



# how to make the wifi\_stress.svg file apply to the binary file WiFiTestTool in the ubuntu system

Create a desktop launcher for WiFiTestTool and point its icon to wifi\_stress.svg. This makes the SVG show up in menus, dock, and launcher instead of the default binary icon. [\[1\]](#) [\[2\]](#)

## 1. Put files in suitable locations

Assume:

- Binary path: /opt/wifi/WiFiTestTool
- Icon path: /opt/wifi/icons/wifi\_stress.svg

Steps:

1. Create folders (if needed):

```
sudo mkdir -p /opt/wifi/icons
```

2. Copy files:

```
sudo cp WiFiTestTool /opt/wifi/  
sudo cp wifi_stress.svg /opt/wifi/icons/  
sudo chmod +x /opt/wifi/WiFiTestTool
```

## 2. Create a .desktop file

Create a launcher file so the desktop environment knows your app, its name, and its icon. [\[2\]](#) [\[1\]](#)

1. Create file:

```
sudo nano /usr/share/applications/wifitesttool.desktop
```

2. Put this content (adjust paths if different):

```
[Desktop Entry]  
Type=Application  
Name=WiFi Stress Tool  
Comment=WiFi stress test utility  
Exec=/opt/wifi/WiFiTestTool  
Icon=/opt/wifi/icons/wifi_stress.svg  
Terminal=false  
Categories=Network;Utility;
```

3. Save, then update desktop database (optional but recommended):

```
sudo update-desktop-database
```

### 3. Make the launcher visible and trusted

Depending on desktop (GNOME, etc.), you may need to:

- Log out/in or run:

```
gio set /usr/share/applications/wifitesttool.desktop "metadata::trusted" yes
```

- Or copy to user applications:

```
mkdir -p ~/.local/share/applications  
cp /usr/share/applications/wifitesttool.desktop ~/.local/share/applications/
```

You should now see “WiFi Stress Tool” with the **wifi\_stress.svg** icon in:

- Application menu / launcher.
- Dock, once running.
- Desktop shortcut if you drag it out from the menu. [\[3\]](#) [\[4\]](#) [\[1\]](#)

### 4. Quick test

- Run from menu: search “WiFi Stress Tool”.
- Or run:

```
gtk-launch wifitesttool
```

- When it’s running, confirm the dock/taskbar shows your SVG icon. [\[1\]](#) [\[2\]](#)

If `WiFiTestTool` is a terminal-only tool and you want a terminal to open, change `Terminal=true` in the `.desktop` file.

✱

## how to use pyinstaller to generate the binary file and make the wifi\_stress.svg as the icon of this bin file

Use PyInstaller’s `--icon` option, but first convert `wifi_stress.svg` into a format PyInstaller and the OS actually use (PNG on Linux, ICO on Windows). [\[21\]](#) [\[22\]](#) [\[23\]](#)

## 1. Convert SVG to a suitable format

On **Linux**, PyInstaller can accept a PNG and the desktop environment will use that as the app icon.<sup>[23]</sup>

Example conversion with ImageMagick:

```
sudo apt install imagemagick
convert -background none -resize 256x256 wifi_stress.svg wifi_stress.png
```

On **Windows**, convert to ICO (if you ever target it):<sup>[24] [21]</sup>

```
convert -background none -resize 256x256 wifi_stress.svg wifi_stress.ico
```

## 2. Build the binary with PyInstaller

From the folder containing your main script, run for **Linux**:

```
pyinstaller --onefile --icon=wifi_stress.png main.py
```

For **Windows**:

```
pyinstaller --onefile --windowed --icon=wifi_stress.ico main.py
```

- `--onefile` produces a single binary (your “WiFiTestTool” equivalent).<sup>[25] [24]</sup>
- `--icon` embeds the icon resource into the executable on supported platforms or into the bundled app structure.<sup>[22] [21]</sup>

The generated binary will be in the `dist/` directory.

## 3. Use the icon in the desktop launcher (Linux)

Even though the binary has an icon, desktop environments typically rely on `.desktop` files and icon paths.<sup>[26]</sup>

Create or update your `.desktop` file to point at the PNG you generated:

```
[Desktop Entry]
Type=Application
Name=WiFi Stress Tool
Exec=/path/to/dist/main
Icon=/path/to/wifi_stress.png
Terminal=false
Categories=Network;Utility;
```

This ensures the **wifi\_stress** icon appears in the launcher, dock, and menus when your PyInstaller binary is used.<sup>[27] [26]</sup>

# I have a .spec

*-- mode: python ; coding: utf-8 --*

```
block_cipher = None

a = Analysis(
    ['wifi_test_newgui.py'],
    pathex=[],
    binaries=[],
    datas=[
        ('technexion_logo103770.svg', '.'), # 只打包 Logo
    ],
    hiddenimports=[
        'PyQt5.QtCore',
        'PyQt5.QtGui',
        'PyQt5.QtWidgets',
        'PyQt5.QtSvg',
        'serial',
        'serial.tools',
        'serial.tools.list_ports',
    ],
    hookspath=[],
    hooksconfig={},
    runtime_hooks=[],
    excludes=[],
    win_no_prefer_redirects=False,
    win_private_assemblies=False,
    cipher=block_cipher,
    noarchive=False,
)

pyz = PYZ(a.pure, a.zipped_data, cipher=block_cipher)

exe = EXE(
    pyz,
    a.scripts,
    a.binaries,
    a.zipfiles,
    a.datas,
    [],
    name='WiFiTestTool',
    debug=False,
    bootloader_ignore_signals=False,
    strip=False,
    upx=True,
```

```
upx_exclude=[],
runtime_tmpdir=None,
console=True,
disable_windowed_traceback=False,
argv_emulation=False,
target_arch=None,
codesign_identity=None,
entitlements_file=None,
)
```

how to modify it to auto generate the binary with icon

Add an `icon=` argument to the EXE call and (on Linux) point it to a PNG converted from `wifi_stress.svg`. The `.spec` file then always builds the binary with that icon. [\[42\]](#) [\[43\]](#) [\[44\]](#)

## 1. Prepare the icon file

PyInstaller's `icon=` expects an ICO on Windows, and typically a PNG on Linux. [\[45\]](#) [\[42\]](#)  
Convert your SVG once (Linux example):

```
sudo apt install imagemagick
convert -background none -resize 256x256 wifi_stress.svg wifi_stress.png
```

Put `wifi_stress.png` next to your `.spec` file or in a known path.

## 2. Modify your .spec file

Update your EXE definition to include `icon='wifi_stress.png'` (adjust path as needed). [\[46\]](#) [\[42\]](#)

```
# -*- mode: python ; coding: utf-8 -*-

block_cipher = None

a = Analysis(
    ['wifi_test_newgui.py'],
    pathex=[],
    binaries=[],
    datas=[
        ('technexion_logo103770.svg', '.'), # 只打包 Logo
    ],
    hiddenimports=[
        'PyQt5.QtCore',
        'PyQt5.QtGui',
        'PyQt5.QtWidgets',
        'PyQt5.QtSvg',
        'serial',
        'serial.tools',
        'serial.tools.list_ports',
    ],
    hookspath=[],
    hooksconfig={},
    runtime_hooks=,
```

```

        excludes=[],
        win_no_prefer_redirects=False,
        win_private_assemblies=False,
        cipher=block_cipher,
        noarchive=False,
    )

    pyz = PYZ(a.pure, a.zipped_data, cipher=block_cipher)

    exe = EXE(
        pyz,
        a.scripts,
        a.binaries,
        a.zipfiles,
        a.datas,
        [],
        name='WiFiTestTool',
        debug=False,
        bootloader_ignore_signals=False,
        strip=False,
        upx=True,
        upx_exclude=[],
        runtime_tmpdir=None,
        console=True,
        disable_windowed_traceback=False,
        argv_emulation=False,
        target_arch=None,
        codesign_identity=None,
        entitlements_file=None,
        icon='wifi_stress.png',      # <--- add this line
    )

```

Then rebuild using the spec:

```
pyinstaller wifi_test_newgui.spec
```

### 3. Show icon in desktop/launcher (Linux)

Even with `icon=` set, many Linux desktops use the `.desktop` file's `Icon=` field.<sup>[47] [45]</sup>

- Copy `wifi_stress.png` to a stable path (e.g. `/opt/wifi/icons/`).
- Set `Icon=/opt/wifi/icons/wifi_stress.png` in your `.desktop` file for `WiFiTestTool`.

✱✱

1. <https://stackoverflow.com/questions/2126545/embedding-an-icon-in-a-linux-executable>
2. [https://www.reddit.com/r/Ubuntu/comments/1coifz9/i\\_have\\_a\\_binary\\_file\\_cgtk\\_how\\_can\\_i\\_make\\_it/](https://www.reddit.com/r/Ubuntu/comments/1coifz9/i_have_a_binary_file_cgtk_how_can_i_make_it/)
3. <https://forums.opensuse.org/t/how-to-create-svg-icons-in-15-4/151582>
4. <https://ubuntu-mate.community/t/change-icons-of-the-alt-tab-windows-switcher/15527>
5. <https://learn.microsoft.com/vi-vn/windows-hardware/test/hlk/testref/879993b8-2581-452f-b237-47727101c4f9>

6. <https://www.netspotapp.com/wifi-scanner/linux-wifi-scanner.html>
7. <https://learn.microsoft.com/en-us/windows-hardware/test/hlk/testref/879993b8-2581-452f-b237-47727101c4f9>
8. <https://help.ubuntu.com/stable/ubuntu-help/net-wireless-troubleshooting-initial-check.html.en>
9. [https://www.reddit.com/r/msp/comments/17svn2e/wifi\\_stress\\_testing/](https://www.reddit.com/r/msp/comments/17svn2e/wifi_stress_testing/)
10. [https://www.reddit.com/r/linux4noobs/comments/1o1fei8/how\\_do\\_i\\_change\\_the\\_icon\\_for\\_a\\_link\\_that\\_points/](https://www.reddit.com/r/linux4noobs/comments/1o1fei8/how_do_i_change_the_icon_for_a_link_that_points/)
11. <https://help.ubuntu.com/stable/ubuntu-help/net-wireless-troubleshooting-hardware-check.html.en>
12. <https://cybersquad6351.netlify.app/netfloodx>
13. <https://www.geeksforgeeks.org/linux-unix/how-to-monitor-access-point-signal-strength-with-wifi-linux/>
14. <https://community.spiceworks.com/t/how-to-stress-test-a-wifi-website/802532>
15. [https://www.reddit.com/r/linuxhardware/comments/qxk1vw/how\\_would\\_i\\_test\\_my\\_wifi\\_card\\_on\\_linux\\_is\\_there\\_a/](https://www.reddit.com/r/linuxhardware/comments/qxk1vw/how_would_i_test_my_wifi_card_on_linux_is_there_a/)
16. <https://blog.csdn.net/lan120576664/article/details/103496241>
17. <https://cloud.tencent.com/developer/article/1551869>
18. [https://xajhuang.com:3100/avs/avs\\_mtk\\_voice/src/commit/5d8c4aded05a8b1fb8d9b1ef5b9892a593a1bc96/src/connectivity/bt\\_stack/bluedroid\\_turnkey](https://xajhuang.com:3100/avs/avs_mtk_voice/src/commit/5d8c4aded05a8b1fb8d9b1ef5b9892a593a1bc96/src/connectivity/bt_stack/bluedroid_turnkey)
19. <https://www.baeldung.com/linux/wifi-network-channels-test-tools>
20. <https://github.com/williamhaley/wifi-testing>
21. <https://pyinstaller.org/en/stable/usage.html>
22. <https://pyinstaller.org/en/stable/man/pyinstaller.html>
23. <https://stackoverflow.max-everyday.com/2020/12/pyinstaller打包tkinter-ui介面icon/>
24. <https://pyinstaller.org/en/v4.1/usage.html>
25. <https://www.devdungeon.com/content/pyinstaller-tutorial>
26. <https://github.com/orgs/pyinstaller/discussions/7599>
27. <https://forums.opensuse.org/t/how-to-create-svg-icons-in-15-4/151582>
28. <https://stackoverflow.com/questions/29291113/pyinstaller-setting-icons-dont-change>
29. <https://pyinstaller-icon.com>
30. <https://stackoverflow.com/questions/64603218/pyinstaller-exe-integrated-with-png-background-and-ico-as-icon>
31. [https://www.reddit.com/r/learnpython/comments/1bu1le6/pyinstaller\\_wont\\_include\\_icon/](https://www.reddit.com/r/learnpython/comments/1bu1le6/pyinstaller_wont_include_icon/)
32. <https://stackoverflow.com/questions/66742654/how-to-add-an-icon-to-pyinstaller-file>
33. <https://snuow.com/blog/pyinstallerでアイコンicon-icoを設定する方法/>
34. <https://www.youtube.com/watch?v=bV8nB5jEid4>
35. <https://stackoverflow.com/questions/79005296/making-python-exe-file-with-using-pyinstaller-icon-option-doesnt-change-file>
36. [https://www.reddit.com/r/learnpython/comments/hshrm2/pyinstaller\\_isnt\\_applying\\_the\\_app\\_icon\\_i\\_provide/](https://www.reddit.com/r/learnpython/comments/hshrm2/pyinstaller_isnt_applying_the_app_icon_i_provide/)
37. <https://www.pyinstaller-icon.com>

38. <https://ithelp.ithome.com.tw/articles/10247305>
39. <https://forum.qt.io/topic/138864/how-to-set-an-icon-for-an-exe-file-using-pyinstaller>
40. <https://stackoverflow.com/questions/51697707/setting-icon-for-pyinstaller-application>
41. <https://www.youtube.com/watch?v=HdZC8WZMPO8>
42. <https://stackoverflow.com/questions/9946760/add-image-to-spec-file-in-pyinstaller>
43. <https://thequickadvisor.com/how-do-i-change-the-icon-of-an-exe-in-pyinstaller/>
44. <https://python-forum.io/thread-7654.html>
45. <https://www.pythonguis.com/tutorials/packaging-pyqt5-applications-linux-pyinstaller/>
46. <https://pyinstaller.org/en/v5.13.1/spec-files.html>
47. <https://github.com/orgs/pyinstaller/discussions/7599>
48. <https://pythonhosted.org/PyInstaller/spec-files.html>
49. [https://www.reddit.com/r/learnpython/comments/np9dj/any\\_way\\_to\\_get\\_svg\\_images\\_like\\_a\\_chessboard\\_in/](https://www.reddit.com/r/learnpython/comments/np9dj/any_way_to_get_svg_images_like_a_chessboard_in/)
50. <https://github.com/pyinstaller/pyinstaller/issues/8784>
51. <https://www.pythonguis.com/tutorials/packaging-data-files-pyqt5-with-qresource-system/>
52. <https://darkdragon-darkdragon.blogspot.com/2020/12/python-pyinstaller.html>
53. <https://pyinstaller.org/en/stable/usage.html>
54. <https://www.youtube.com/watch?v=bV8nB5jEid4>
55. <https://groups.google.com/g/pyinstaller/c/BKyatnbBFxs>
56. <https://blog.aaronhktan.com/posts/2018/05/14/pyqt5-pyinstaller-executable>
57. [https://www.reddit.com/r/learnpython/comments/1045pp8/impossible\\_to\\_use\\_images\\_in\\_exe\\_from\\_pyinstaller/](https://www.reddit.com/r/learnpython/comments/1045pp8/impossible_to_use_images_in_exe_from_pyinstaller/)
58. <https://stackoverflow.com/questions/76261745/python-qt5-pyinstaller-does-not-set-the-icons-correctly-for-the-exe-file>
59. [https://blog.csdn.net/qq\\_45662588/article/details/118859938](https://blog.csdn.net/qq_45662588/article/details/118859938)
60. <https://pyinstaller.org/en/stable/spec-files.html>
61. <https://www.youtube.com/watch?v=HdZC8WZMPO8>
62. <https://forum.qt.io/topic/138864/how-to-set-an-icon-for-an-exe-file-using-pyinstaller>