Set correct __cause__ for errors about invalid awaitables returned from __aiter__ and __anext__.

Fixes to memory allocation in _PyCode_SetExtra. Patch by Brian Coleman.

Fix minor regression of PyEval_CallObjectWithKeywords. It should raise TypeError when kwargs is not a dict. But it might cause segv when args=NULL and kwargs is not a dict.

Support __rmod__ for subclasses of str being called before str.__mod__. Patch by Martijn Pieters.

Fix stack_effect computation for CALL_FUNCTION_EX. Patch by Matthieu Dartiailh.

Fix incorrect handling of signed zeros in complex constructor for complex subclasses and for inputs having a __complex__ method. Patch by Serhiy Storchaka.

Fixed possibly dereferencing undefined pointers when creating weakref objects.

Fixed use-after-free problem in key sharing dict.

Prevent RunMainFromImporter overwriting sys.path[0].

Fixed possible BytesWarning when compare the code objects. Warnings could be emitted at compile time.

Fixed a crash when pass the iterable keyword argument to sorted().

Fix memory leak and use-after-free in os module (path_converter).

Fix regression in bytes(x) when x.__index__() raises Exception.

Do not include <sys/random.h> if it does not exist.

Correct the positioning of the syntax error caret for indented blocks. Based on patch by Michael Layzell.

Fixed bytes formatting of octals with zero padding in alternate form.

On Android, operating system data is now always encoded/decoded to/from UTF-8, instead of the locale encoding to avoid inconsistencies with os.fsencode() and os.fsdecode() which are already using UTF-8.

functools.lru_cache() was susceptible to an obscure reentrancy bug triggerable by a monkey-patched len() function.

f-string expressions are no longer accepted as docstrings and by ast.literal_eval() even if they do not include expressions.

Fixed setting the offset attribute of SyntaxError by PyErr_SyntaxLocationEx() and PyErr_SyntaxLocationObject().

Fix the cross compilation of xxlimited when Python has been built with Py_DEBUG defined.

Optimize _PyDict_NewPresized() to create correct size dict. Improve speed of dict literal with constant keys up to 30%.

Update zlib to 1.2.11.

Allow use of path-like object as a single argument in ConfigParser.read(). Patch by David Ellis.

Fix out of bound iteration in asyncio.Future.remove_done_callback implemented in C.

asyncio.subprocess.SubprocessStreamProtocol no longer closes before all pipes are closed.

Fix Task.current_task and Task.all_tasks implemented in C to accept None argument as their pure Python implementation.

Fix asyncio to support instantiation of new event loops in child processes.

Fix assertion error in threading._DummyThread.is_alive().

Add a test that checks that cwd parameter of Popen() accepts PathLike objects. Patch by Sayan Chowdhury.

Start a transaction implicitly before a DML statement. Patch by Aviv Palivoda.

Altering a kwarg dictionary passed to functools.partial() no longer affects a partial object after creation.

Fix file object leak in aifc.open() when file is given as a filesystem path and is not in valid AIFF format. Patch by Anthony Zhang.

Various updates to typing module: typing.Counter, typing.ChainMap, improved ABC caching, etc. Original PRs by Jelle Zijlstra, Ivan Levkivskyi, Manuel Krebber, and Łukasz Langa.

Fix datetime.fromtimestamp() regression introduced in Python 3.6.0: check minimum and maximum years.

Fix weakref spewing exceptions during interpreter shutdown when used with a rare combination of multiprocessing and custom codecs.

Prevent infinite loop in pathlib.Path.mkdir

Fixed out-of-bounds buffer access in the group() method of the match object. Based on patch by WGH.

Fix subprocess.Popen.wait() when the child process has exited to a stopped instead of terminated state (ex: when under ptrace).

Fix a regression in argparse that help messages would wrap at non-breaking spaces.

Fixed the comparison of mock.MagickMock with mock.ANY.

Restore the provisional status of typing module, add corresponding note to documentation. Patch by Ivan L.

Fixed infinite recursion in the repr of uninitialized ctypes.CDLL instances.

Fix an important omission by adding Deque to the typing module.

Fixed race condition in C implementation of functools.lru_cache. KeyError could be raised when cached function with full cache was simultaneously called from different threads with the same uncached arguments.

In urllib.request, suffixes in no_proxy environment variable with leading dots could match related hostnames again (e.g. .b.c matches a.b.c). Patch by Milan Oberkirch.

Fix unittest.mock._Call helper: don't ignore the name parameter anymore. Patch written by Jiajun Huang.

functools.lru_cache() now respects PEP 468 and preserves the order of keyword arguments. f(a=1, b=2) is now cached separately from f(b=2, a=1) since both calls could potentially give different results.

inspect.getframeinfo() now correctly shows the first line of a context. Patch by Sam Breese.

Offsets in a ZIP file created with extern file object and modes "w" and "x" now are relative to the start of the file.

Allow random.Random.seed() to use high quality OS randomness rather than the pid and time.

Fixed bug in secrets.randbelow() which would hang when given a negative input. Patch by Brendan Donegan.

Prevent infinite loop in pathlib.resolve() on Windows

Fixed recursion errors in large or resized curses.textpad.Textbox. Based on patch by Tycho Andersen.

Fix weakrefs in the pure python version of collections.OrderedDict move_to_end() method. Contributed by Andra Bogildea.

curses.ascii predicates now work correctly with negative integers.

old keys should not remove new values from WeakValueDictionary when collecting from another thread.

Remove editor artifacts from Tix.py.

Neaten-up empty population error on random.choice() by suppressing the upstream exception.

Fixed a crash when deallocate deep ElementTree.

Fix bugs in WeakValueDictionary.setdefault() and WeakValueDictionary.pop() when a GC collection happens in another thread.

Fixed a crash in resource.prlimit() when passing a sequence that doesn't own its elements as limits.

multiprocessing.set_forkserver_preload() would crash the forkserver process if a preloaded module instantiated some multiprocessing objects such as locks.

dbm.dumb now supports reading read-only files and no longer writes the index file when it is not changed.

The chown() method of the tarfile.TarFile class does not fail now when the grp module cannot be imported, as for example on Android platforms.

IDLE colors f-string prefixes (but not invalid ur prefixes).

Add 10% to coverage of IDLE's test_configdialog. Update and augment description of the configuration system.

Removes readme.txt from the installer

Ignores blank lines in ._pth files (Patch by Alexey Izbyshev)

Correctly handle special console filenames (patch by Eryk Sun)

Implement PEP 529 for io.FileIO (Patch by Eryk Sun)

Prevent crash when passing invalid arguments into msvcrt module.

winreg does not truncate string correctly (Patch by Eryk Sun)

Deprecate WindowsRegistryFinder and disable it by default.

Function PySlice_GetIndicesEx() is replaced with a macro if Py_LIMITED_API is not set or set to the value between 0x03050400 and 0x03060000 (not including) or 0x03060100 or higher.

Fixed the declaration of some public API functions. PyArg_VaParse() and PyArg_VaParseTupleAndKeywords() were not available in limited API. PyArg_ValidateKeywordArguments(), PyArg_UnpackTuple() and Py_BuildValue() were not available in limited API of version < 3.3 when PY_SSIZE_T_CLEAN is defined.

All stable API extensions added after Python 3.2 are now available only when Py_LIMITED_API is set to the PY_VERSION_HEX value of the minimum Python version supporting this API.

Link the documentation to its source file on GitHub.

Document smtpd.py as effectively deprecated and add a pointer to aiosmtpd, a third-party asyncio-based replacement.

Add canonical header link on each page to corresponding major version of the documentation. Patch by Matthias Bussonnier.

Fix Python 2 syntax in code for building the documentation.

Skip test_asyncore and test_eintr poll failures on macOS. Skip some tests of select.poll when running on macOS due to unresolved issues with the underlying system poll function on some macOS versions.

to match the behaviour of the `re.LOCALE` flag, test_re.test_locale_flag now uses `locale.getpreferredencoding(False)` to determine the candidate encoding for the test regex (allowing it to correctly skip the test when the default locale encoding is a multi-byte encoding)

Disallow -j0 to be combined with -T/-l in regrtest command line arguments.

Fix the tests that bind() a unix socket and raise PermissionError on Android for a non-root user.

Add the support.setswitchinterval() function to fix test_functools hanging on the Android armv7 qemu emulator.

sys.version and the platform module python_build(), python_branch(), and python_revision() functions now use git information rather than hg when building from a repo.

Update Windows build and OS X installers to use OpenSSL 1.0.2k.

Set Android compilation and link flags.

Fix implicit declaration of function _setmode. Patch by Masayuki Yamamoto

Removes hard dependency on hg.exe from PCBuild/build.bat

Added missed names to PC/python3.def.

lockf() is available on Android API level 24, but the F_LOCK macro is not defined in android-ndk-r13.

Fix the compilation error that occurs because if_nameindex() is available on Android API level 24, but the if_nameindex structure is not defined.

Do not add the directory for installing C header files and the directory for installing object code libraries to the cross compilation search paths. Original patch by Thomas Petazzoni.

Do not define sys.implementation._multiarch on Android.