

# Corrigindo Bugs no CPython

# Só entre nós...

Python tem bugs...

(não contem pra ninguém 😜)

**olist**

# O Bug

```
RuntimeError: dictionary changed size during iteration
File "python3.5/runpy.py", line 170, in _run_module_as_main
    "__main__", mod_spec)
File "python3.5/runpy.py", line 85, in _run_code
    exec(code, run_globals)
```

```
[ ...segue... ]
```

```
File "requests/sessions.py", line 531, in get
    return self.request('GET', url, **kwargs)
File "requests/sessions.py", line 509, in request
    prep.url, proxies, stream, verify, cert
File "requests/sessions.py", line 686, in merge_environment_settings
    env_proxies = get_environ_proxies(url, no_proxy=no_proxy)
File "requests/utils.py", line 696, in get_environ_proxies
    return getproxies()
File "urllib/request.py", line 2393, in getproxies_environment
    for name, value in os.environ.items():
File "_collections_abc.py", line 676, in __iter__
    for key in self._mapping:
File "python3.5/os.py", line 702, in __iter__
    for key in self._data:
```

# O Bug

Race condition durante a iteração sobre as variáveis de ambiente do sistema operacional

# Implementando uma solução

... antes de submeter o bug...

# Tentativa #1

- Copia o dicionário antes de iterar sobre ele

```
1  diff --git a/Lib/os.py b/Lib/os.py
2  index e293eca..5d28572 100644
3  --- a/Lib/os.py
4  +++ b/Lib/os.py
5  @@ -697,7 +697,8 @@ class _Environ(MutableMapping):
6       raise KeyError(key) from None
7
8       def __iter__(self):
9  -         for key in self._data:
10  +         data = dict(self._data)
11  +         for key in data:
12             yield self.decodekey(key)
13
14     def __len__(self):
```

# Tentativa #1

(fazer teste)

- Copia o dicionário antes de iterar sobre ele

```
15 diff --git a/Lib/test/test_os.py b/Lib/test/test_os.py
16 index 746b3f8..9cbe4b9 100644
17 --- a/Lib/test/test_os.py
18 +++ b/Lib/test/test_os.py
19 @@ -828,6 +828,21 @@ class EnvironTests(mapping_tests.BasicTestMappingProtocol):
20         self.assertIs(cm.exception.args[0], missing)
21         self.assertTrue(cm.exception.__suppress_context__)
22
23 +     def test_iter_error_when_os_environ_changes(self):
24 +         def _iter_environ_change():
25 +             for key, value in os.environ.items():
26 +                 yield key, value
27 +
28 +         iter_environ = _iter_environ_change()
29 +         key, value = next(iter_environ) # start iteration over os.environ
30 +
31 +         # add a new key in os.environ mapping
32 +         new_key = "__{}".format(key)
33 +         os.environ[new_key] = value
34 +
35 +         next(iter_environ) # force iteration over modified mapping
36 +         self.assertEqual(os.environ[new_key], value)
37 +
38
39 class WalkTests(unittest.TestCase):
```

# Tentativa #1

The screenshot shows the Python bug tracker interface for issue 30441. The browser address bar shows `bugs.python.org/issue30441`. The page header includes the Python logo and a search bar. The left sidebar contains navigation links for Python Home, About, News, Documentation, Downloads, Community, Foundation, Developer's Guide, and the Issue Tracker. The main content area displays the issue details, including classification, process, and a list of files. The issue is titled "os.environ raises RuntimeError: dictionary changed size during iteration" and is classified as a behavior issue, resolved, affecting the Library (Lib) component in Python 3.5, 3.6, and 3.7. The issue is closed, with a resolution of fixed. The process section shows dependencies, assigned to, nosy list, priority, and keywords. The files section lists four diff files uploaded by osantana. The pull requests section shows three merged pull requests. The messages section shows a message from Osvaldo Santana Neto (osantana) dated 2017-05-23 14:35.

**classification**

<b>Title:</b> os.environ raises RuntimeError: dictionary changed size during iteration	
<b>Type:</b> behavior	<b>Stage:</b> resolved
<b>Components:</b> Library (Lib)	<b>Versions:</b> Python 3.7, Python 3.6, Python 3.5

**process**

<b>Status:</b> closed	<b>Resolution:</b> fixed
<b>Dependencies:</b>	<b>Superseder:</b>
<b>Assigned To:</b>	<b>Nosy List:</b> Danilo Shiga, Jelle Zijlstra, mark.dickinson, martin.panter, osantana, pitrou, serhiy.storchaka
<b>Priority:</b> normal	<b>Keywords:</b> patch

Created on 2017-05-23 14:35 by osantana, last changed 2017-07-05 03:46 by osantana. This issue is now **closed**.

**Files**

File name	Uploaded	Description	Edit
<a href="#">fix_os_environ_iteration_issue.diff</a>	osantana, 2017-05-23 14:35	patch with fix and test	
<a href="#">fix_os_environ_iter_issue_low_level_thread_lock.diff</a>	osantana, 2017-05-24 22:37	(use this) use <code>_thread.allocate_lock()</code> as lock provider	
<a href="#">fix_os_environ_iter_issue_threading_lock.diff</a>	osantana, 2017-05-24 22:46		
<a href="#">fix_os_environ_iter_issue_low_level_thread_lock_2.diff</a>	osantana, 2017-05-25 17:08		

**Pull Requests**

URL	Status	Linked	Edit
<a href="#">PR 2409</a>	merged	osantana, 2017-06-26 14:32	
<a href="#">PR 2556</a>	merged	serhiy.storchaka, 2017-07-04 04:22	
<a href="#">PR 2557</a>	merged	serhiy.storchaka, 2017-07-04 04:24	

**Messages (27)**

[msg294250 - \(view\)](#) **Author: Osvaldo Santana Neto (osantana) \*** **Date: 2017-05-23 14:35**

We're facing ocasional RuntimeError exceptions in a multithreaded application when one of the threads creates new entries to the environment (os.environ).

I'm not so sure if the attached patch fixes this issue the right way, so, feel free to propose another approach.

<https://bugs.python.org/issue30441>

olist



# Tentativa #2

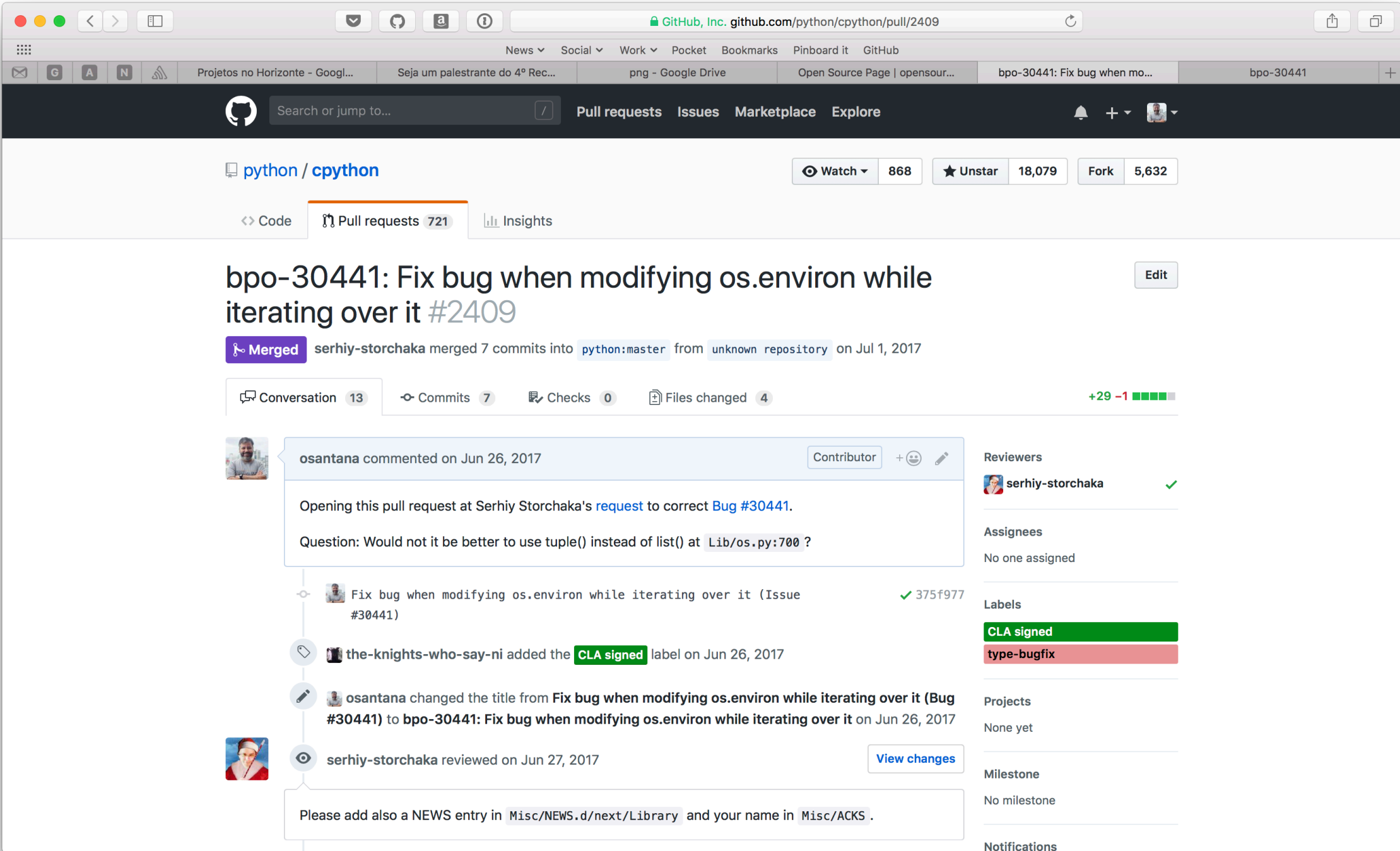
```
1 diff --git a/Lib/os.py b/Lib/os.py
2 index e293ecae7f..4dfa41c712 100644
3 --- a/Lib/os.py
4 +++ b/Lib/os.py
5 @@ -25,6 +25,7 @@ and opendir), and leave all pathname manipulation to os.path
6     import abc
7     import sys, errno
8     import stat as st
9     +import _thread
10
11     _names = sys.builtin_module_names
12
13 @@ -672,6 +673,7 @@ class _Environ(MutableMapping):
14         self.putenv = putenv
15         self.unsetenv = unsetenv
16         self._data = data
17     + self._lock = _thread.allocate_lock()
18
19     def __getitem__(self, key):
20         try:
21 @@ -697,7 +699,9 @@ class _Environ(MutableMapping):
22         raise KeyError(key) from None
23
24     def __iter__(self):
25     -     for key in self._data:
26     +     with self._lock:
27     +         keys = list(self._data)
28     +         for key in keys:
29                 yield self.decodekey(key)
30
31     def __len__(self):
```

# Solução Final

4 ■■■■■ Lib/os.py View 🖥 ▼

⚙	@@ -697,7 +697,9 @@ def __delitem__(self, key):		
697	697		raise KeyError(key) from None
698	698		
699	699		def __iter__(self):
700	-		for key in self._data:
	700	+	# list() from dict object is an atomic operation
	701	+	keys = list(self._data)
	702	+	for key in keys:
701	703		yield self.decodekey(key)
702	704		
703	705		def __len__(self):
⚙			

# Github Pull Request



**<https://github.com/python/cpython/pull/2409>**

**olist**

# A Recompensa?

**Ter um Python  
ainda melhor**

# E entrar pra história! 😊

The screenshot shows the GitHub interface for the `python/cpython` repository. The breadcrumb navigation path is `python / cpython / Misc / ACKS`, with `Misc / ACKS` circled in red. Below the breadcrumb, a commit by `jreese` is visible. The file `ACKS` is 1825 lines (1820 sloc) and 26.2 KB. The content of the file is a list of contributors, with line 1130, `Osvaldo Santana Neto`, highlighted in yellow.

```
1 Acknowledgements
2 -----
3
4 This list is not complete and not in any useful order, but I would
5 like to thank everybody who contributed in any way, with code, hints,
6 bug reports, ideas, moral support, endorsement, or even complaints...
7 Without you, I would've stopped working on Python long ago!
8
9 --Guido
10
11 PS: In the standard Python distribution, this file is encoded in UTF-8
12 and the list is in rough alphabetical order by last names.
13
14 Aahz
15 Michael Abbott
...
1126
1127 Tony Nelson
1128 Trent Nelson
1129 Andrew Nester
1130 Osvaldo Santana Neto
1131 Chad Netzer
1132 Max Neunhöffer
1133 Anthon van der Neut
1134 George Neville-Neil
```

<https://github.com/python/cpython/blob/master/Misc/ACKS#L1130>

olist