

Ruby - Bug #13021

`Zlib.gunzip` modifies argument String

12/10/2016 02:00 AM - znz (Kazuhiro NISHIYAMA)

Status:	Closed	
Priority:	Normal	
Assignee:	naruse (Yui NARUSE)	
Target version:		
ruby -v:	ruby 2.4.0dev (2016-12-10 trunk 57036) [x86_64-linux]	Backport: 2.1: DONTNEED, 2.2: DONTNEED, 2.3: DONTNEED
Description % irb -r irb/completion --simple-prompt >> RUBY_DESCRIPTION => "ruby 2.4.0dev (2016-12-10 trunk 57036) [x86_64-linux]" >> require 'zlib' => true >> gz=Zlib.gzip('test') => "\x1F\x8B\b\x00\xBF`KX\x00\x03+I-.\x01\x00\xf~\x7F\xD8\x04\x00\x00\x00" >> Zlib.gunzip(gz) => "test" >> gz => "+I-.\x01\x00\xf~\x7F\xD8\x04\x00\x00\x00"		
Related issues: Related to Ruby - Feature #13020: Zlib.gzip and Zlib.gunzip		
		Closed

Associated revisions

Revision f46a1377464d080ccbf2b943d0fe7df18500e6a - 12/10/2016 08:26 AM - nobu (Nobuyoshi Nakada)

zlib.c: replace with substring

- ext/zlib/zlib.c (zstream_discard_input): replace with unread portion substring, not modifying the input buffer directly. [ruby-core:78567] [Bug #13021]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@57042 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision f46a1377 - 12/10/2016 08:26 AM - nobu (Nobuyoshi Nakada)

zlib.c: replace with substring

- ext/zlib/zlib.c (zstream_discard_input): replace with unread portion substring, not modifying the input buffer directly. [ruby-core:78567] [Bug #13021]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@57042 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 12/10/2016 02:00 AM - znz (Kazuhiro NISHIYAMA)

- Related to Feature #13020: Zlib.gzip and Zlib.gunzip added

#2 - 12/10/2016 03:48 AM - nobu (Nobuyoshi Nakada)

```
diff --git i/ext/zlib/zlib.c w/ext/zlib/zlib.c
index 78860132df..0ce14198cc 100644
--- i/ext/zlib/zlib.c
+++ w/ext/zlib/zlib.c
@@ -879,9 +879,8 @@ zstream_discard_input(struct zstream *z, long len)
     z->input = Qnil;
     }
     else {
- memmove(RSTRING_PTR(z->input), RSTRING_PTR(z->input) + len,
- RSTRING_LEN(z->input) - len);
- rb_str_resize(z->input, RSTRING_LEN(z->input) - len);
+ z->input = rb_str_substr(z->input, len,
```

```

+   RSTRING_LEN(z->input) - len);
+ }
+ }

@@ -2406,9 +2405,7 @@ gzfile_read_raw_ensure(struct gzfile *gz, long size)
    VALUE str;

    if (gz->io == Qundef) { /* Zlib.gunzip */
-   if (NIL_P(gz->z.input))
-     rb_bug("unexpected condition: both gz->io and gz->z.input are nil");
-   if (RSTRING_LEN(gz->z.input) < size)
+   if (NIL_P(gz->z.input) || RSTRING_LEN(gz->z.input) < size)
      rb_raise(cGzError, "unexpected end of string");
    }
    while (NIL_P(gz->z.input) || RSTRING_LEN(gz->z.input) < size) {
diff --git i/test/zlib/test_zlib.rb w/test/zlib/test_zlib.rb
index 60fb84dd0d..7f9ec981fd 100644
--- i/test/zlib/test_zlib.rb
+++ w/test/zlib/test_zlib.rb
@@ -1128,25 +1128,25 @@
    end

    def test_gzip
-     actual = Zlib.gzip("foo")
+     actual = Zlib.gzip("foo".freeze)
      actual[4, 4] = "\x00\x00\x00\x00" # replace mtime
      actual[9] = "\xff" # replace OS
      expected = %w[1f8b080000000000ff4bcbcf07002165738c03000000].pack("H*")
      assert_equal expected, actual

-     actual = Zlib.gzip("foo", 0)
+     actual = Zlib.gzip("foo".freeze, 0)
      actual[4, 4] = "\x00\x00\x00\x00" # replace mtime
      actual[9] = "\xff" # replace OS
      expected = %w[1f8b080000000000ff010300fcff666f6f2165738c03000000].pack("H*")
      assert_equal expected, actual

-     actual = Zlib.gzip("foo", 9)
+     actual = Zlib.gzip("foo".freeze, 9)
      actual[4, 4] = "\x00\x00\x00\x00" # replace mtime
      actual[9] = "\xff" # replace OS
      expected = %w[1f8b08000000000002ff4bcbcf07002165738c03000000].pack("H*")
      assert_equal expected, actual

-     actual = Zlib.gzip("foo", 9, Zlib::FILTERED)
+     actual = Zlib.gzip("foo".freeze, 9, Zlib::FILTERED)
      actual[4, 4] = "\x00\x00\x00\x00" # replace mtime
      actual[9] = "\xff" # replace OS
      expected = %w[1f8b08000000000002ff4bcbcf07002165738c03000000].pack("H*")
@@ -1155,7 +1155,7 @@
    end

    def test_gunzip
      src = %w[1f8b08000000000000034bcbcf07002165738c03000000].pack("H*")
-     assert_equal 'foo', Zlib.gunzip(src)
+     assert_equal 'foo', Zlib.gunzip(src.freeze)

      src = %w[1f8b08000000000000034bcbcf07002165738c03000001].pack("H*")
      assert_raise(Zlib::GzipFile::LengthError){ Zlib.gunzip(src) }

```

#3 - 12/10/2016 08:26 AM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Closed

Applied in changeset r57042.

zlib.c: replace with substring

- ext/zlib/zlib.c (zstream_discard_input): replace with unread portion substring, not modifying the input buffer directly.
[\[ruby-core:78567\]](#) [Bug #13021]