

python / cpython Public
[Code](#)
[Issues](#)
5k+
[Pull requests](#)
2.2k
[Actions](#)
[Projects](#)
[Security and q](#)

## Commit 28463db


 encukou authored on Jun 3, 2025 · ✖ 58 / 61 · Verified
[3.12] [gh-135034](#): Remove test\_realpath\_permission ([GH-135093](#))

The test was added in [gh-110298](#), with a fix that was never backported to 3.12 and below.

It was most likely skipped in the GHA run.

[3.12](#) (#135093) · v3.12.13 ... v3.12.11

1 parent [fcf3ea0](#) commit 28463db

**1 file changed** +0 -45 lines changed
[↑ Top](#)



Lib/test

test\_ntpath.py

**1 file changed** +0 -45 lines changed


 Lib/test/test\_ntpath.py ...

```

↑... @@ -836,51 +836,6 @@ def test_realpath_cwd(self):
836 836         test_file_long,
837 837         ntpath.realpath("file.txt", **kwargs))
838 838
839 - @unittest.skipUnless(HAVE_GETFINALPATHNAME, 'need _getfinalpathname')
840 - def test_realpath_permission(self):
841 -     # Test whether python can resolve the real filename of a
842 -     # shortened file name even if it does not have permission to access it.
843 -     ABSTFN = ntpath.realpath(os_helper.TESTFN)
844 -
845 -     os_helper.unlink(ABSTFN)

```

```
846     -         os_helper.rmtree(ABSTFN)
847     -         os.mkdir(ABSTFN)
848     -         self.addCleanup(os_helper.rmtree, ABSTFN)
849     -
850     -         test_file = ntpath.join(ABSTFN, "LongFileName123.txt")
851     -         test_file_short = ntpath.join(ABSTFN, "LONGFI~1.TXT")
852     -
853     -         with open(test_file, "wb") as f:
854     -             f.write(b"content")
855     -         # Automatic generation of short names may be disabled on
856     -         # NTFS volumes for the sake of performance.
857     -         # They're not supported at all on ReFS and exFAT.
858     -         p = subprocess.run(
859     -             # Try to set the short name manually.
860     -             ['fsutil.exe', 'file', 'setShortName', test_file, 'LONGFI~1.TXT'],
861     -             creationflags=subprocess.DETACHED_PROCESS
862     -         )
863     -
864     -         if p.returncode:
865     -             raise unittest.SkipTest('failed to set short name')
866     -
867     -         try:
868     -             self.assertPathEqual(test_file, ntpath.realpath(test_file_short))
869     -         except AssertionError:
870     -             raise unittest.SkipTest('the filesystem seems to lack support for
short filenames')
871     -
872     -         # Deny the right to [S]YNCHRONIZE on the file to
873     -         # force nt._getfinalpathname to fail with ERROR_ACCESS_DENIED.
874     -         p = subprocess.run(
875     -             ['icacls.exe', test_file, '/deny', '*S-1-5-32-545:(S)'],
876     -             creationflags=subprocess.DETACHED_PROCESS
877     -         )
878     -
879     -         if p.returncode:
880     -             raise unittest.SkipTest('failed to deny access to the test file')
881     -
882     -         self.assertPathEqual(test_file, ntpath.realpath(test_file_short))
883     -
884     839     def test_expandvars(self):
```

885 840

```
with os_helper.EnvironmentVarGuard() as env:
```

886 841

```
env.clear()
```



**Comments** 0