

Fixed a leak when the garbage collector fails to add an object with the `__del__` method or referenced by it into the `data:gc.garbage` list. `:c:func:PyGC_Collect` can now be called when an exception is set and preserves it.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\ [cpython-main] [Misc] [NEWS.d] 3.7.0b5.rst, line 7); [backlink](#)**

Unknown interpreted text role "data".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\ [cpython-main] [Misc] [NEWS.d] 3.7.0b5.rst, line 7); [backlink](#)**

Unknown interpreted text role "c:func".

Fix module `_globals` parameter of `warnings.warn_explicit()`: don't crash if module `_globals` is not a dict.

The new `os.posix_spawn` added in 3.7.0b1 was removed as we are still working on what the API should look like. Expect this in 3.8 instead.

Fixed miscellaneous bugs in converting annotations to strings and optimized parentheses in the string representation.

Fix a leak in `set_symmetric_difference()`.

Fix unaligned accesses in `siphash24()`. Patch by Rolf Eike Beer.

Due to unexpected compatibility issues discovered during downstream beta testing, reverted `:issue:29463`. `docstring` field is removed from `Module`, `ClassDef`, `FunctionDef`, and `AsyncFunctionDef` ast nodes which was added in 3.7a1. Docstring expression is restored as a first statement in their body. Based on patch by Inada Naoki.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\ [cpython-main] [Misc] [NEWS.d] 3.7.0b5.rst, line 67); [backlink](#)**

Unknown interpreted text role "issue".

Fix a crash in `ctypes.cast()` in case the type argument is a `ctypes` structured data type. Patch by Eryk Sun and Oren Milman.

When cancelling the task due to a timeout, `:meth:asyncio.wait_for` will now wait until the cancellation is complete.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\ [cpython-main] [Misc] [NEWS.d] 3.7.0b5.rst, line 90); [backlink](#)**

Unknown interpreted text role "meth".

Fix `gather` to propagate cancellation of itself even with `return_exceptions`.

Support protocol type switching in `SSLTransport.set_protocol()`.

Pause the transport as early as possible to further reduce the risk of `data_received()` being called before `connection_made()`.

Fix a race condition in `SSLProtocol.connection_made()` of `asyncio.sslproto`: start immediately the handshake instead of using `call_soon()`. Previously, `data_received()` could be called before the handshake started, causing the handshake to hang or fail.

Fixed bug where calling `write_eof()` on a `_SelectorSocketTransport` after it's already closed raises `AttributeError`.

Make `asyncio.all_tasks()` return only pending tasks.

Avoid blocking on file IO in `sendfile` fallback code

Fix `RuntimeError` after closing loop that used `run_in_executor`

Fix `Task.__repr__` crash with Cython's bogus coroutines

Fix `transport.set_protocol()` to support switching between `asyncio.Protocol` and `asyncio.BufferedProtocol`. Fix `loop.start_tls()` to work with `asyncio.BufferedProtocols`.

Pickles of type variables and subscripted generics are now future-proof and compatible with older Python versions.

Fixed `:func:uuid.uuid1` on FreeBSD.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\ [cpython-main] [Misc] [NEWS.d] 3.7.0b5.rst, line 207); [backlink](#)**

Unknown interpreted text role "func".

Finalize and document preliminary and experimental TLS 1.3 support with OpenSSL 1.1.1

Fix possible SIGSEGV when `asyncio.Future` is created in `__del__`

Fixed a bug in the Python implementation of the JSON decoder that prevented the cache of parsed strings from clearing after finishing the decoding. Based on patch by c-fos.

Change TLS 1.3 cipher suite settings for compatibility with OpenSSL 1.1.1-pre6 and newer. OpenSSL 1.1.1 will have TLS 1.3 ciphers enabled by default.

Do not simplify arguments to *typing.Union*. Now *Union[Manager, Employee]* is not simplified to *Employee* at runtime. Such simplification previously caused several bugs and limited possibilities for introspection.

Add a new `block_on_close` class attribute to `ForkingMixIn` and `ThreadingMixIn` classes of `mod:`socketserver``.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\ [cpython-main] [Misc] [NEWS.d] 3.7.0b5.rst, line 268); [backlink](#)**

Unknown interpreted text role "mod".

`tempfile._candidate_tempdir_list` should consider common TEMP locations

`argparse` subparsers are once again not required by default, reverting the change in behavior introduced by bpo-26510 in 3.7.0a2.

`dataclasses.make_dataclass` now checks for invalid field names and duplicate fields. Also, added a check for invalid field specifications.

Prevent `uuid.get_node` from using a DUID instead of a MAC on Windows. Patch by Zvi Effron

Fix race condition with `ReadTransport.resume_reading` in Windows proactor event loop.

Fix failure in `typing.get_type_hints()` when `ClassVar` was provided as a string forward reference.

Optimize `asyncio.ensure_future()` by reordering if checks: 1.17x faster.

Add errors param to `cgi.parse_multipart` and make an encoding in `FieldStorage` use the given errors (needed for Twisted). Patch by Amber Brown.

Change `dataclasses.Fields` repr to use the repr of each of its members, instead of str. This makes it more clear what each field actually represents. This is especially true for the 'type' member.

Fix `dataclasses` to work if using literal string type annotations or if using PEP 563 "Postponed Evaluation of Annotations". Only specific string prefixes are detected for both `ClassVar` ("ClassVar" and "typing.ClassVar") and `InitVar` ("InitVar" and "dataclasses.InitVar").

Minor fixes in `typing` module: add annotations to `NamedTuple.__new__`, pass `*args` and `**kwargs` in `Generic.__new__`. Original PRs by Paulius Ąrka and Chad Dombrova.

Updated alias mapping with glibc 2.27 supported locales.

Fix trailing quotation marks getting deleted when looking up byte/string literals on `pydoc`. Patch by Andr  s Delfino.

The function `platform.linux_distribution` and `platform.dist` now trigger a `DeprecationWarning` and have been marked for removal in Python 3.8

Update error message when constructing invalid inspect.Parameters Patch by Dong-hee Na.

Fix FD leak in `_SelectorSocketTransport` Patch by Vlad Starostin.

The `urllib.robotparser`'s `__str__` representation now includes wildcard entries and the "Crawl-delay" and "Request-rate" fields. Patch by Michael Lazar.

The `ssl` module now contains `OP_NO_RENEGOTIATION` constant, available with OpenSSL 1.1.0h or 1.1.1.

Support arrays >=2GiB in `mod:`ctypes``. Patch by Segev Finer.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Misc\NEWS.d\ [cpython-main] [Misc] [NEWS.d] 3.7.0b5.rst, line 460); [backlink](#)**

Unknown interpreted text role "mod".

Document that `asyncio.wait()` does not cancel its futures on timeout.

Document [PEP 567](#) changes to `asyncio`.

Update HMAC md5 default to a `DeprecationWarning`, bump removal to 3.8.

Fix broken pypi link

Add missing documentation for `typing.AsyncContextManager`.

Ignore `test_posix_fallocate` failures on BSD platforms that might be due to running on ZFS.

Remove the `_xxsubinterpreters` module (meant for testing) and associated helpers. This module was originally added recently in 3.7b1.

Ensures module definition files for the stable ABI on Windows are correctly regenerated.

Enable CI builds on Visual Studio Team Services at <https://python.visualstudio.com/cpython>

Add `-Wno-cast-function-type` for gcc 8 for silencing warnings about function casts like casting to PyCFunction in method definition lists.

The `.editrc` file in user's home directory is now processed correctly during the readline initialization through editline emulation on macOS.

IDLE: Cleanup `codecontext.py` and its test.

IDLE's code context now recognizes `async` as a block opener.

Add docstrings and tests for `codecontext`.