Meta

- Project: cpython
- Project Bug Identifier: bpo-30400
- Link to Bug on Project Bug Tracker: <imagine bugs.python.org url here>
- Current Stage: Bug Fix Accepted or Merged
- Brief description (This can change as you progress) :

The Python shutil module has a function called copyfile that has a race condition when the file being copied is modified. The race condition occurs between the time when it is initially checked and when it is actually copied.

Updates

Week 1

Current Stage: Bug Identified

Bug discovered when file move checker was used to analyze a dead trace of shutil.rename() moving a file from one disk to another under Linux. Further investigation revealed that this case occurs because moving a file this way means the rename() system call cannot be used. This leads to a fallback manual copy being performed.

Week 2

Current Stage: Source of Bug Diagnosed

Further work on this bug indicates that the problem exists because there the current implementation of the internal function copyfile() does not double check for changes in the file being copied. Further investigation is required to come up with a good way to fix this.

Week 3

Current Stage: Bug Fix Implemented

Analysis of coreutils mv source code indicates one approach to fix this problem is to store the inode number of the file being moved on the initial check. You can then confirm that it hasn't changed after being open()'d by using fstat() on its file descriptor. Fix has been implemented and is available on this fork: https://github.com/myname/cpython

Current Stage: Bug Fix Submitted to Project:

Bug has been reported to the cpython bug tracker and as an issue on cpython's github according to project standards. Bug has been assigned identifier bpo-30400. Bug tracker link: <link here> GitHub issue: link <link here>

Week 4

Current Stage: Communication with Project in Progress

Strategy implemented in fix was not accepted by project maintainers. They suggested an alternate strategy involving open()'ing the file as early as possible and using file descriptor-dependent system calls from then on out to resolve the race condition. Currently working on this new fix

Current Stage: Bug Fix Accepted or Merged

Fix with suggested strategy has been submitted, accepted, and merged. Bug has been marked as resolved at <Link to python bug tracker>