Bacon Bits

Delicious bits of Excel and Access Training from DataPig Technologies
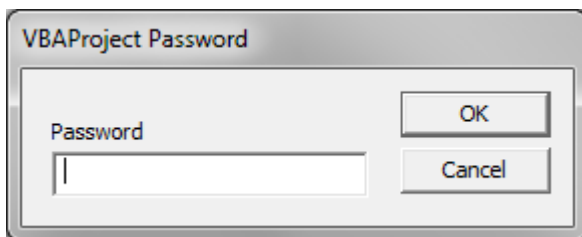
Home   Add-ins and Training

Hack into Password Protected VBA Projects

 datapig   October 8, 2014   Excel VBA, Worksheet Protection

 107 Comments

A few days ago, I had to pull out one of my old tricks to hack into a password protected VBA module. The file I was working on was protected by a former (evil) employee who, for some reason, did not leave the VBA password.

I'm notoriously lazy, so there is no way I'm retyping code that's already there. I'd rather hack into the workbook.

Today, I'll walk you through the steps I use to crack VBA passwords. I learned this trick a while back from one of my Excel Boot Camp students. I can't remember who. Sorry…whoever you are. Even though I can't remember your name, I value our friendship greatly.

Step 1: Download a Hex Editor

You'll need a Hex Editor. I use one called XVI32. Why this one? Because it's the one my mysterious friend showed me. I'm not really

Try these Add-ins!

Search for a Topic

Search

Recent Posts

Me vs Hungarian Notation

Create SQL Server Tables from Excel Data – Free Tool

Excel Against Humanity

Best Practices for Organizing VBA Modules

Pulling SSRS Data Directly into Excel with PowerPivot

Past Posts

Select Month   ▼

into Hex Editors, so I couldn't tell you if it's good or bad. I just know it does the trick. Download the XVI32 Hex Editor and have it ready to go.

.

Step 2: Save your Workbook as XLS and MAKE A BACKUP COPY

If you're working with an XLSM file, you'll need to save it as an XLS. Be sure you make a backup of your workbook before you start. Because you're diddling the insides of the workbook, there is always a chance you could royally screw up. You'll want a backup.

.

Step 3: Open Your Workbook in the Hex Editor

Fire up the Hex Editor you downloaded and then (within the editor) select and open your workbook. In the XVI32 editor, my workbook looks like this:
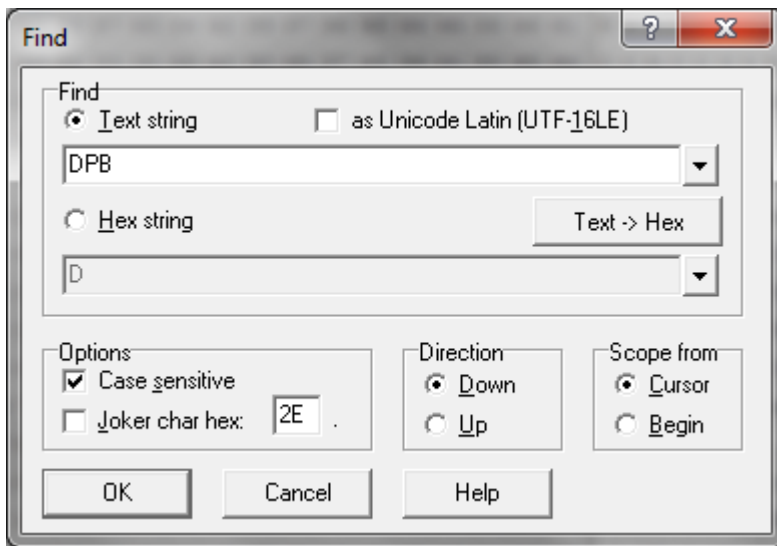

.
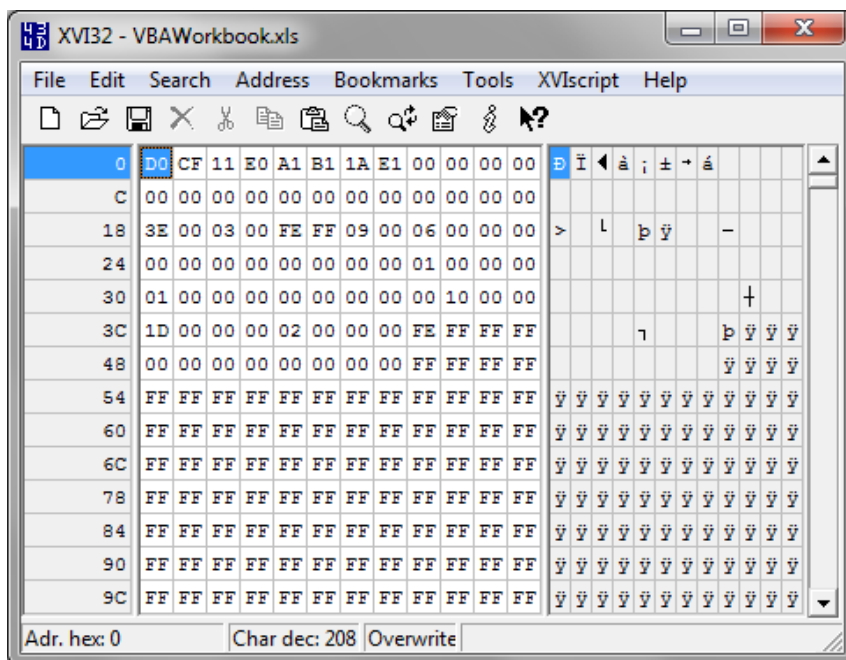
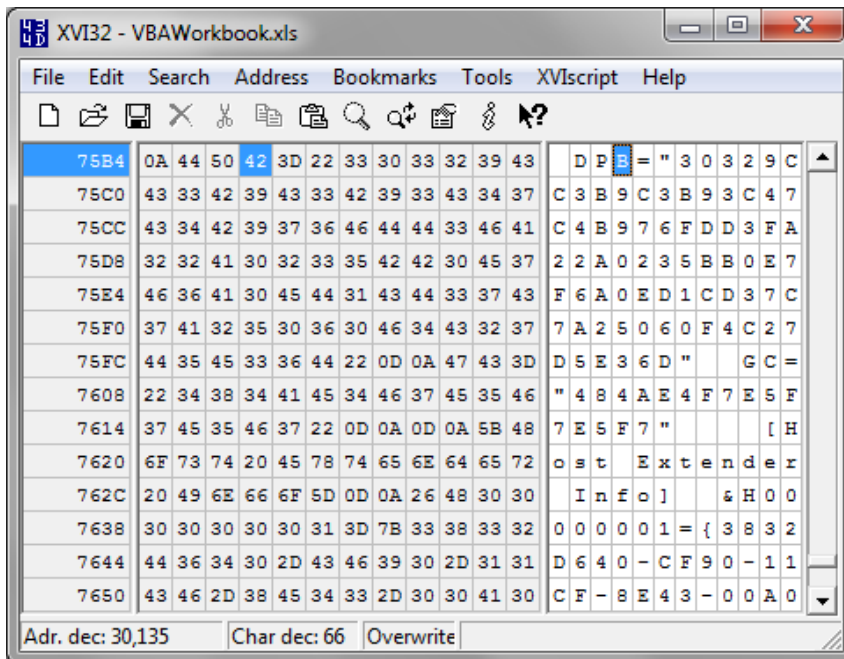## Step 4: Find and Replace the DPB Keyword

In your Hex Editor, look for the text string DPB.

Sites I Visit

BeExcellence

Chandoo Excel Tips

Chris Web BI Blog

Clearly and Simply

Contextures

Daily Dose of Excel

Excel Campus

Excel TV

ExcelGuru

PowerPivot Pro

PTS Blog

SpreadsheetGuru

Your Sum Buddy

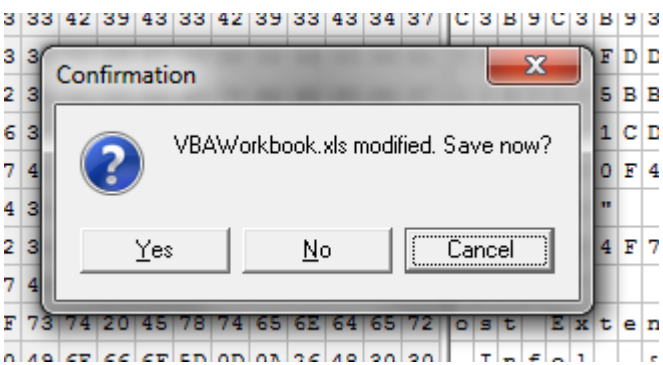This text string apparently indicates Excel's tag for the VBA Password.

In order to confuse Excel and get around the VBA Password prompt, you just need to muddle the tag a little. You can do this by simply changing the DPB to something like DPX (replace the B with an X).
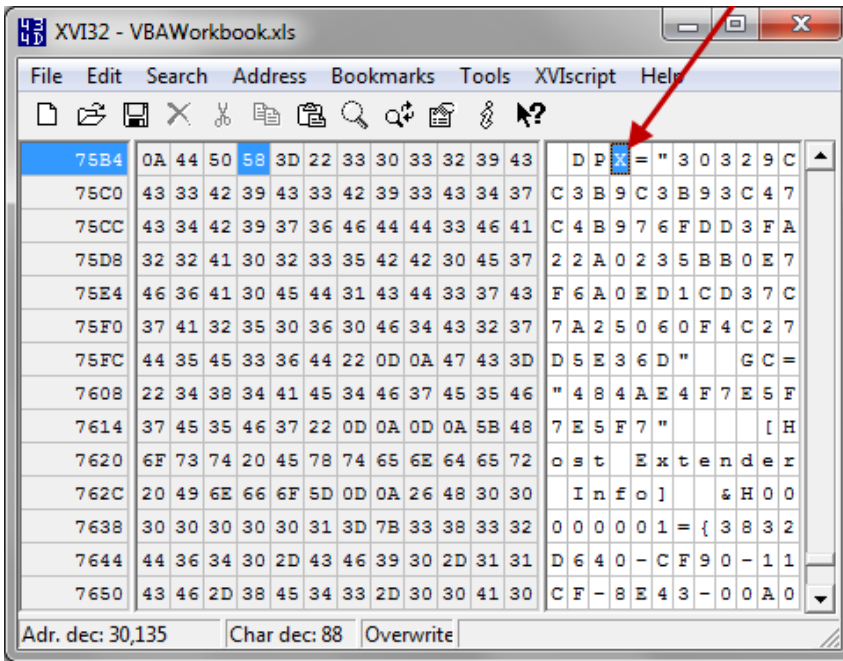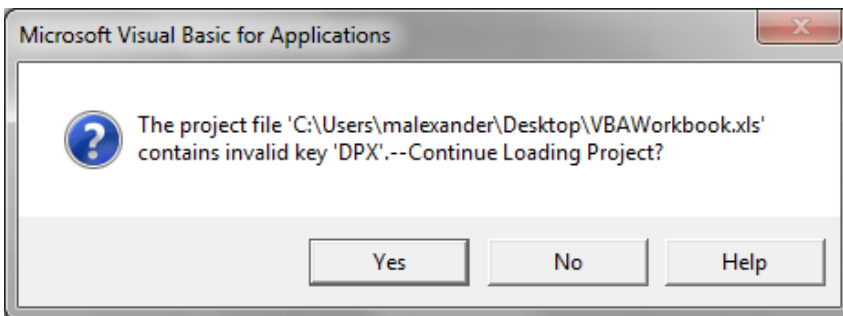
Step 5: Save Your Changes

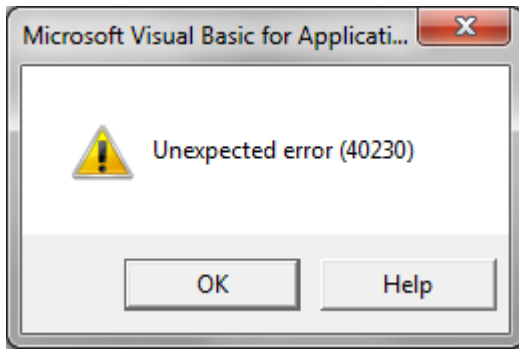Close the Hex Editor and be sure to save your changes

Step 6: Open your Newly Diddled Workbook

Open the workbook you just finished molesting. You will get a few messages.

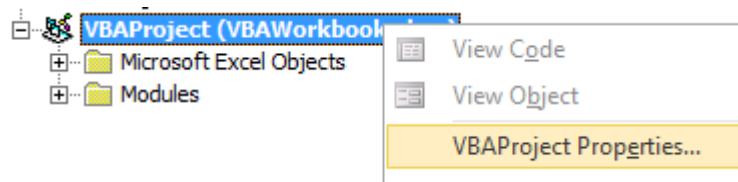First this one, which you will say Yes to:

Then you'll see this one (a few times). Don't panic, and keep clicking OK.
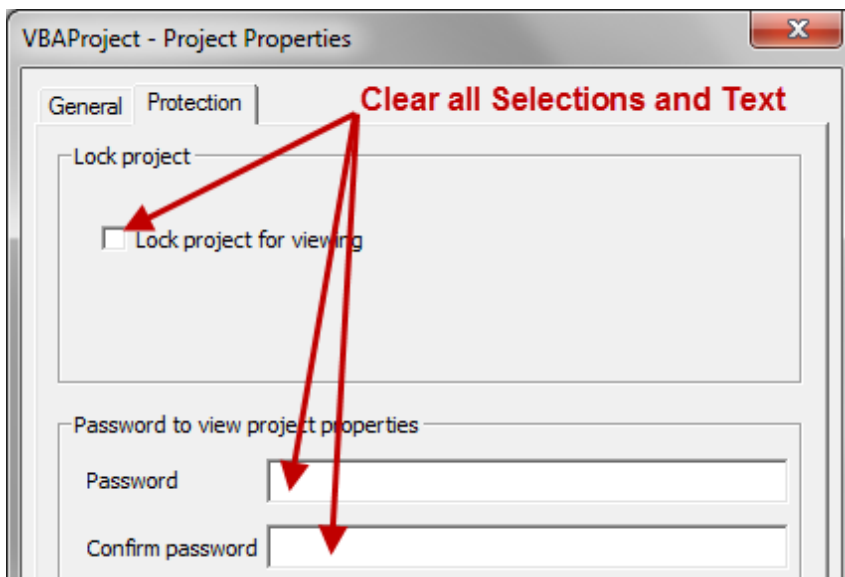

Step 7: Remove the VBA Protection

When Excel stops its bitching and moaning, go to the Visual Basic Editor (Alt+F11) and then select View>>Project Explorer.

Right click on the VBAProject for your workbook and select VBAProject Properties.

In the Properties dialog box, go to the Protection tab and clear any selections and text you see there.

Step 8: Save a Close your Newly Unprotected Workbook.

Save and close the workbook. At this point, you'll have full reign over the VBA in the workbook.

There you have it – another trick for your toolkit.

I could suggest that you use this trick only for good and righteous purposes, but you won't listen anyway.

🏷 <u>Excel</u>, <u>Tips and Tricks</u>, <u>VBA</u>, <u>Worksheet Protection</u>

<div style="border:1px solid #ccc; padding:10px;">← 5 Rarely Used VBA Tricks</div>

<div style="border:1px solid #ccc; padding:10px;">Add a DropBox Account to your Office 2013 Save As Screen →</div>

107 thoughts on "Hack into Password Protected VBA Projects"

1. **Dale M**
   April 3, 2017 at 3:57 pm

   Worked well, even for Access 2010.
   The only thing was, in order to fully remove the password, I had to apply another password, save it, and then remove it again.
   Thanks dude.

2. **Chas**
   April 10, 2017 at 1:44 pm

   Downloaded the hex editor. Loaded the non xlsm file. The editor was unable to find the string?

3. **Tim**
   April 10, 2017 at 3:29 pm

   Followed your instructions and when I did a search on DPB it returned no results. Any suggestions of what to look for?

4. **Stephan**
   April 10, 2017 at 5:18 pm

   I tried it on a Microsoft Acces MDE database, but it doesn't work for me 🙁 any other suggestions ??

5. **Anonymous**

   April 16, 2017 at 4:04 pm

   This was really helpful – I needed to open up a VBA Excel file as nobody knew what password it was saved in and this article helped me!

   Thanks!

6. **mukesh**

   July 29, 2017 at 8:34 pm

   hi,

   i have followed same step but getting error "hexa text invalid" please help how to use this tool

7. **Cyn**

   August 25, 2017 at 1:13 pm

   For an .xlsm you need to do it a slightly different way.

   1.Change the extension of the .xlsm file to .zip.

   2.Open the .zip file (with WinZip or WinRar etc) and go to the xl folder.

   3.Extract the vbaProject.bin file and open it in a Hex Editor.

   4.Search for DPB and replace with DPx and save the file.

   5.Replace the old vbaProject.bin file with this new on in the zipped file.

   6.Change the file extension back to .xlsm.

   7.Open workbook skip through the warning messages.

   8.Open up Visual Basic inside Excel.

   9.Go to Tools > VBAProject Properties > Protection Tab.

   10.Put in a new password and save the .xlsm file.

   11.Close and re open and your new password will work.

[← Older Comments]

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

post comment

☐ Confirm you are NOT a robot

Copyright © 2017 Bacon Bits. Theme by Colorlib Powered by WordPress   Default footer text