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.

We create a string marked STR_NOFREE with 0.to_s. to_s for Fixnum has a special optimization 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.

```
100.times do
  1000.times do
    str = 0.to_s
    str.send(:initialize, capacity: 10000)
  end
end
```

We can see the leak through the following graph of the Resident Set Size (RSS) comparing the branch vs. master (at commit 26153667f91f0c883f6af6b61fac2c0df5312b45).

Associated revisions

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 650af7d2 - 09/11/2021 05:00 AM - nagachika (Tomoyuki Chikanaga)
merge revision(s) 5d815542815fe8b939239751bba7f80b79c97d6: [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(-)
Backport changed from 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: UNKNOWN to 2.6: REQUIRED, 2.7: REQUIRED, 3.0: REQUIRED

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.

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.

Applied in changeset git|5d815542815fe8b939239750bba7f8f0b79c97d6.

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.

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 650af7d29d98de6a3c2631e31edc6fbe435ece89 merged revision(s) 5d815542815fe8b939239750bba7f8f0b79c97d6.