# Bug 1797052 - CVE-2020-9391 kernel: brk discards top byte of addresses on aarch64, causing heap corruption in glibc malloc

**Keywords:** Security ×

**Status:** CLOSED ERRATA

**Alias:** None

**Product:** Fedora

**Component:** kernel

**Version:** 31

**Hardware:** Unspecified

**OS:** Unspecified

**Priority:** unspecified

**Severity:** high

**Target Milestone:** ---

**Assignee:** Kernel Maintainer List

**QA Contact:** Fedora Extras Quality Assurance

**Docs Contact:**

**URL:** https://koschei.fedoraproject.org/pac...

**Whiteboard:**

**Duplicates (1):** ~~1814238~~ (view as bug list)

**Depends On:**

**Blocks:** ARMTracker ~~1775043~~ ~~1790090~~ ~~1790091~~ ~~CVE-2020-9391~~

**TreeView+** depends on / blocked

**Reported:** 2020-01-31 19:52 UTC by Miro Hrončok

**Modified:** 2020-03-17 13:48 UTC (History)

**CC List:** 36 users (show)

**Fixed In Version:** kernel-5.5.6-201.fc31

**Doc Type:** If docs needed, set a value

**Doc Text:**

**Clone Of:**

**Environment:**

**Last Closed:** 2020-02-29 03:21:09 UTC

**Type:** Bug

**Dependent Products:**

---

| Attachments | (Terms of Use) |
|---|---|
| Add an attachment (proposed patch, testcase, etc.) | |

Miro Hrončok 2020-01-31 19:52:03 UTC — Description

```
Description of problem:
Package python27 fails to build from source in Fedora rawhide on aarch64:

...
test_args_error (test.test_io.CBufferedWriterTest) ... ok
test_close_error_on_close (test.test_io.CBufferedWriterTest) ... ok
test_constructor (test.test_io.CBufferedWriterTest) ... make: *** [Makefile:894: test] Segmentation fault (core dumped)
error: Bad exit status from /var/tmp/rpm-tmp.el2Exf (%check)

This is reproducible.

Version-Release: 2.7.17-1.fc32

Steps to Reproduce:
$ fedpkg clone python27
$ cd python27
$ fedpkg build

Additional info:
This package is tracked by Koschei. See:
https://koschei.fedoraproject.org/package/python27

The first failing build is https://koschei.fedoraproject.org/build/7736139

It includes gcc 10 and an glibc update.
```

Miro Hrončok 2020-02-03 10:42:16 UTC — Comment 1

```
python36:


Fatal Python error: Segmentation fault
Current thread 0x0000ffff73fff1e0 (most recent call first):
  File "/builddir/build/BUILD/Python-3.6.10/Lib/http/client.py", line 622 in _safe_read
  File "/builddir/build/BUILD/Python-3.6.10/Lib/http/client.py", line 472 in read
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/test_wsgiref.py", line 293 in run_client
  File "/builddir/build/BUILD/Python-3.6.10/Lib/threading.py", line 864 in run
  File "/builddir/build/BUILD/Python-3.6.10/Lib/threading.py", line 916 in _bootstrap_inner
  File "/builddir/build/BUILD/Python-3.6.10/Lib/threading.py", line 884 in _bootstrap
Thread 0x0000ffffaa47dcc0 (most recent call first):
  File "/builddir/build/BUILD/Python-3.6.10/Lib/socketserver.py", line 803 in write
  File "/builddir/build/BUILD/Python-3.6.10/Lib/wsgiref/handlers.py", line 453 in _write
  File "/builddir/build/BUILD/Python-3.6.10/Lib/wsgiref/handlers.py", line 279 in write
  File "/builddir/build/BUILD/Python-3.6.10/Lib/wsgiref/handlers.py", line 180 in finish_response
  File "/builddir/build/BUILD/Python-3.6.10/Lib/wsgiref/handlers.py", line 138 in run
  File "/builddir/build/BUILD/Python-3.6.10/Lib/wsgiref/simple_server.py", line 133 in handle
  File "/builddir/build/BUILD/Python-3.6.10/Lib/socketserver.py", line 724 in __init__
  File "/builddir/build/BUILD/Python-3.6.10/Lib/socketserver.py", line 364 in finish_request
  File "/builddir/build/BUILD/Python-3.6.10/Lib/socketserver.py", line 351 in process_request
  File "/builddir/build/BUILD/Python-3.6.10/Lib/socketserver.py", line 320 in _handle_request_noblock
  File "/builddir/build/BUILD/Python-3.6.10/Lib/socketserver.py", line 300 in handle_request
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/test_wsgiref.py", line 298 in test_interrupted_write
  File "/builddir/build/BUILD/Python-3.6.10/Lib/unittest/case.py", line 622 in run
  File "/builddir/build/BUILD/Python-3.6.10/Lib/unittest/case.py", line 670 in __call__
  File "/builddir/build/BUILD/Python-3.6.10/Lib/unittest/suite.py", line 122 in run
  File "/builddir/build/BUILD/Python-3.6.10/Lib/unittest/suite.py", line 84 in __call__
  File "/builddir/build/BUILD/Python-3.6.10/Lib/unittest/suite.py", line 122 in run
  File "/builddir/build/BUILD/Python-3.6.10/Lib/unittest/suite.py", line 84 in __call__
  File "/builddir/build/BUILD/Python-3.6.10/Lib/unittest/suite.py", line 122 in run
  File "/builddir/build/BUILD/Python-3.6.10/Lib/unittest/suite.py", line 84 in __call__
  File "/builddir/build/BUILD/Python-3.6.10/Lib/unittest/runner.py", line 176 in run
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/support/__init__.py", line 1921 in _run_suite
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/support/__init__.py", line 2017 in run_unittest
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/libregrtest/runtest.py", line 178 in test_runner
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/libregrtest/runtest.py", line 182 in runtest_inner
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/libregrtest/runtest.py", line 127 in runtest
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/libregrtest/main.py", line 407 in run_tests_sequential
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/libregrtest/main.py", line 514 in run_tests
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/libregrtest/main.py", line 617 in _main
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/libregrtest/main.py", line 582 in main
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/libregrtest/main.py", line 638 in main
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/regrtest.py", line 46 in _main
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/regrtest.py", line 50 in <module>
  File "/builddir/build/BUILD/Python-3.6.10/Lib/runpy.py", line 85 in _run_code
  File "/builddir/build/BUILD/Python-3.6.10/Lib/runpy.py", line 193 in _run_module_as_main
/var/tmp/rpm-tmp.tdlZds: line 67: 348542 Segmentation fault      (core dumped) WITHIN_PYTHON_RPM_BUILD= LD_LIBRARY_PATH=$ConfDir $ConfDir/python -m test.regrtest
-wW --slowest --findleaks -x test_distutils -x test_bdist_rpm -x test_gdb -x test_faulthandler


python35:
test_constructor (test.test_io.CBufferedWriterTest) ... Fatal Python error: Segmentation fault
```

```
Current thread 0x0000ffffadf43c80 (most recent call first):
/var/tmp/rpm-tmp.JpDd3o: line 42: 955334 Segmentation fault      (core dumped) WITHIN_PYTHON_RPM_BUILD= LD_LIBRARY_PATH=$ConfDir $ConfDir/python -m test.regrtest
--verbose --findleaks -x test_distutils -x test_faulthandler -x test_gdb
error: Bad exit status from /var/tmp/rpm-tmp.JpDd3o (%check)


python34:

test_constructor (test.test_io.CBufferedWriterTest) ... Fatal Python error: Segmentation fault
Current thread 0x0000ffff8d389c60 (most recent call first):
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/case.py", line 200 in handle
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/case.py", line 745 in assertRaises
  File "/builddir/build/BUILD/Python-3.4.10/Lib/test/test_io.py", line 1446 in test_constructor
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/case.py", line 618 in run
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/case.py", line 666 in __call__
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/suite.py", line 122 in run
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/suite.py", line 84 in __call__
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/suite.py", line 122 in run
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/suite.py", line 84 in __call__
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/suite.py", line 122 in run
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/suite.py", line 84 in __call__
  File "/builddir/build/BUILD/Python-3.4.10/Lib/unittest/runner.py", line 168 in run
  File "/builddir/build/BUILD/Python-3.4.10/Lib/test/support/__init__.py", line 1779 in _run_suite
  File "/builddir/build/BUILD/Python-3.4.10/Lib/test/support/__init__.py", line 1813 in run_unittest
  File "/builddir/build/BUILD/Python-3.4.10/Lib/test/regrtest.py", line 1278 in test_runner
/var/tmp/rpm-tmp.Pf2Vjj: line 41: 1623633 Segmentation fault      (core dumped) WITHIN_PYTHON_RPM_BUILD= LD_LIBRARY_PATH=$ConfDir $ConfDir/python -m
test.regrtest --verbose --findleaks -x test_distutils -x test_faulthandler -x test_gdb -x test_venv
```

Victor Stinner 2020-02-10 09:18:10 UTC Comment 2

Quick check with unpatched Python 2.7.17 (from python.org) compiled with gcc -O0 (disable all compiler optimizations): the whole test suite pass, only test_ctypes fails (see below). So this issue sounds like a compiler (GCC) issue.

I tested gcc-10.0.1-0.7.fc32.aarch64. The latest build (2020-01-23) used an old gcc 10.0.1-0.4.fc32: https://koschei.fedoraproject.org/package/python27

Maybe the fix has been fixed between gcc 10.0.1-0.4.fc32 and gcc-10.0.1-0.7.fc32.aarch64. Or maybe the issue comes from compiler optimizations.

--

test_ctypes failure:

```
test test_ctypes failed -- Traceback (most recent call last):
  File "/Python-2.7.17/Lib/ctypes/test/test_win32.py", line 130, in test_struct_by_value
    self.assertEqual(ret.left, left.value)
AssertionError: -200 != 10
```

* ~~https://bugzilla.redhat.com/show_bug.cgi?id=1174037~~
* https://bugs.python.org/issue32203
* https://bitbucket.org/cffi/cffi/issues/312/tests-failed-with-armv8
* https://bugs.gentoo.org/610626

Victor Stinner 2020-02-10 09:28:50 UTC Comment 3

I created scratch builds:

* python27-2.7.17-2.fc32.src.rpm: https://koji.fedoraproject.org/koji/taskinfo?taskID=41445285
* python36-3.6.10-2.fc32.src.rpm: https://koji.fedoraproject.org/koji/taskinfo?taskID=41445241

Victor Stinner 2020-02-10 09:44:53 UTC Comment 4

I compiled Python 3.6.10 (tarball from python.org) with "./configure && make": gcc -O3 (no PGO nor LTO): test_io pass successfully, but test_ctypes.test_callbacks() does crash with SIGILL:

```
<mock-chroot> sh-5.0# gdb -args ./python -m test -v test_ctypes
GNU gdb (GDB) Fedora 9.1-1.fc32
(...)
(gdb) run
Starting program: /builddir/Python-3.6.10/python -m test -v test_ctypes

== CPython 3.6.10 (default, Feb 10 2020, 09:29:03) [GCC 10.0.1 20200130 (Red Hat 10.0.1-0.7)]
== Linux-5.3.15-300.fc31.aarch64-aarch64-with-fedora-32-Rawhide little-endian
(...)
test_subclass (ctypes.test.test_arrays.ArrayTestCase) ... ok
test_byval (ctypes.test.test_as_parameter.AsParamPropertyWrapperTestCase) ... ok
test_callbacks (ctypes.test.test_as_parameter.AsParamPropertyWrapperTestCase) ...
Program received signal SIGILL, Illegal instruction.
0x0000fffff7fe5058 in ?? ()

(gdb) where
#0  0x0000fffff7fe5058 in ?? ()
#1  0x0000ffffe9799000 in ?? ()
#2  0x0000000000000010 in ?? ()
Backtrace stopped: previous frame identical to this frame (corrupt stack?)
```

Victor Stinner 2020-02-10 09:54:47 UTC Comment 5

> test_ctypes.test_callbacks() does crash with SIGILL

Florian Weimer wrote a fix one year ago which is part of libffi 3.3 release, but Fedora Rawhide still provides an old libffi-3.1-24.fc32.aarch64 (libffi 3.1 was released in 2014).

* https://github.com/libffi/libffi/commit/44a6c28545186d7864248792795284415 6fc7ab5
* https://github.com/libffi/libffi/issues/470
* https://bugs.python.org/issue36024

Victor Stinner 2020-02-10 10:06:50 UTC Comment 6

> Florian Weimer wrote a fix one year ago which is part of libffi 3.3 release, but Fedora Rawhide still provides an old libffi-3.1-24.fc32.aarch64 (libffi 3.1 was released in 2014).

~~bz1873890~~ proposes to upgrade to libffi 3.3.

Victor Stinner 2020-02-10 10:27:08 UTC Comment 7

> python27-2.7.17-2.fc32.src.rpm: https://koji.fedoraproject.org/koji/taskinfo?taskID=41445285

test_ctypes.test_constructor() crashed on Aarch64. "[GCC 10.0.1 20200130 (Red Hat 10.0.1-0.7)"

> python36-3.6.10-2.fc32.src.rpm: https://koji.fedoraproject.org/koji/taskinfo?taskID=41445241

Build successfully on Aarch64.

Victor Stinner 2020-02-10 10:54:27 UTC Comment 8

> test_ctypes.test_constructor() crashed on Aarch64. "[GCC 10.0.1 20200130 (Red Hat 10.0.1-0.7)"

I fail to reproduce this issue when I build Python manually :-( I tested:

```
Python 2.7.17 (default, Feb 10 2020, 10:24:12)
[GCC 10.0.1 20200130 (Red Hat 10.0.1-0.7)] on linux2

=> gcc-10.0.1-0.7.fc32.aarch64

Commands:

tar -xf Python-2.7.17.tar.xz
cd Python-2.7.17

export 'CFLAGS=-O2 -g -pipe -Wall -Werror=format-security -Wp,-D_FORTIFY_SOURCE=2 -Wp,-D_GLIBCXX_ASSERTIONS -fexceptions -fstack-protector-strong -grecord-gcc-
switches -specs=/usr/lib/rpm/redhat/redhat-hardened-cc1 -specs=/usr/lib/rpm/redhat/redhat-annobin-cc1 -fasynchronous-unwind-tables -fstack-clash-protection -
D_GNU_SOURCE -fPIC -fwrapv'
export 'OPT=-O2 -g -pipe -Wall -Werror=format-security -Wp,-D_FORTIFY_SOURCE=2 -Wp,-D_GLIBCXX_ASSERTIONS -fexceptions -fstack-protector-strong -grecord-gcc-
switches -specs=/usr/lib/rpm/redhat/redhat-hardened-cc1 -specs=/usr/lib/rpm/redhat/redhat-annobin-cc1 -fasynchronous-unwind-tables -fstack-clash-protection -
D_GNU_SOURCE -fPIC -fwrapv'
export 'LDFLAGS=-Wl,-z,relro -Wl,--as-needed  -Wl,-z,now -specs=/usr/lib/rpm/redhat/redhat-hardened-ld'

./configure --build=aarch64-redhat-linux-gnu --host=aarch64-redhat-linux-gnu --program-prefix= --disable-dependency-tracking --prefix=/usr --exec-prefix=/usr --
bindir=/usr/bin --sbindir=/usr/sbin --sysconfdir=/etc --datadir=/usr/share --includedir=/usr/include --libdir=/usr/lib64 --libexecdir=/usr/libexec --
localstatedir=/var --sharedstatedir=/var/lib --mandir=/usr/share/man --infodir=/usr/share/info --enable-ipv6 --enable-shared --enable-unicode=ucs4 --with-
dbmliborder=gdbm:ndbm:bdb --with-system-expat --with-system-ffi --with-dtrace --with-tapset-install-dir=/usr/share/systemtap/tapset

make 'EXTRA_CFLAGS=-O2 -g -pipe -Wall -Werror=format-security -Wp,-D_FORTIFY_SOURCE=2 -Wp,-D_GLIBCXX_ASSERTIONS -fexceptions -fstack-protector-strong -grecord-
gcc-switches -specs=/usr/lib/rpm/redhat/redhat-hardened-cc1 -specs=/usr/lib/rpm/redhat/redhat-annobin-cc1 -fasynchronous-unwind-tables -fstack-clash-protection -
D_GNU_SOURCE -fPIC -fwrapv ' -j5

LD_LIBRARY_PATH=$PWD ./python -m test -v test_ctypes
LD_LIBRARY_PATH=$PWD ./python -m test -v test_io

=> both tests pass successfully
```

Victor Stinner 2020-02-10 12:22:07 UTC Comment 9

I failed to reproduce the crash with: "rpmbuild --rebuild python27-2.7.17-2.fc32.src.rpm", all tests passed.

Ben Cotton 2020-02-11 16:33:54 UTC Comment 10

This bug appears to have been reported against 'rawhide' during the Fedora 32 development cycle.
Changing version to 32.

Miro Hrončok 2020-02-12 11:56:46 UTC Comment 11

This now blocks a setuptools upgrade, trough https://src.fedoraproject.org/rpms/python27/pull-request/6

Victor Stinner 2020-02-13 10:51:50 UTC Comment 12

New attempt using mock --force-arch.

On the host:
```
---
sudo dnf install qemu-user-static
mock -r fedora-32-aarch64 --forcearch aarch64 --init
mock -r fedora-32-aarch64 --forcearch aarch64 --install dnf
mock -r fedora-32-aarch64 --forcearch aarch64 --enable-network --shell
---
```

In the mock container:
```
---
dnf install -y fedpkg
# use git with HTTPS, since fedpkg wants to use Kerberos,
# but I failed to get a fedoraproject.org Kerberos token in mock
git clone https://src.fedoraproject.org/rpms/python27.git
cd python27

dnf install -y dnf-plugins-core
dnf builddep -y python27

fedpkg --release master srpm
rpmbuild --rebuild python*.src.rpm
---
```

Miro Hrončok 2020-02-13 12:31:11 UTC Comment 13

FTR here is a scratchbuild that packages the entire builddir for inspection:

https://koji.fedoraproject.org/koji/taskinfo?taskID=41477522

Unpack it in an empty folder via:

```
rpm2cpio python27-2.7.17-2.fc33.aarch64.rpm | cpio -idmv
```

This is how it was created (in case it is garbage collected):

```
-  WITHIN_PYTHON_RPM_BUILD= EXTRATESTOPTS="$EXTRATESTOPTS" make test
+  WITHIN_PYTHON_RPM_BUILD= EXTRATESTOPTS="$EXTRATESTOPTS" make test || (cd && tar -czvf %{buildroot}/builddir.tar.gz %{_builddir})

...

 %files
+/builddir.tar.gz
```

Miro Hrončok 2020-02-13 15:43:28 UTC Comment 14

On aarch64-test02.fedorainfracloud.org f31 (aarch64 vm), with up to date mock 2.0, I did not reproduce the issue (rawhide mockbuild --enablerepo=local).

Victor Stinner 2020-02-13 17:48:41 UTC Comment 15

I managed to reproduce the crash in a Fedora 31 VM (which is running on AArch64 baremetal):

```
dnf install fedpkg -y
fedpkg clone python27 --anonymous
cd python27
```

```
fedpkg mockbuild --mock-config fedora-rawhide-aarch64 --no-clean-all --enablerepo=local

Enter the mock with --shell:

cd /builddir/build/BUILD/Python-2.7.17/build/optimized
# command extract from: make test
LD_LIBRARY_PATH=/builddir/build/BUILD/Python-2.7.17/build/optimized ./python -Wd -3 -E -tt  /builddir/build/BUILD/Python-2.7.17/Lib/test/regrtest.py -l  test_io
```

Victor Stinner 2020-02-14 08:50:04 UTC Comment 16

Some notes on this issue.

== Hardware AArch64 vs emulated AArch64 ==

Reproduced on baremetal:

* Crashes have been seen on Koji which builds packages in a Fedora Rawhide mock container hosted on Fedora 31 which runs on AArch64 baremetal: Fedora 31 VM => Fedora Rawhide container
* I reproduced the issue in a Fedora Rawhide mock container running on Fedora 31 VM which runs on a Centos 8 which runs on AArch64 baremetal: Centos 8 => Fedora 31 VM => Fedora Rawhide container

Not reproduced on emulated AArch64:

* I failed to reproduce the issue on aarch64-test01.fedorainfracloud.org which is an AArch64 VM: <unknown OS but likely x86-64> => Fedora 31 AArch64 VM => Fedora Rawhide container
* Miro failed to reproduce the issue on aarch64-test02.fedorainfracloud.org which is also an AArch64 VM: <unknown OS but likely x86-64> => Fedora 31 AArch64 VM => Fedora Rawhide container
* I failed to reproduce the issue on an AArch64 container created by mock --force-arch=aarch64 running on my x86-64 laptop: Fedora 31 (x86-64) => Fedora Rawhide AArch64 container (QEMU User Mode)

== Packages and tests ==

Crashs seen in packages:

* python27
* python34
* python35
* python36

It seems like building python27 trigger the crash at each package build (like 3 failures on 3 builds: 100%), whereas it occurs randomly when building the python36 package (1 failure on between 3 and 5 attempts, I don't recall, sorry).

Tests which crashed:

* python27: test_io.CBufferedWriterTest.test_constructor()
* python34: test_io.CBufferedWriterTest.test_constructor()
* python35: test.test_io.CBufferedWriterTest.test_constructor()
* python36:

  * test_wsgiref: test_interrupted_write()
  * test_random.test_choices_algorithms()

== How tests are run ==

The python27 and python36 packages run all test files in the same process (-jN option not used): "Run tests sequentially". I didn't check python34 and python35, but they are likely doing the same.

python27:

```
LD_LIBRARY_PATH=/builddir/build/BUILD/Python-2.7.17/build/optimized ./python -Wd -3 -E -tt  /builddir/build/BUILD/Python-2.7.17/Lib/test/regrtest.py -l --verbose
-x test_gdb
== CPython 2.7.17 (default, Jan 30 2020, 00:00:00) [GCC 10.0.1 20200130 (Red Hat 10.0.1-0.7)]
==   Linux-5.4.17-200.fc31.aarch64-aarch64-with-fedora-32-Rawhide little-endian
==   /builddir/build/BUILD/Python-2.7.17/build/test_python_26593
== CPU count: 8
Run tests sequentially
(...)
test_readonly_attributes (test.test_io.PyBufferedReaderTest) ... ok
test_repr (test.test_io.PyBufferedReaderTest) ... ok
test_threads (test.test_io.PyBufferedReaderTest) ... skipped "resource u'cpu' is not enabled"
test_uninitialized (test.test_io.PyBufferedReaderTest) ... ok
test_args_error (test.test_io.CBufferedWriterTest) ... ok
test_close_error_on_close (test.test_io.CBufferedWriterTest) ... ok
test_constructor (test.test_io.CBufferedWriterTest) ... make: *** [Makefile:894: test] Segmentation fault (core dumped)
error: Bad exit status from /var/tmp/rpm-tmp.2GniPt (%check)
```

python36:

```
STARTING: CHECKING OF PYTHON FOR CONFIGURATION: optimized
+ WITHIN_PYTHON_RPM_BUILD=
+ LD_LIBRARY_PATH=/builddir/build/BUILD/Python-3.6.10/build/optimized
+ /builddir/build/BUILD/Python-3.6.10/build/optimized/python -m test.regrtest -wW --slowest --findleaks -x test_distutils -x test_bdist_rpm -x test_gdb -x
test_faulthandler
== CPython 3.6.10 (default, Jan 30 2020, 00:00:00) [GCC 10.0.1 20200130 (Red Hat 10.0.1-0.7)]
== Linux-5.4.17-200.fc31.aarch64-aarch64-with-fedora-32-Rawhide little-endian
== cwd: /builddir/build/BUILD/Python-3.6.10/build/optimized/build/test_python_26844
== CPU count: 8
== encodings: locale=UTF-8, FS=utf-8
Run tests sequentially
(...)
0:14:44 load avg: 0.62 [267/405] test_random
Fatal Python error: Segmentation fault

Current thread 0x0000ffff97239cc0 (most recent call first):
  File "/builddir/build/BUILD/Python-3.6.10/Lib/random.py", line 356 in choices
  File "/builddir/build/BUILD/Python-3.6.10/Lib/test/test_random.py", line 696 in test_choices_algorithms
  (...)
```

== Reproduce the issue manually ==

The python27 bug is the easiest to reproduce manually:

```
cd /builddir/build/BUILD/Python-2.7.17/build/optimized
LD_LIBRARY_PATH=/builddir/build/BUILD/Python-2.7.17/build/optimized ./python -Wd -3 -E -tt  /builddir/build/BUILD/Python-2.7.17/Lib/test/regrtest.py -l --verbose
test_io
```

Output:

```
(...)
test_writelines_userlist (test.test_io.CBufferedRandomTest) ... ok
test_writes (test.test_io.CBufferedRandomTest) ... ./bug.sh: line 1:   117 Segmentation fault      (core dumped) LD_LIBRARY_PATH=/builddir/build/BUILD/Python-
2.7.17/build/optimized ./python -Wd -3 -E -tt /builddir/build/BUILD/Python-2.7.17/Lib/test/regrtest.py -l --verbose test_io
```

Sadly, any subtle change in the command line, memory allocators, source code, etc. makes the bug disappear :-( It makes the bug really hard to investigate.

== Debug ==

gdb traceback in python27:

```
test_writes (test.test_io.CBufferedRandomTest) ...
Program received signal SIGSEGV, Segmentation fault.
0x0000fffff7cba478 in _int_malloc () from /lib64/libc.so.6
Missing separate debuginfos, use: dnf debuginfo-install bzip2-libs-1.0.8-2.fc32.aarch64 openssl-libs-1.1.1d-6.fc32.aarch64 zlib-1.2.11-21.fc32.aarch64
(gdb) where
#0  0x0000fffff7cba478 in _int_malloc () from /lib64/libc.so.6
#1  0x0000fffff7cbb29c in malloc () from /lib64/libc.so.6
#2  0x0000fffff7e48ca0 in string_repeat (a=0xffffe9fff290, n=<optimized out>) at /builddir/build/BUILD/Python-2.7.17/Objects/stringobject.c:1115
#3  0x0000fffff7e9d2d0 in PyEval_EvalFrameEx (
    f=f@entry=Frame 0xaaaaaad6af30, for file /builddir/build/BUILD/Python-2.7.17/Lib/test/test_io.py, line 1185, in check_writes (self=
<CBufferedRandomTest(_resultForDoCleanups=<TextTestResult(_original_stdout=<file at remote 0xfffff7af51e0>, dots=False, skipped=[(<CIOTest(_resultForDoCleanups=
```

```
<...>, _type_equality_funcs={<type at remote 0xfffff7f7ce18>: 'assertTupleEqual', <type at remote 0xfffff7f81e78>: 'assertMultiLineEqual', <type at remote
0xfffff7f74690>: 'assertListEqual', <type at remote 0xfffff7f78b10>: 'assertSetEqual', <type at remote 0xfffff7f78c98>: 'assertSetEqual', <type at remote
0xfffff7f767e8>: 'assertDictEqual'}, _testMethodDoc=None, _testMethodName='test_unbounded_file', _cleanups=[]) at remote 0xfffffe9fa4250>, 'test can only run in a
32-bit address space'), (<PyIOTest(_resultForDoCleanups=<...>, _type_equality_funcs={<type at remote 0xfffff7f7ce18>: 'assertTupleEqual', <type at remote
0xfffff7f81e78>: 'assertMultiLineEqual', <type at remote 0xfffff7f74690>: 'assertListEqual', <type at remote 0xfffff7f...(truncated),
throwflag=throwflag@entry=0)
    at /builddir/build/BUILD/Python-2.7.17/Python/ceval.c:1485
(...)

A crash in malloc() is very likely a memory overflow which occurred "previously".

I tried different things to make the bug more likely or to get more information when it happens:

* set MALLOC_CHECK_=2 or MALLOC_CHECK_=3 environment variable: enable glibc memory debugger
* use Valgrind: pymalloc allocator of python27 emits tons of false alarm. python27 should be rebuilt with ./configure --with-valgrind and Valgrind should use
Misc/valgrind.suppr suppression file of Python. But I had troubles to reproduce the issue if I modify the code. I should try again.
* Use python36 which has builtin memory debugger which can be enabled with PYTHONMALLOC=debug at runtime (no need to rebuild). Sadly, the bug is really hard to
reproduce on python36. On python36, -X dev command line option can be used to enable PYTHONMALLOC=debug.
```

Victor Stinner 2020-02-14 16:10:14 UTC Comment 17

```
I can reproduce the crash with the following commands which don't use the Fedora package at all, only Python tarball from python.org:
---
set -e -x
curl -O https://www.python.org/ftp/python/2.7.17/Python-2.7.17.tar.xz
tar -xf Python-2.7.17.tar.xz
cd Python-2.7.17
mkdir build
cd build
../configure -C \
 --enable-ipv6 \
 --enable-shared \
 --enable-unicode=ucs4 \
 --with-system-expat \
 --with-system-ffi \
 CC=gcc \
 'LDFLAGS=-specs=/usr/lib/rpm/redhat/redhat-hardened-ld'
make clean
CFLAGS='-fno-strict-aliasing -O2 -g -pipe -Wp,-D_FORTIFY_SOURCE=2 -fstack-protector-strong -grecord-gcc-switches -specs=/usr/lib/rpm/redhat/redhat-hardened-cc1 -
fstack-clash-protection -D_GNU_SOURCE -fPIC -fwrapv -DNDEBUG'
make EXTRA_CFLAGS="$CFLAGS" -j12
LD_LIBRARY_PATH=$PWD ./python -m test -v test_io
---

Reproduced with versions:
---
# rpm -q gcc glibc redhat-rpm-config
gcc-9.2.1-1.fc32.3.aarch64
glibc-2.30.9000-29.fc32.aarch64
redhat-rpm-config-147-1.fc32.noarch
---
```

Victor Stinner 2020-02-14 16:31:08 UTC Comment 18

```
Even more simplified commands to configure + make Python:

../configure -C --enable-shared --enable-unicode=ucs4 --with-system-expat --with-system-ffi  CC=gcc OPT=''
make -j12 EXTRA_CFLAGS='-fno-strict-aliasing -O2 -fwrapv -DNDEBUG'
LD_LIBRARY_PATH=$PWD ./python -m test -v test_io
```

Victor Stinner 2020-02-14 16:49:30 UTC Comment 19

```
Oh, I can now reproduce the crash on Fedora 31 AArch64 as well, using the commands of Comment 17 + Comment 18,

(...)
[vstinner@python-builder-fedora-stable-aarch64 build]$ LD_LIBRARY_PATH=$PWD ./python -m test -v test_io
(...)
test_writes (test.test_io.CBufferedRandomTest) ... Segmentation fault (core dumped)

[vstinner@python-builder-fedora-stable-aarch64 build]$ rpm -q gcc glibc redhat-rpm-config
gcc-9.2.1-1.fc31.aarch64
glibc-2.30-10.fc31.aarch64
redhat-rpm-config-142-1.fc31.noarch

[vstinner@python-builder-fedora-stable-aarch64 build]$ uname -a
Linux python-builder-fedora-stable-aarch64 5.4.17-200.fc31.aarch64 #1 SMP Sat Feb 1 18:45:35 UTC 2020 aarch64 aarch64 aarch64 GNU/Linux
```

Victor Stinner 2020-02-14 16:59:59 UTC Comment 20

```
Valgrind doesn't see any error. I configure without pymalloc (Python memory allocator), so glibc malloc() is used directly.

<mock-chroot> sh-5.0# LD_LIBRARY_PATH=$PWD valgrind --log-file=valgrind.log ./python -m test -v test_io
(...)

<mock-chroot> sh-5.0# cat valgrind.log
==40376== Memcheck, a memory error detector
==40376== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==40376== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==40376== Command: ./python -m test -v test_io
==40376== Parent PID: 2
==40376==
==40376==
==40376== HEAP SUMMARY:
==40376==     in use at exit: 4,051,358 bytes in 24,065 blocks
==40376==   total heap usage: 11,690,828 allocs, 11,666,763 frees, 1,197,486,256 bytes allocated
==40376==
==40376== LEAK SUMMARY:
==40376==    definitely lost: 0 bytes in 0 blocks
==40376==    indirectly lost: 0 bytes in 0 blocks
==40376==      possibly lost: 1,695,882 bytes in 8,677 blocks
==40376==    still reachable: 2,355,476 bytes in 15,388 blocks
==40376==         suppressed: 0 bytes in 0 blocks
==40376== Rerun with --leak-check=full to see details of leaked memory
==40376==
==40376== For lists of detected and suppressed errors, rerun with: -s
==40376== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)


--

Using jemalloc, the test doesn't crash anymore:

<mock-chroot> sh-5.0# LD_PRELOAD=/usr/lib64/libjemalloc.so.2 LD_LIBRARY_PATH=$PWD ./python -m test -v test_io
(...)
Tests result: SUCCESS

--

Using MALLOC_CHECK_=3, the bug hides as well:

<mock-chroot> sh-5.0# MALLOC_CHECK_=3 LD_LIBRARY_PATH=$PWD ./python -m test -v test_io
(...)
```

```
Tests result: SUCCESS
```

Victor Stinner 2020-02-14 17:11:18 UTC Comment 21

Simplified commands which reproduces the issue on Fedora 31 AArch64:

```
./configure OPT="-O3 -ggdb" --without-pymalloc && make clean && make -j10 && ./python -m test -v test_io
```

But test_io doesn't crash if Python is built with gcc -O2. It may be a compiler bug.

Note: I never saw this bug on other architectures. It seems specific to AArch64.

Victor Stinner 2020-02-14 23:55:34 UTC Comment 22

New reproducer full script:

```
---
set -e -x

# disable ASLR
sudo bash -c 'echo 0 > /proc/sys/kernel/randomize_va_space'

test -e Python-2.7.17.tar.xz || curl -O https://www.python.org/ftp/python/2.7.17/Python-2.7.17.tar.xz
tar -xf Python-2.7.17.tar.xz
cd Python-2.7.17
mkdir build
cd build
../configure -C OPT="-O0 -ggdb" --without-pymalloc
make -j10

cat > tests << EOF
test.test_io.BufferedRandomTest.test_threads
test.test_io.CBufferedRandomTest.test_constructor
test.test_io.CBufferedRandomTest.test_writes_and_seeks
test.test_io.PyBufferedWriterTest.test_writes_and_seeks
EOF

export PYTHONHASHSEED=424969

./python -m test --matchfile=tests -v test_io
# sometimes the first run behaves differently because of the creaton of .pyc files
./python -m test --matchfile=tests -v test_io
---
```

I disabled ASLR and set a fixed Python hash seed to reduce randomness. I also disabled SELinux, just in case.

Shorter script to reproduce the crash:

```
---
set -e -x
PYTHONHASHSEED=424969 ./python -m test -m test.test_io.CBufferedRandomTest.test_constructor -m test.test_io.CBufferedRandomTest.test_writes_and_seeks -m
test.test_io.PyBufferedWriterTest.test_writes_and_seeks -v test_io
# sometimes the first run doesn't crash
PYTHONHASHSEED=424969 ./python -m test -m test.test_io.CBufferedRandomTest.test_constructor -m test.test_io.CBufferedRandomTest.test_writes_and_seeks -m
test.test_io.PyBufferedWriterTest.test_writes_and_seeks -v test_io
---
```

Victor Stinner 2020-02-14 23:56:56 UTC Comment 23

After a very long refactoring...

I manage to simply test_io (3409 lines of Python code, ignoring all imports) to just 3 malloc+free calls...

Reproducer C program:

```
---
#include <stdio.h>
#include <stdlib.h>

#define PY_SSIZE_T_MAX ((ssize_t)(((size_t)-1)>>1))

void my_alloc(size_t size)
{
        void *ptr;
        ptr = malloc(size);
        if (ptr != NULL) {
                printf("malloc(%zu) -> ok\n", size);
                free(ptr);
        }
        else {
                printf("malloc(%zu) -> FAIL\n", size);
        }
}

int main()
{
        int i;
        for(i=0; i<2; i++) {
                my_alloc(1170037);
        }

        my_alloc(PY_SSIZE_T_MAX);

        for(i=0; i<4; i++) {
                my_alloc(1170037);
        }

        printf("ok\n");
        return 0;
}
---
```

Example:

```
---
malloc(1170037) -> ok
malloc(1170037) -> ok
malloc(9223372036854775807) -> FAIL
Segmentation fault (core dumped)
---
```

Victor Stinner 2020-02-15 00:19:57 UTC Comment 24

I can still reproduce the crash with glibc-2.30-4.fc31.aarch64 which is the oldest version available on Koji for Fedora 31: glibc-2.30-3.fc31 has been trashed.

Victor Stinner 2020-02-15 00:23:19 UTC Comment 25

Centos 8 AArch64 doesn't seem to be affected: Comment 23 reproducer doesn't crash.

```
$ rpm -q glibc
glibc-2.28-72.el8_1.1.aarch64
```

Victor Stinner 2020-02-15 01:14:44 UTC Comment 26

This bug was tricky to detect since Valgrind didn't complain, the bug was worked around when using jemalloc (hint!), and MALLOC_CHECK_ also hides the bug!

```
[vstinner@python-builder-fedora-stable-aarch64 ~]$ MALLOC_CHECK_=3 ./malloc
malloc(1170037) -> ok
malloc(1170037) -> ok
malloc(9223372036854775807) -> FAIL
malloc(1170037) -> ok
malloc(1170037) -> ok
malloc(1170037) -> ok
malloc(1170037) -> ok
ok
```

Victor Stinner 2020-02-15 01:22:42 UTC Comment 27

Internally, malloc(PY_SSIZE_T_MAX) calls mmap(PY_SSIZE_T_MAX) which fails and then sbrk(0x7fffffffffee2000) which also fails. After that, the next malloc() call does crash.

The problem is that the sbrk() calls *reduces* the size of the heap from 1,306,624 bytes to 135,168 bytes.

Fedora Release Engineering 2020-02-16 04:34:18 UTC Comment 28

Dear Maintainer,

your package has not been built successfully in 32. Action is required from you.

If you can fix your package to build, perform a build in koji, and either create
an update in bodhi, or close this bug without creating an update, if updating is
not appropriate [1]. If you are working on a fix, set the status to ASSIGNED to
acknowledge this. Following the latest policy for such packages [2], your package
will be orphaned if this bug remains in NEW state more than 8 weeks.

A week before the mass branching of Fedora 33 according to the schedule [3],
any packages not successfully rebuilt at least on Fedora 31 will be
retired regardless of the status of this bug.

[1] https://fedoraproject.org/wiki/Updates_Policy
[2] https://docs.fedoraproject.org/en-US/fesco/Fails_to_build_from_source_Fails_to_install/
[3] https://fedoraproject.org/wiki/Releases/33/Schedule

Miro Hrončok 2020-02-16 15:26:58 UTC Comment 29

Removing the F32FTBFS tracker, gcc builds just fine, it's other packages that don't build.

Florian Weimer 2020-02-18 12:33:24 UTC Comment 31

Catalin Marinas posted a kernel patch (“mm: Avoid creating virtual address aliases in brk()/mmap()/mremap()”):

http://lists.infradead.org/pipermail/linux-arm-kernel/2020-February/712003.html

Victor Stinner 2020-02-18 12:34:31 UTC Comment 32

This issue is a regression caused by the following kernel commit which landed in Linux kernel 5.4 which has been released at Nov 24, 2019:
https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=ce18d171cb7368557e6498a3ce111d7d3dc03e4d

Koji was running Linux kernel 5.3 or older until mi-January 2020 (I don't know the exact day). We only started to notice the crash recently when we tried to rebuild python27 which reproduces the crash in a reliable way.

The first python27 build which failed ran at Jan 20, 2020 and it used a kernel 5.5.0 according to RPM installed in the buildroot (but the build doesn't provide all logs, so I'm not 100% sure). The previous (successful) python27 build was done in 2019-10-21.

Miro Hrončok 2020-02-19 16:22:45 UTC Comment 33

The aarch64-test02.fedorainfracloud.org f31 aarch64 vm got an updated kernel, so we might be able to reproduce there now as well.

~~Jeremy Cline~~ 2020-02-19 21:39:09 UTC Comment 34

I've picked the fix up for f30 in kernel-5.4.21-100.fc30 along with today's Rawhide build (kernel-5.6.0-0.rc2.git2.1.fc33). It should also arrive in F31 via the rebase to v5.5.5.

Miro Hrončok 2020-02-20 07:35:04 UTC Comment 35

Thanks, Jeremy!

Miro Hrončok 2020-02-23 08:16:34 UTC Comment 36

Fedora Infra ticket to update the kernel on aarch64 Koji: https://pagure.io/fedora-infrastructure/issue/8677

Miro Hrončok 2020-02-23 18:30:29 UTC Comment 37

(In reply to Miro Hrončok from comment #33)
> The aarch64-test02.fedorainfracloud.org f31 aarch64 vm got an updated
> kernel, so we might be able to reproduce there now as well.

FTR I was able to reproduce the crash on aarch64-test01.fedorainfracloud.org (not enough disk space on aarch64-test02) with kernel 5.4.19-200.fc31.aarch64.

Charalampos Stratakis 2020-02-24 15:40:28 UTC Comment 38

There hasn't been a new fixed build for F31 as the fix was applied on top of the 5.5.5-200 rebase. Could a build be created?

~~Jeremy Cline~~ 2020-02-24 15:53:20 UTC Comment 39

v5.5.6 just got released so it'll be in today's kernel-5.5.6-200.fc31 build, sorry about that.

Fedora Update System 2020-02-25 12:29:20 UTC Comment 40

FEDORA-2020-3cd64d683c has been submitted as an update to Fedora 31. https://bodhi.fedoraproject.org/updates/FEDORA-2020-3cd64d683c

Florian Weimer 2020-02-25 13:41:43 UTC Comment 41

Upstream commit: https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=dcde237319e626d1ec3c9d8b7613032f0fd4663a

Florian Weimer 2020-02-25 18:04:43 UTC Comment 43

Posted to oss-security: https://www.openwall.com/lists/oss-security/2020/02/25/6

Fedora Update System 2020-02-27 18:34:51 UTC Comment 44

kernel-5.5.6-201.fc31, kernel-headers-5.5.6-200.fc31, kernel-tools-5.5.6-200.fc31 has been pushed to the Fedora 31 testing repository. If problems still persist, please make note of it in this bug report.
See https://fedoraproject.org/wiki/QA:Updates_Testing for
instructions on how to install test updates.
You can provide feedback for this update here: https://bodhi.fedoraproject.org/updates/FEDORA-2020-3cd64d683c

Fedora Update System 2020-02-29 03:21:09 UTC Comment 45

kernel-5.5.6-201.fc31, kernel-headers-5.5.6-200.fc31, kernel-tools-5.5.6-200.fc31 has been pushed to the Fedora 31 stable repository. If problems still persist, please make note of it in this bug report.

Victor Stinner 2020-02-29 16:05:56 UTC Comment 46

Using aarch64-test01.fedorainfracloud.org, I was able to reproduce the crash using Comment 23 program (C code using malloc) on 5.4.19-200.fc31.aarch64.

I confirm that I'm not longer able to reproduce the crash on 5.5.6-201.fc31.aarch64.

Thanks for the fix ;-)

Justin M. Forbes 2020-03-17 13:48:42 UTC Comment 47

*** ~~Bug 1814230~~ has been marked as a duplicate of this bug. ***

Note
You need to log in before you can comment on or make changes to this bug.