SIMPLIFIED PACKAGING FOR MSIX

TMEditX

TMurgent Technologies, LLP

TMurgent Technologies

Using TMEditX

INTRODUCTION

TMEditX is the premium MSIX Package Editor tool that allows for the best compatibility results when packaging standard applications for delivery as MSIX packages.

Using TMEditX assumes prior familiarity with using the Microsoft MSIX Packaging Tool (MMPT). It completely replaces the use of our PsfTooling utility, as well as any manual use of the PSF that might have been done previously.

The packager will first use the MMPT to capture a customized installation of the application into a base MSIX package. This capture would not use PSF Tooling or make other adjustments to the packaging. It is also not necessary to have the MMPT sign the package, although it is OK if it does.

INSTALLATION, REGISTRATION, CONFIGURATION

Although TMEditX may be installed on the packaging VM, it is typically installed on the Test VM. TMEditX is an MSIX package.

After installation, start TMEditX from the start menu. The tool will go to the File→Options panel. Click the Registration button at the top, and when prompted paste in the license key. This step requires an active internet connection to complete product registration.

Set other configurations on this options page, including setting up the code signing that will be used. The options shown in the following image are commonly used.

The default signing uses a code signing certificate file (PFX). If this will be used, select the Signtool method, browse to the PSF file, and enter the password and a url to a timestamping service as shown in the following image.
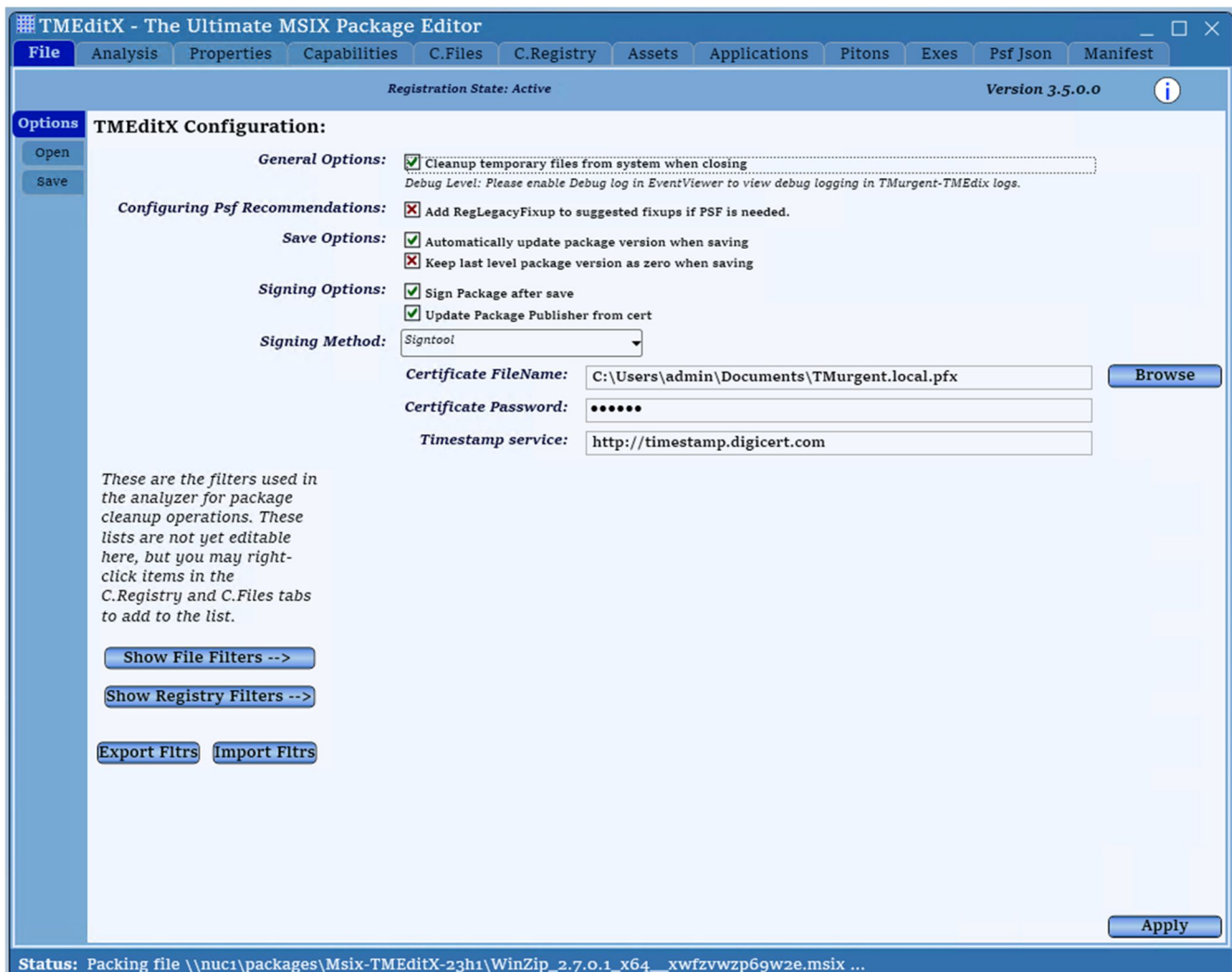
If using the Digicert KeyLocker code signing service, you map also install and configure the Digicert Tools, select the Digicert method and add appropriate configuration for that style of signing instead.

When you click on the Apply button on the bottom right, the registration, and all configuration will be securely stored.

OPENING AN MSIX PACKAGE FOR EDIT

After starting TMEdit, do the following
1. Along the top tabs, select the **File** tab.
2. On the left side, click the **Open** sub-tab.
3. On the right side of the display, click the **Open** button.
4. In the file dialog, browse to the MSIX file you want to edit and select the file.

TMEditX will open the package, unpack the contents into a temporary folder, analyze the package, and display the analyzed.
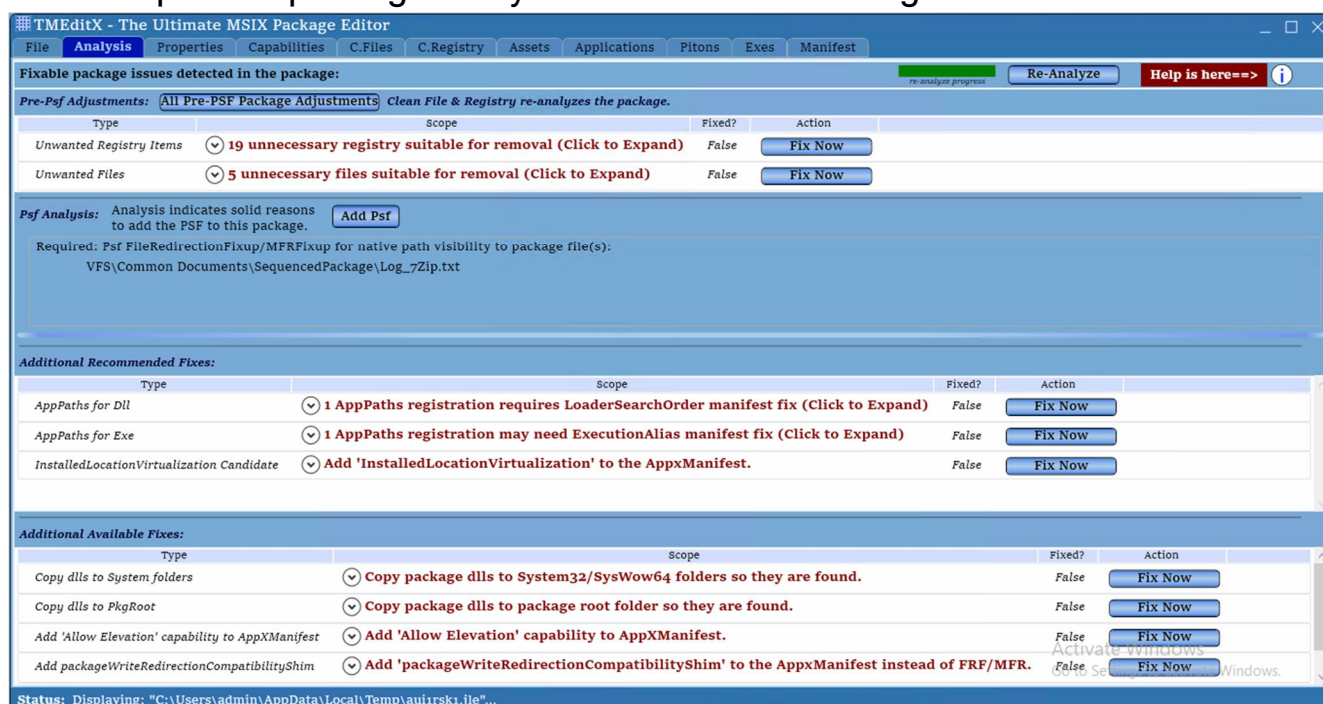
You will see the progress on this display.
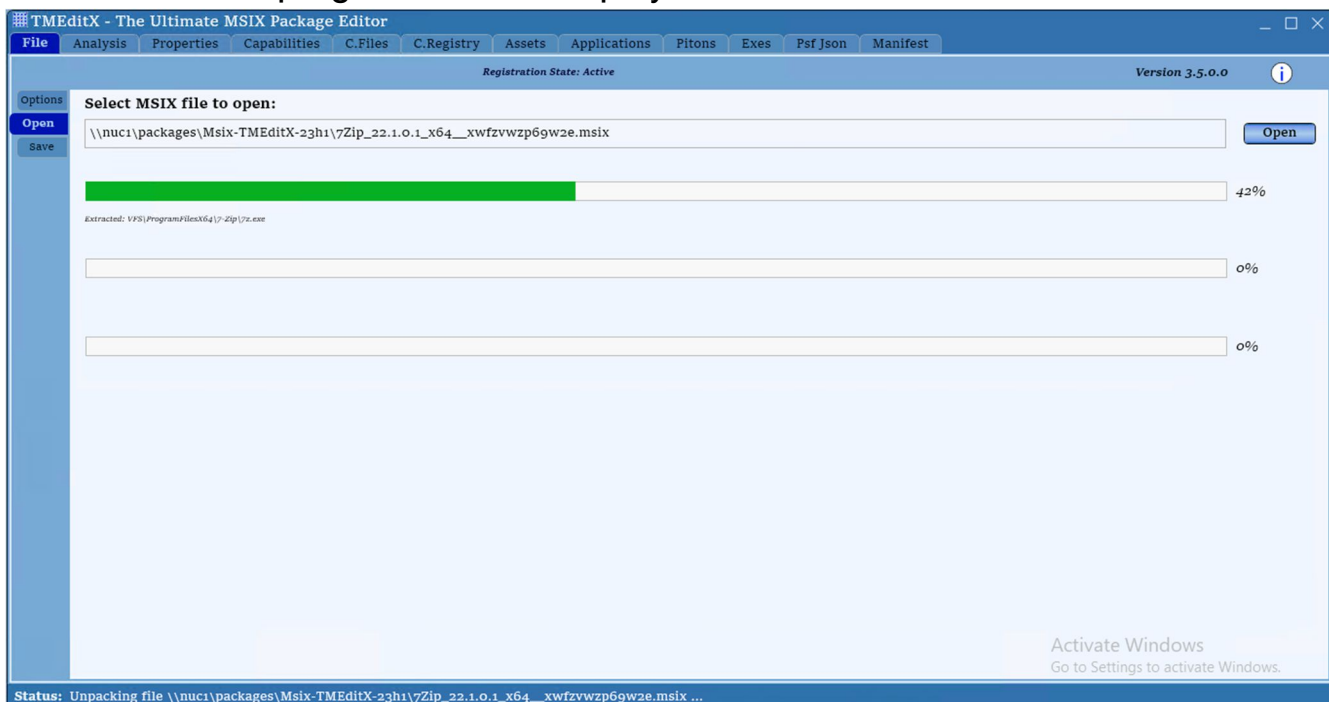
When the analysis is complete, it will be displayed on the **Analysis** Tab as shown in this example.

PROCESSING THE ANALYSIS

An example of a package analysis is shown in the image below.

For most applications, do the following.

5. Click the **Pre-PSF Package Modifications** button. This button will perform up to three actions for you:
    - Additional package cleanup (removal of unnecessary file and registry items),
    - Remove any Squirrel updaters the application uses (popular in Electron based apps).
    - Fix issues with captured Portable Apps.

    After performing these actions, the analysis will be rerun..

6. Read the Psf Analysis results.
    - If the PSF is recommended, click the **Add PSF** Button. In the dialog that appears, click the **Apply** button on the bottom.
    - If the PSF is not recommended, the button will be labeled **Add PSF Anyway**, in that case move to step 7.

7. Look at the middle section labeled *Additional Recommended Fixes.* Click any buttons labeled **Fix Now** that are in this section, working from the top down. Do not click buttons in the *Additional Available Fixes* section.

SAVING THE MODIFIED PACKAGE

8. Click on the **File Tab** at the top, and then the **Save** sub-tab on the left.
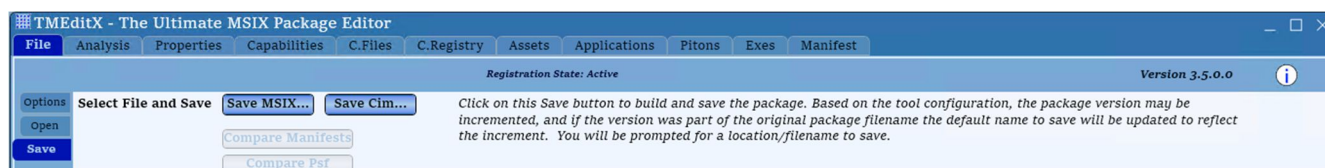
9. Click on the **Save MSIX** button, select a place to save the edited package. You typically do not want to overwrite the package you started with, so we recommend saving into a different folder.

If requested in the tool configuration, the version number will be updated and the Subject field extracted from the certificate and placed in the Publisher field of the AppXManifest. The package will be saved and (if configured) signed. When completed, you will see text showing some of the output. If both the MakeMSIX and Signtool phases complete with return codes of 0, your package is ready for testing and the text will be displayed in a green color. Otherwise it will be in red and you can read the results. A successful example is shown below:
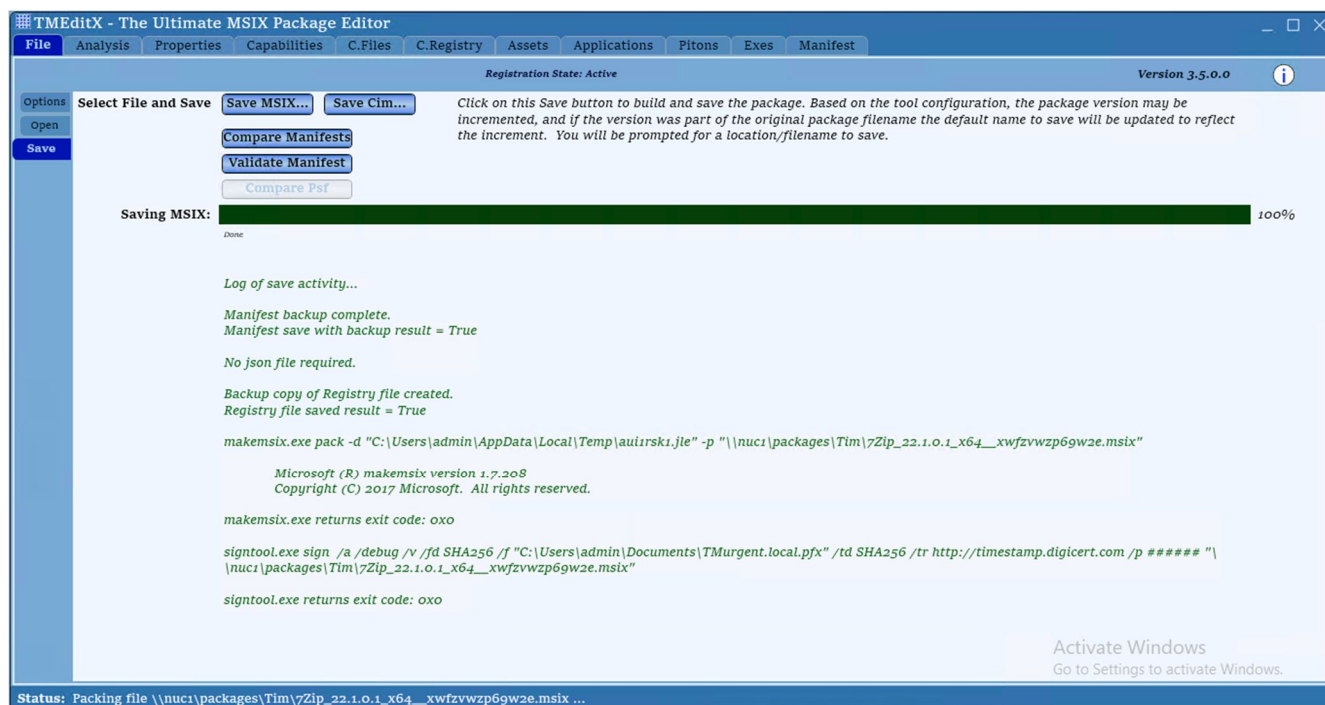
ADVANCED USAGE OF TMEDITX

Although most applications may be handled using the instructions described so far, TMEditX has more capabilities that an advanced package could use.

The analysis provided by TMEditX looks at the intent of the original application installation, and the package as generated by the MMPT. It understands all of the recent additions to MSIX not yet handled by the MMPT, as well as quite a few situations where the MMPT is less than perfect and we can improve. In addition to the PSF analysis also done by PsfTooling, this tool contains nearly 50 additional types of fixes it can detect and perform.

But no amount of analysis can make every package perfect. By diagnosing your package, possibly using tools like Process Monitor, or by requesting the Debug version of the Psf (on the Psf Editor popup window) with DebugView, you can learn to diagnose certain conditions and retry editing the package, possibly selecting different Psf Components or configuration, or possibly in using some of the *Additional Available Fixes.*

For example:
- Package Properties may be modified on the **Properties** tab, In some packages it might be necessary to increase the Min Version field.
- Additional Capabilities may be declared on the **Capabilities** tab. For example, if the primary app launches a child process that requires elevation, this needs to be declared.
- File and Registry editing is available on the **C.Files** and **C.Registry** tabs.
- On the **Assets** tab, you can replace graphics generated for shortcuts and the app package with either higher quality graphics. Sometimes the MMPT grabs the first shortcut it finds for the package, which might not be the right one.
- The **Applications** and **Pitons** tab show all of the detail about what the package requested for shortcuts and other entry-points to the package like file type associations, and the current status if supported by the Manifest.
- The **Exes** tab shows all of the exe files in the package, what type they are and if there is a manifest for things like elevation that might cause a UAC prompt.

- The **Json** tab appears if the PSF has been added to the package, This shows the *config.json* file that configures the PSF in your package and may be edited.
- The **Manifest** tab shows the AppXManifest file that will be saved. Because the remainder of the tool changes the manifest, this interface does not allow for manual editing of the AppXManifest. If needed, you can always save the package in TMEditX and then re-open for edit in the MMPT to manually edit the AppXManifest.

The GUI also has internal documentation that is available from the "i" button on the top right of the GUI. There is even a section on using TMEditX from the command line to automatically fixup the package without the GUI.

MSIXDEPLOY

TMeditX package also publishes a secondary app called MsixDeploy. This is a test tool that can simplify your initial testing process.

After launching, configure the directory where you store the packages and click the apply button. Then go to the **MSIX** tab to see a list of all packages in that share folder. Right click on a package for a menu of options. In the image below you can see how to install the package.

Once the package is installed, click the radio button "**Installed Packages**" just below the tabs so that you can work with any installed packages. Right click on an installed package for quick access to package files and debugging steps.

Here is an example:
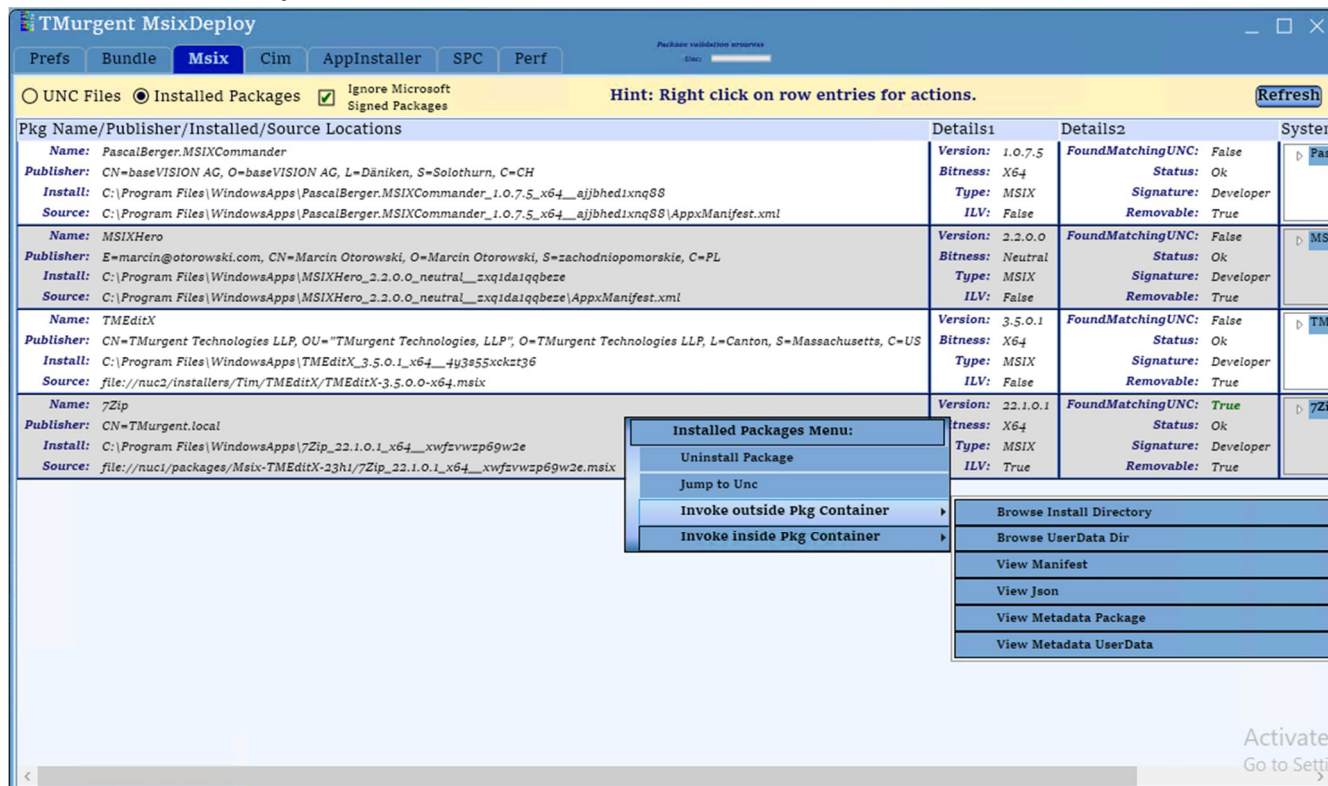
To achieve best compatibility, attending one of our training classes where we teach not just technique, but application and debugging fundamentals is essential.

PURCHASING AND LICENSING

TMEditX is available for purchase from TMurgent Technologies directly from our website (listed at the bottom of the package). The license is on a per-named user basis, and includes one year of support and new versions. There is a separate yearly renewal to keep getting support and new versions.

Discounts are available for bulk purchases. You can buy online or request a quote to pay by purchase order.

The software may be downloaded for trial use. It is fully functional, but will not fix your package sometimes to encourage that you purchase the product.

REPORT CARD ON MSIX

How good is TMEditX? Each year we issue a report card on MSIX that includes some extensive application compatibility testing. Although we are always improving the software, results from January say that:

* With the MMPT alone, we see about 30% compatibility in typical applications that large enterprises are deploying.

* With MMPT and PsfTooling we see around 60%.

* With MMPT and TMEditX, around 75%.

You can check out our latest report card from our website.