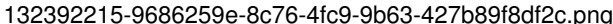


## Ruby - Bug #18154

### String#initialize leaks memory for STR\_NOFREE strings

09/07/2021 06:43 PM - peterzhu2118 (Peter Zhu)

<b>Status:</b>	Closed	
<b>Priority:</b>	Normal	
<b>Assignee:</b>		
<b>Target version:</b>		
<b>ruby -v:</b>		
<b>Backport:</b>		2.6: REQUIRED, 2.7: DONE, 3.0: DONE
<b>Description</b>		
<b>GitHub PR:</b> <a href="https://github.com/ruby/ruby/pull/4814">https://github.com/ruby/ruby/pull/4814</a>		
<p>There is a memory leak in calling the constructor on a string that is marked STR_NOFREE (e.g. a string created from a C string literal). The script below reproduces the memory leak. This is reproducible on all maintained Rubies (2.6.8, 2.7.4, 3.0.2, master) on Ubuntu 20.04.</p> <p>We create a string marked STR_NOFREE with 0.to_s. to_s for Fixnum has a <a href="#">special optimization</a> for the value 0 (it directly converts it to a C string literal). When we call String#initialize with a capacity it creates a buffer using malloc but does not unset the STR_NOFREE flag. This causes the buffer to be permanently leaked.</p> <pre>100.times do   1000.times do     # 0.to_s is a special case that creates a string from a C string literal.     # https://github.com/ruby/ruby/blob/26153667f91f0c883f6af6b61fac2c0df5312b45/numeric.c#L3393     # C string literals are always marked STR_NOFREE.     str = 0.to_s     # Call String#initialize again to create a buffer with a capacity of 10000     # characters.     str.send(:initialize, capacity: 10000)   end end  # Output the Resident Set Size (memory usage, in KB) of the current Ruby process. puts `ps -o rss= -p #{\$\$}` end</pre> <p>We can see the leak through the following graph of the Resident Set Size (RSS) comparing the branch vs. master (at commit 26153667f91f0c883f6af6b61fac2c0df5312b45).</p> 		

#### Associated revisions

**Revision 5d815542815fe8b939239750bba7f8f0b79c97d6 - 09/08/2021 02:20 PM - peterzhu2118 (Peter Zhu)**

[Bug #18154] Fix memory leak in String#initialize

String#initialize can leak memory when called on a string that is marked with STR\_NOFREE because it does not unset the STR\_NOFREE flag.

**Revision 5d815542815fe8b939239750bba7f8f0b79c97d6 - 09/08/2021 02:20 PM - peterzhu2118 (Peter Zhu)**

[Bug #18154] Fix memory leak in String#initialize

String#initialize can leak memory when called on a string that is marked with STR\_NOFREE because it does not unset the STR\_NOFREE flag.

**Revision 5d815542 - 09/08/2021 02:20 PM - peterzhu2118 (Peter Zhu)**

[Bug #18154] Fix memory leak in String#initialize

String#initialize can leak memory when called on a string that is marked with STR\_NOFREE because it does not unset the STR\_NOFREE flag.

**Revision 650af7d29d98de6a3c2631e31edc6f6be435ece89 - 09/11/2021 05:00 AM - nagachika (Tomoyuki Chikanaga)**

merge revision(s) 5d815542815fe8b939239750bba7f8f0b79c97d6: [Backport #18154]

[Bug #18154] Fix memory leak in String#initialize

String#initialize can leak memory when called on a string that is marked with STR\_NOFREE because it does not unset the STR\_NOFREE flag.

---

```
string.c          | 2 +-
test/ruby/test_string.rb | 10 ++++++++
2 files changed, 11 insertions(+), 1 deletion(-)
```

**Revision 650af7d29d98de6a3c2631e31edc6fbe435ece89 - 09/11/2021 05:00 AM - nagachika (Tomoyuki Chikanaga)**

merge revision(s) 5d815542815fe8b939239750bba7f8f0b79c97d6: [Backport #18154]

[Bug #18154] Fix memory leak in String#initialize

String#initialize can leak memory when called on a string that is marked with STR\_NOFREE because it does not unset the STR\_NOFREE flag.

---

```
string.c          | 2 +-
test/ruby/test_string.rb | 10 ++++++++
2 files changed, 11 insertions(+), 1 deletion(-)
```

**Revision 650af7d2 - 09/11/2021 05:00 AM - nagachika (Tomoyuki Chikanaga)**

merge revision(s) 5d815542815fe8b939239750bba7f8f0b79c97d6: [Backport #18154]

[Bug #18154] Fix memory leak in String#initialize

String#initialize can leak memory when called on a string that is marked with STR\_NOFREE because it does not unset the STR\_NOFREE flag.

---

```
string.c          | 2 +-
test/ruby/test_string.rb | 10 ++++++++
2 files changed, 11 insertions(+), 1 deletion(-)
```

**Revision d55426f800546cbc3b333ae7ab98c1893f710612 - 11/24/2021 10:31 AM - U.Nakamura**

merge revision(s) 5d815542815fe8b939239750bba7f8f0b79c97d6: [Backport #18154]

[Bug #18154] Fix memory leak in String#initialize

String#initialize can leak memory when called on a string that is marked with STR\_NOFREE because it does not unset the STR\_NOFREE flag.

---

```
string.c          | 2 +-
test/ruby/test_string.rb | 10 ++++++++
2 files changed, 11 insertions(+), 1 deletion(-)
```

**Revision d55426f800546cbc3b333ae7ab98c1893f710612 - 11/24/2021 10:31 AM - U.Nakamura**

merge revision(s) 5d815542815fe8b939239750bba7f8f0b79c97d6: [Backport #18154]

[Bug #18154] Fix memory leak in String#initialize

String#initialize can leak memory when called on a string that is marked with STR\_NOFREE because it does not unset the STR\_NOFREE flag.

---

```
string.c          | 2 +-
test/ruby/test_string.rb | 10 ++++++++
2 files changed, 11 insertions(+), 1 deletion(-)
```

**Revision d55426f8 - 11/24/2021 10:31 AM - U.Nakamura**

merge revision(s) 5d815542815fe8b939239750bba7f8f0b79c97d6: [Backport #18154]

[Bug #18154] Fix memory leak in String#initialize

String#initialize can leak memory when called on a string that is marked with STR\_NOFREE because it does not unset the STR\_NOFREE flag.

---

```
string.c          | 2 +-
test/ruby/test_string.rb | 10 ++++++++
2 files changed, 11 insertions(+), 1 deletion(-)
```

## History

---

### #1 - 09/07/2021 06:43 PM - peterzhu2118 (Peter Zhu)

- Backport changed from 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: UNKNOWN to 2.6: REQUIRED, 2.7: REQUIRED, 3.0: REQUIRED

### #2 - 09/07/2021 08:23 PM - Eregon (Benoit Daloze)

Should it be allowed to even call #initialize on a already-initialized String?  
I would think not, for any class.

Doesn't change this is worth fixing though.

### #3 - 09/07/2021 08:43 PM - peterzhu2118 (Peter Zhu)

Indeed, nobody should ever call #initialize on any object more than once. However, making it illegal for calling #initialize multiple times will likely be a breaking change as it's probably a feature used out in the wild.

### #4 - 09/08/2021 02:21 PM - peterzhu2118 (Peter Zhu)

- Status changed from Open to Closed

Applied in changeset [git|5d815542815fe8b939239750bba7f8f0b79c97d6](#).

---

[Bug [#18154](#)] Fix memory leak in String#initialize

String#initialize can leak memory when called on a string that is marked with STR\_NOFREE because it does not unset the STR\_NOFREE flag.

### #5 - 09/11/2021 05:19 AM - nagachika (Tomoyuki Chikanaga)

- Backport changed from 2.6: REQUIRED, 2.7: REQUIRED, 3.0: REQUIRED to 2.6: REQUIRED, 2.7: REQUIRED, 3.0: DONE

ruby\_3\_0 650af7d29d98de6a3c2631e31edc6f8e435ece89 merged revision(s) 5d815542815fe8b939239750bba7f8f0b79c97d6.

### #6 - 11/24/2021 10:31 AM - usa (Usaku NAKAMURA)

- Backport changed from 2.6: REQUIRED, 2.7: REQUIRED, 3.0: DONE to 2.6: REQUIRED, 2.7: DONE, 3.0: DONE

ruby\_2\_7 d55426f800546cbc3b333ae7ab98c1893f710612 merged revision(s) 5d815542815fe8b939239750bba7f8f0b79c97d6.