

Setting Up Windows software on Linux with WineTools

BOTTLE OPENER

No matter how simple the underlying principle of Wine might seem, the details are often tricky. Many settings must be configured individually for every program, and checking these settings can be difficult and time consuming. The mere attempt to install a Windows program may involve hours of searching on the Internet, as well as a fair smattering of trial and error, until you finally have the application running. And there is no guarantee you will be successful. Only a third of all Windows programs can be talked into running on Linux.

I took over Frank Hendriksen's orphaned WineTools project over a year ago, and since then I have worked on extending WineTools to provide an installer for Windows programs in Linux. So far, I have mainly concentrated on the big gun Office and graphics applications, that is, Microsoft Office and the Office viewers, Internet Explorer, Adobe Photoshop, and Illustrator. A large number of shareware, freeware, and Open Source programs have been added in the meantime. If alternatives on Linux exist, these alternatives are described along with their Windows equivalents.

Fast Install

Download WineTools from [1] and store it somewhere appropriate for your distribution. RPM and tar.gz versions are available, some distributions have started adding the package to their

repositories, Debian for example, and there are rumors of a Gentoo package.

The current WineTools version is 2.1.2. The RPM package is easy to install. After completing the download, become root and enter `rpm -ivh Winetools...` to install the software. If RPM complains that the `gettext` package is missing, install the package from the distribution CDs to ensure that international locales are supported. WineTools will default to English if `Gettext` or `gettext.sh` is missing. Installing the `tar.gz` file is just as easy. After unpacking the tool in the right directory by typing:

```
tar -xpvzf Winetools...
```

run `install.sh` with root privileges. This copies the files to `/usr/local/winetools` and also creates links in `/usr/local/bin`,

allowing you to launch WineTools by giving the `wt2` command.

When you launch WineTools, it is very important that you work in X and are not the root user, as the program will not launch otherwise. If you have not yet created a user directory for storing your Wine files, WineTools will let you know and display a short HOWTO on creating the directory. This is followed by a dialog with a short copyright note and an introductory text (see Figure 1.)

Make sure you read the intro; it not only gives you legal information concerning the licenses you need if you want to use the installable software, it also contains a call for support by the project, and some details of the Wine version used.

The WineTools main menu has four sections for preparing and installing pro-



original photo: www.sxc.hu

WineTools is an installation program that helps users set up Windows software on Linux without wasting time with experiments.

BY JOACHIM VON THADDEN



Figure 1: Read the introduction while you are installing; it contains licensing information and details of the Wine version you are using.

grams, controlling the Wine environment, backing up Wine installations, and finally, for displaying the introduction and copyright information.

Basic Installation with Internet Explorer 6

Our first exercise is to set up a basic system on which all other program installations will build. We need to work through the *Base set-up* menu completely. Of course you only need to select one version of Internet Explorer in your own language. The *Create a fake Windows drive* entry creates a `~/wine` directory, deleting a directory with the same name if it exists. You might like to back up any important files you have in the directory before you proceed. During the *Base setup*, WineTools asks you for the path to your CD ROM drive, your user name, and your organization name in order to prepare the simulated Windows version as well as possible.

After completing these preparatory steps, you can start with the installation. To improve readability, you will want to install the Arial font. This font is followed by the installation of DCOM98, which gives Wine the required OLE functionality. Although DCOM is implemented by the Wine libraries, this implementation is incomplete and experimental. Some programs do not support it and will refuse to cooperate at the install stage. This is why WineTools uses the original Microsoft version. Using the

original DCOM has one disadvantage however. Microsoft only allows use of DCOM in combination with a Microsoft Windows 98 or newer operating system. So make sure you have Win 98 or a later Windows in your software cupboard.

Windows License Required

Assuming you accept the license conditions when you run the menu item, the installation should then complete without any trouble. The Wine debug messages on the command line refer to functions that have not been implemented and can be ignored. Attentive readers may have noted that WineTools pops up windows to let you know what is going on. For example, WineTools simulates a Windows boot after completing each installation and ensures that all your Wine programs have been shut down.

After installing the MS Foundation Classes, let's move on to the last and most important step: the basic Internet Explorer 6 Service Pack 1 installation. There are two things that make this installation interesting. First, Internet Explorer or parts of Internet Explorer are required components for many programs, and the programs will not work properly without them. For example, more or less all Windows programs use the IE6 libraries to display HTML content. Another reason for the importance of IE is that this program gives you a complete system update. The additional libraries can be very important for other programs. Double clicking the right language version launches into a completely automatic install, including Internet download. Be patient; it might take a while for the setup tool to find the right initialization files on the Internet and complete the download.

WineTools will store the files it downloads in `~/winetools/sys` to avoid having to repeat the marathon session if it needs to reinstall the files. This gives you the option of using WineTools on a computer without an Internet connection: first, install the software on a computer with an Internet connection, and then burn the whole set of downloads located below `~/winetools` onto a CD.

Printing with CUPS, and the Old-Fashioned Way

Wine is perfectly set up for printing with CUPS. Any printers configured in CUPS

will automatically be available to any Wine applications. But if you still use the legacy printcap system, you will need the last menu item to configure Wine to use your printing environment. Wine has a *generic.ppd* that defines a generic printer. The path to this file can differ depending on your distribution; so you may need to modify the path entry. *locate generic.ppd* will point you in the direction of the file.

Before you start trying out a few initial applications, let's look at another WineTools function. There is a *Show Info* checkbox down at the bottom of each Winetools window (Figure 2). If you enable the checkbox, selecting a menu item will not only take you to the program installation but also display a website in your browser. Testing Internet Explorer took us to the Microsoft IE6 installation page, which reports that Internet Explorer has "Advanced Security Technologies."

Wine Security?

You may be wondering how vulnerable Windows programs such as IE6 make your Linux installation. Faced with a never-ending series of security warnings, it does seem a bit strange to be using Internet Explorer. So how vulnerable does using Windows software with Wine make a Linux host?

In theory, a security hole in IE6 could be used to attack a system with Wine, and there is no way of predicting whether the injected code will actually run on Wine. The odds are the same as for any other Windows software. There is a moderately humorous article at [2] in which the author describes his fruit-



Figure 2: The Show Info checkbox displays a website with additional Help support.

less attempts to test five viruses for Wine compatibility.

Which ever way look at it, vulnerability is not a major concern for Linux machines; it normally takes *root* privileges to modify a program. Local user data may be subject to some risk. Changing a binary file that uses the Linux ELF format will simply destroy the file.

Additionally, Windows viruses typically use file extensions to locate interesting targets, and these extensions are completely untypical of Linux binaries. The fact is that many system libraries that malevolent programs could use either do not exist in Wine, or have been replaced by Linux Library versions that are only writable by *root*. Windows files installed under Wine are a different matter; in fact, they may be just as vulnerable under Linux with Wine as they are on Windows.

There are two approaches to mitigating the risk and preventing damage to local Linux files. The easiest way is by avoiding configuring drive mappings for Linux directories in Wine, and avoiding similar links on the simulated C drive, which is like putting Wine in a sandbox. This means that files have to be located below the directory tree accessible to Wine.

What this means in terms of administration is removing the *h:* and *z:* links below *~/wine/dosdevices*, as both of these links point to the Linux filesystem tree. *H:* maps to the logged on user's home directory and *Z:* points to the filesystem root. The other links that WineTools creates, *c:*, *d:*, *e:*, *f:*, and *t:* restrict Wine to its own subdirectory, *~/wine/drive_c*, the CD ROM, the temporary files below */tmp* and the WineTools cache directory. This step plugs the biggest security gaps. Of course, it does mean that Wine can't retrieve files from other locations, so you may need to copy any required files to */tmp* or *~/wine/drive_c* as a work-around.

The other approach to security with Wine is to create a special user account for Wine and run security-critical software such as IE6 in that user's Wine instance. By default, WineTools does not use either of the security options I just described; this design provides for maximum flexibility and allows users free-

dom of choice. If you need more security, I would recommend using the second approach.

IE6 and Media Player on Linux

You can launch the newly-installed Internet Explorer by entering *ie6* at the command line. WineTools places the script in *~/bin*, so make sure your path has an entry for *~/bin*. If not, add the following entry to the *.bash_profile* or *.bashrc* file:

```
export PATH=$HOME/bin:$PATH
```

Don't forget to log off and back on again to apply the changes.

IE6 should now launch as expected and take you to the MSN welcome page. Thank goodness you can configure IE6's start behavior in *Tools/Internet options*. There is no need to set up the Internet connection using a wizard (like on Windows) Wine handles Internet access transparency via the underlying Linux operating system.

You might note a few usability issues with IE6. For example, you can use existing bookmarks, but you can't create new bookmarks, as IE6 complains about invalid characters.

These problems with bookmarks are hardly likely to faze a typical Linux user whose use of IE6 will tend to be fairly sporadic. Another thing you might notice is that IE6 crashes more often than on its native platform. If the program freezes, just wait for it to come back. It shouldn't take too long to wake up again. But accessing the local filesystem almost

always causes a crash.

Don't bother installing Java. The only Java environment that will run is the old Microsoft Java. Applications that embed Internet Explorer, such as the Ebay browser Lauge, stand a good chance of working, but browser extensions under-achieved in all of our tests.

No Chance for Java

Despite the drawbacks, Internet Explorer is still a useful tool for accessing sites that use Javascript without support for other browsers, or for testing your own sites or web software, and – as previously mentioned – because more or less every program that handles HTML uses the IE6 HTML libraries.

Installing IE6 automatically installs Media Player. You can launch the player by typing *wmplayer* at the command line, but don't expect too much of it. Sound may be missing altogether or unsynchronized. Videos can also have strange colors; if so, you might like to disable hardware acceleration in the Media Player settings. Recent versions of Media Player will not be stable if they run at all.

Codec packs give users the ability to add codecs. The SLD Codec Pack turns out to be a very useful source. You can install it via the *Tools* group in the *Install tested software* menu. After the obligatory simulated reboot, which WineTools automatically performs, the codecs should be enabled, which will allow you to view MPEG 2 files. Avoid installing the Media Player version provided with the codec pack – it crashes when launched. Other codec packs turned out to be very unstable in our tests.

After returning to the WineTools main menu, it is a good idea to install frequently used system software. *Windows System Software* takes you to the current Windows installer, the Visual Basic and C++ Runtimes, the ODBC database engine and drivers, the current XML libraries, and Windows Script. Just go through the list and install every single one of these products, making sure that you choose the language version appropriate to your locale. When installing the Windows installer, WineTools will warn you that the installer may not be able to handle older programs. But this is something that does not happen a lot in live operations. If you are prompted to



Figure 3: You should at least have Arial as a TrueType font, and having more typical Windows fonts won't hurt.

reboot after completing an installation, it really doesn't make any difference whether you say yes or no, as WineTools takes care of this reboot for you.

The font installation is very similar to the system software installation (Figure 3); to launch into the install, select *Install Microsoft TrueType core fonts*, and go through the list until you have all the fonts. These fonts are only available to Wine; the Linux host will not see them.

The "Install Tested Software" item gives you a hierarchically organized menu with more than 100 categorized entries for software that has been tested with WineTools, including some of the big commercial packages such as Adobe Illustrator or Photoshop, and shareware or freeware tools such as VirtualDub or The Bat.

Versions 97 and 2000 of Microsoft Office perform well on Wine. Winword, Excel, and Powerpoint are all fast and reliable. The other applications are usable with some restrictions. Access tends to freeze. If you have trouble getting Access to work with Wine, Cross-

over Office is a useful alternative.

Office 2000 as the Killer Application

WineTools manages software dependencies and issues warnings. For example, when you start to install Office 2000, WineTools tells you that the program requires *SplashKiller.zip*, and that the program is not installed. You can install *SplashKiller.zip* later, so just click OK to confirm. In the next dialog, WineTools displays some notes on the particulars of the current installation. Some Office Installers ask you for the drive with the install files; this will be the CD drive, which WineTools has mapped to D:. You are also warned not to install the Office assistants (the paper clip and so on), because they don't work and can make Office unusable. The Symantec fax service will not work either.

Hands Off that Paper Clip!

For some programs, WineTools gives the option of manually selecting a different location or language version of the

installer. You can keep the defaults in the case of the Office CD. WineTools will search the CD autonomously for an Autorun entry. You will not need to help WineTools unless it fails to find the Autorun entry. The next step is to mount the CD and launch the install.

After entering the CD key and selecting the CD ROM drive (D:), select a user-defined installation. You can install any components you like, except for the Office assistants mentioned previously (located below Office Tools). The installation should look just like on any Windows system. At the end of the process, the installer asks if you would like to reboot, but no matter what you decide to do, WineTools will simulate a reboot anyway and tell you the names of the scripts that it has stored below `~/bin`. Finally, WineTools asks you if you would like to unmount the CD. If you intend to use an Exchange server with Outlook, answer no to this question.

You can now launch the installed programs at the command line. The start scripts are titled *msword*, *msexcel*, *msac-*

Want to know what's up next?

Subscribe to Linux Magazine Preview,
our free monthly email newsletter!



WWW.LINUX-MAGAZINE.COM/NEWSLETTER



Figure 4: WineTools notes exactly which software you have installed, even though Windows may not have noticed.

cess, *mspowerpoint* and *msoutlook*, and the tools are *msfinder*, *msquery*, and *msbinder*.

Working with Outlook and Access

Outlook can be used with POP3 and IMAP mailboxes or an Exchange server, and the other PIM functions such as contacts and appointments should all work fine. If you are interested in using your Outlook data on Linux, you might like to give this a try, but there are a few things you need to take care of before you do. Outlook display an annoying splash screen when launched, the screen covers up the dialogs and makes it more or less impossible to complete the configuration. Before you launch Outlook for the first time, you will need to install the Splash Screen Killer from below Tools. But WineTools tells you all about this when you start installing Office. Having installed the killer, launch Outlook for the first time by typing *msoutlook*. A word of caution at this point: make sure you choose the *Internet only* option when setting up a mail account; if you do not, any subsequent configuration

steps will fail and this will leave you with an unusable Outlook version. After you have filled out the fields in the dialogs, and canceled the Symantec Fax installer, Outlook should launch. Try to quit Outlook immediately. The client typically hangs leaving you with no alternative but:

```
wineserver -kill
```

that is, to kill any Wine instances that happen to be running right now. The next time you enter *msoutlook*, Outlook should be completely usable. There is one minor annoyance: the icons in the left Outlook bar are completely black, but you can select *View | Outlook toolbar* to disable the icons and replace them with a folder list.

If you will be using Outlook with an Exchange server, you will need to

restart Outlook and change your mail account type in *Tools | Options | Email accounts*. Click on *Reconfigure email account* to do so. Then relaunch Outlook. Outlook will access the CD and install the missing libraries; you can then follow standard procedures to configure the account.

Microsoft Access works, sometimes. How far you get will depend on the application and the database drivers you use. While some applications run without any complaints, others will continually crash.

You might like to integrate your Windows applications with the desktop, to launch Word automatically from the browser, the mail client, or the desktop, for example. Although the commercial Crossover Office gives you all of these features automatically, some manual steps are involved for Wine users.

Integration with the Linux Desktop

You can tweak browser settings to launch the scripts stored below *~/bin*. Both KDE and Gnome support easy handling of the file types. Many pro-

grams also read the mail capabilities files, so you could add the following entry for Winword to your *~/mailcap* file:

```
application/msword; /home/thadden/bin/msword %s
```

Don't worry about Cut and Paste functionality; if you press [Ctrl] + [C] in WinWord to copy a section of text, you can paste the text into Open Office without losing the formatting information by pressing [Ctrl] + [V]. Wine is capable of exchanging rich text format texts via the clipboard.

WineTools Capabilities

There are still a few entries in the WineTools main menu that we haven't touched on thus far: the Control Center entry takes you to Wine's equivalent to the Windows Control Center. When you get there, you will find all kinds of configuration applets for ODBC management or IE6 Internet options. There is a Registry Editor, and *Show Installed Software* gives you a complete log of every program installed via WineTools; the *Uninstaller* entry lets you deinstall these programs.

Another useful feature is the ability to back up your complete Wine environment for archiving on CD, or to have a safe copy before you experiment with your settings. The entry for editing the Wine configuration is for gurus only. It can have a critical impact on the installed Windows software, so make sure you know exactly what you are doing, and create a backup before you start.

Community help is needed to continue the development of WineTools. If you have managed to talk another Windows program into running on Wine, please mail me, and let me know, so that I can add the program to the next version of WineTools. I am also planning to set up a website to make it easier for people to contribute. ■

INFO

- [1] WineTools homepage: <http://www.von-thadden.de/Joachim/WineTools>
- [2] Windows viruses on Wine: <http://os.newsforge.com/article.pl?sid=05/01/25/1430222&from=rss>