security, which they describe as follows:

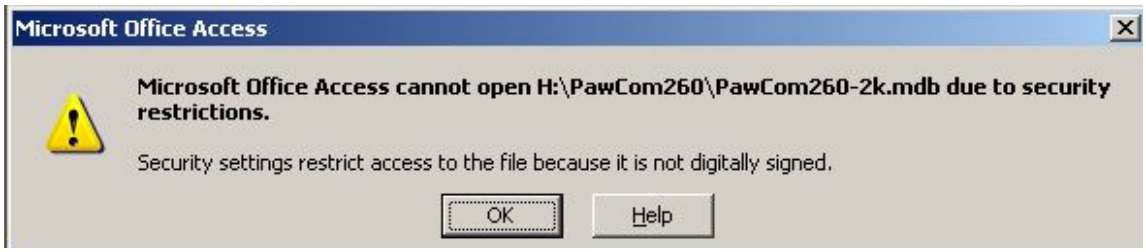
- Very high. Only macros installed in trusted locations will be allowed to run. All other signed and unsigned macros are disabled. (This level doesn't appear in MS Access.)
- High. Only signed macros from trusted sources will be allowed to run. Unsigned macros are automatically disabled.
- Medium. You can choose to run potentially unsafe macros. (Default value)
- Low (not recommended). You are not protected from potentially unsafe macros. Use this setting only if you have virus scanning software installed, or you have checked the safety of all documents you open.

You can modify the security setting in any of the Office applications by going to the menu bar and selecting *Tools : Macros : Security.*

So what does all this mean? When Microsoft says "macro" they mean essentially VBA code. If you have ANY code modules defined in your application then the security check is triggered. Since the MS Access front end for PawCom contains lots of VBA code, it will always trigger the security check. If you are opening an MS Excel spreadsheet and have added some VBA code, that will also trigger the security check.

> Note: If you are trying to open an Excel spreadsheet and you don't think it has any VBA code in it, you may have accidentally created a code module that is empty. The security check is triggered by the presence of modules even if they don't contain any code. You can get around this problem by simply deleting the module.

If the security level is set to high, the security check will display the following if you have VBA code:



If the security level is set to medium, the security check will display the following if you have VBA code:

If you click Cancel, MS Access is put in sandbox mode, where it will not run code expressions that are considered potentially dangerous because they could be used by maliciously written code to access drives, files, or other resources for which they do not have authorization. For the PawCom MS Access front end, this will prevent it from doing performing most of its useful functionality. If you click Open then the PawCom application will run normally.

If the security level is set to low, you won't see any security warning and the application will run normally.

The security checking is really designed to protect you against running dangerous applications that you have downloaded or that you have installed with some product you may not trust. For applications that you are developing you already trust the application because it is yours, so these security messages are just an annoyance.

So what can you do to prevent the security messages from popping up every time you run the application? The easiest thing to do is to set the security level to low. However, this is not a very safe thing to do because Office will use this security level for ALL applications of the same type it runs. You may trust an Access application you are working on, but do you want to trust all other Access applications on your system? A better approach is to digitally sign the application.

Digital Signatures

A digital signature on an application containing VBA code is like a wax seal on an envelope – it confirms that the file originated from the entity who signed it and that the code in the file has not been altered by anyone else. For High or Medium security settings a file signed by a trusted entity will be opened without any security settings. Microsoft Office 2003 and later use Microsoft Authenticode technology to enable you to digitally sign a file or VBA project by using a digital certificate.

Note that a digital signature does not apply to the entire database of an Access application. It covers only those parts of the database that could be modified to do malicious things, such as VBA code, macros, action queries, SQL pass-through queries, data definition queries, the ODBC connection string in queries, and properties of ActiveX controls. If any of these are modified after you sign the application, the digital signature will be removed and the files will not open under Medium or High security.

There are essentially three ways to obtain a digital certificate:

- Create a digital certificate for your own use. Such a certificate would only be used to sign an application for your own use or for sharing within a small workgroup.

- Obtain a digital certificate from your organization's internal certification authority. Some organizations may choose to have a security administrator or group act as its own certification authority and produce and distribute digital certificates by using tools such as Microsoft Certificate Server.
- Obtain a digital certificate from a commercial certification authority such as VeriSign, Inc.

For typical users of Multiware products the first option makes the most sense. You can easily modify the application and update the digital signature yourself.

Note: Since a digital signature becomes invalid as soon as you have made any significant change to the application, Multiware does not bother signing the original PawCom Access front end.

Creating a Digital Certificate for your own use

To create a digital certificate for your own use, you run the Create Digital Certificate utility (selfcert.exe) and enter information about yourself that will be stored in the certificate. Because a digital certificate created in this fashion isn't issued by a formal certification authority, it is called a *self*-signed certificate and VBA projects that are signed by using such a certificate are referred to as self-signed projects.

Search your system for the utility Selfcert.exe. It will be installed in the same folder where Office applications, which by default is something like c:\Program Files\Microsoft Office 2003\Office. If the utility isn't installed on your system, install it as follows:

1. In the Control Panel, double-click **Add/Remove Programs**.
2. On the **Install/Uninstall** tab click Microsoft Office 2003 and then click **Add/Remove**.
3. In the Microsoft Office Maintenance Mode dialog box click the **Add or Remove Features** button.
4. Expand Office Tools and set Digital Signature for **VBA Projects to Run from My Computer**.
5. Click **Update Now**. Selfcert.exe will be installed.

To create a self-signed digital certificate:

1. Run Selfcert.exe from My Computer or Windows Explorer.
2. In the **Your certificate's name** box, type an appropriate description for the certificate you want to create, then click **OK**.

Selfcert.exe will create and install a self-signed certificate that you can use to sign VBA projects on the current computer. To create a self-signed certificate to use on another computer, run Selfcert.exe on that computer.

When you sign a VBA project by using a self-signed certificate and if security is set to Medium or High, the first time you open the document containing the signed VBA project the Security Warning dialog box is displayed indicating that the certificate used to sign the VBA project hasn't been issued by a certification authority and shouldn't be trusted. Obviously you can safely trust a VBA project you have signed yourself by using a self-signed certificate, and if you do so the Security Warning dialog box won't be displayed the next time you open the application.

Signing Your VBA Project

Once you have your digital certificate installed you can sign the VBA project associated with an MS Access application. You may want to sign the project after you have developed and tested the application because any time code in a signed VBA project is modified in any way, its digital signature is removed.

To digitally sign your VBA project:

1. Open the application you want to sign.
2. Open the Visual Basic Editor.
3. On the **Tools** menu, click **Digital Signatures**.
4. Do one of the following:
   - If you haven't previously selected a digital certificate, or want to use another one, click **Choose**, select the certificate, and click **OK** twice.
   - Click OK to use the current certificate.

Note: If the application has been signed previously, clicking **Choose** and selecting a new digital certificate replaces the previous signature. To remove a signature from a previously signed project, click **Remove**.

Office 2007 and Trusted Directories

With Office 2007 Microsoft has provided an additional way to indicate an application is to be trusted, which is to specify anything in a particular directory is ok. This is probably the cleanest way of dealing
With the security because you don't need to bother with resigning an Access or Excel application each time you change the VBA code.

To specify a trusted directory go to the Trust Center screen and click the "Trusted locations" button. The form will allow you to add a new location to a list of trusted locations. Note that you can also specify that all subfolders of a location are to be trusted also. After specifying a trusted location for your application that contains macros you should be able to run it without being hassled by security any further.

Additional Sources

For more information about Digital Certificates and signing go to http://office.microsoft.com and http://msdn.microsoft.com and search for Digital Certificates.