

Ruby - Bug #12536

SIGABRT during GC on Debian GNU/Linux 8 (x86_64) due to insufficient string buffer allocation related with term fill

06/30/2016 09:35 AM - ngoto (Naohisa Goto)

Status:Closed Priority:Normal Assignee: Target version: ruby -v:	Backport:2.1: UNKNOWN, 2.2: DONE, 2.3: DONE
Description x86_64 Debian GNU/Linux 8.4 make test-all [2025/16989] REXMLTests::EncodingTester#test_parse_utf16 = 0.18 s [2026/16989] REXMLTests::EncodingTester#test_parse_utf16_with_utf8_default_intern = 0.18 s [2027/16989] REXMLTests::EncodingTester#test_ticket_89 = 0.00 s uncommon.mk:607: recipe for target 'yes-test-all' failed make: *** [yes-test-all] Aborted SIGABRT \$ ruby -r rexml/document -e 'File.open("test/rexml/data/utf16.xml") { f REXML::Document.new(f) }; GC.start' GDBGCglibcmallocSIGABRT \$ gdb ./ruby GNU gdb (Debian 7.7.1+dfsg-5) 7.7.1 (gdb) run -r rexml/document -e 'File.open("test/rexml/data/utf16.xml") { f REXML::Document.new(f) }; GC.start' Starting program: /XXXXX/ruby -r rexml/document -e 'File.open("test/rexml/data/utf16.xml") { f REXML::Document.new(f) }; GC.start' [Thread debugging using libthread_db enabled] Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1". [New Thread 0x2aaaaaad4700 (LWP 20695)] Program received signal SIGABRT, Aborted. 0x00002aaaab8df067 in __GI_raise (sig=sig@entry=6) at ../nptl/sysdeps/unix/sysv/linux/raise.c:56 56 ../nptl/sysdeps/unix/sysv/linux/raise.c: SIGABRT. (gdb) where #0 0x00002aaaab8df067 in __GI_raise (sig=sig@entry=6) at ../nptl/sysdeps/unix/sysv/linux/raise.c:56 #1 0x00002aaaab8e0448 in __GI_abort () at abort.c:89 #2 0x00002aaaab9229c0 in malloc_printerr (action=<optimized out>, str=0x2aaaab13310 "malloc_check_get_size: memory corruption", ptr=<optimized out>) at malloc.c:5000 #3 0x00002aaaab926d07 in malloc_check_get_size (p=<optimized out>) at hooks.c:114 #4 musable (mem=<optimized out>) at malloc.c:4565 #5 __malloc_usable_size (m=<optimized out>) at malloc.c:4581 #6 0x00005555555594a5a in objspace_malloc_size (objspace=0x5555555a52a90, ptr=0x2aaaaac6b010, hint=0) at gc.c:7637 #7 0x00005555555594f7e in objspace_xfree (objspace=0x5555555a52a90, ptr=0x2aaaaac6b010, old_size=0) at gc.c:7857 #8 0x0000555555559530e in ruby_sized_xfree (x=0x2aaaaac6b010, size=0) at gc.c:7952 #9 0x0000555555559532d in ruby_xfree (x=0x2aaaaac6b010) at gc.c:7959 #10 0x0000555555556739ad in rb_str_free (str=93824997851520) at string.c:1220	


```

==1118== by 0x28BD87: rb_call0 (vm_eval.c:344)
==1118== by 0x28C8E3: rb_call (vm_eval.c:630)
==1118== by 0x28D1D9: rb_funcallv (vm_eval.c:856)
==1118== by 0x12FD35: rb_obj_call_init (eval.c:1319)
==1118== by 0x19543F: rb_class_new_instance (object.c:1853)
==1118== by 0x27BED7: call_cfunc_m1 (vm_insnhelper.c:1462)
==1118== Address 0x94f94f7 is 0 bytes after a block of size 207,463 alloc'd
==1118== at 0x4C28C20: malloc (vg_replace_malloc.c:296)
==1118== by 0x148CB8: objspace_xmalloc0 (gc.c:7782)
==1118== by 0x148E25: objspace_xmalloc2 (gc.c:7810)
==1118== by 0x149077: ruby_xmalloc2 (gc.c:7887)
==1118== by 0x225B49: str_new0 (string.c:699)
==1118== by 0x225D0C: str_new (string.c:716)
==1118== by 0x225D3B: rb_str_new (string.c:722)
==1118== by 0x22920A: rb_str_plus (string.c:1738)
==1118== by 0x2831B7: vm_exec_core (insns.def:1404)
==1118== by 0x293831: vm_exec (vm.c:1653)
==1118== by 0x28B764: vm_call0_body (vm_eval.c:182)
==1118== by 0x28B319: vm_call0 (vm_eval.c:61)
==1118==
==1118==
==1118== HEAP SUMMARY:
==1118==   in use at exit: 4,716,758 bytes in 37,139 blocks
==1118==   total heap usage: 254,757 allocs, 217,618 frees, 10,809,880,914 bytes allocated
==1118==
==1118== LEAK SUMMARY:
==1118==   definitely lost: 863,638 bytes in 6,579 blocks
==1118==   indirectly lost: 1,338,400 bytes in 12,858 blocks
==1118==   possibly lost: 2,095,303 bytes in 10,822 blocks
==1118==   still reachable: 419,417 bytes in 6,880 blocks
==1118==   suppressed: 0 bytes in 0 blocks
==1118== Rerun with --leak-check=full to see details of leaked memory
==1118==
==1118== For counts of detected and suppressed errors, rerun with: -v
==1118== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)

```

```

--- string.c      (revision 55539)
+++ string.c      (working copy)
@@ -1730,16 +1730,18 @@
     rb_encoding *enc;
     char *ptr1, *ptr2, *ptr3;
     long len1, len2;
+
+    int termilen;

     StringValue(str2);
     enc = rb_enc_check_str(str1, str2);
     RSTRING_GETMEM(str1, ptr1, len1);
     RSTRING_GETMEM(str2, ptr2, len2);
-    str3 = rb_str_new(0, len1+len2);
+    termilen = rb_enc_mbminlen(enc);
+    str3 = str_new0(rb_cString, 0, len1+len2, termilen);
     ptr3 = RSTRING_PTR(str3);
     memcpy(ptr3, ptr1, len1);
     memcpy(ptr3+len1, ptr2, len2);
-    TERM_FILL(&ptr3[len1+len2], rb_enc_mbminlen(enc));
+    TERM_FILL(&ptr3[len1+len2], termilen);

     FL_SET_RAW(str3, OBJ_TAINTED_RAW(str1) | OBJ_TAINTED_RAW(str2));
     ENCODING_CODERANGE_SET(str3, rb_enc_to_index(enc),

```

UTB-16 UTB-32

```
[13718/16989] TestRegexp#test_source_unescaped = 0.00 s
[13719/16989] TestRegexp#test_to_suncommon.mk:607: recipe for target 'yes-test-all' failed
make: *** [yes-test-all] Aborted

TestRegexp#test_to_s assertp

$ ./ruby -e 'str = "abcd\u3042"; [:UTF_16BE, :UTF_16LE, :UTF_32BE, :UTF_32LE].each do |es|; enc = Encoding.const_get(es); rs = Regexp.new(str.encode(enc)).to_s; p("(?-mix:abcd\u3042)".encode(enc) == rs); p(enc == rs.encoding); end'

string.c TERM_FILL TERM_LEN termrlen
valgrind

rb_str_new() rb_str_buf_new() capa+1
#define TERM_LEN_MAX 4
ChangeLog NEWS struct RString
RSTRING_EMBED_LEN_MAX term_len RSTRING_EMBED_LEN_MAX (1)
RSTRING_EMBED_LEN_MAX "+1-term_len"
+1
```

Associated revisions

Revision f2ee22371b15411e818b23199161d1658ba08622 - 06/30/2016 10:20 AM - ngoto (Naohisa Goto)

- string.c: Fix memory corruptions when using UTF-16/32 strings.
[Bug #12536] [ruby-dev:49699]
- string.c (TERM_LEN_MAX): Macro for the longest TERM_FILL length, the same as largest value of rb_enc_mbminlen(enc) among encodings.
- string.c (str_new, rb_str_buf_new, str_shared_replace): Allocate +TERM_LEN_MAX bytes instead of +1. This change may increase memory usage.
- string.c (rb_str_new_with_class): Use TERM_LEN of the "obj".
- string.c (rb_str_plus, rb_str_justify): Use str_new0 which is aware of termrlen.
- string.c (str_shared_replace): Copy +termrlen bytes instead of +1.
- string.c (rb_str_times): termrlen should not be included in capa.
- string.c (RESIZE_CAPA_TERM): When using RSTRING_EMBED_LEN_MAX, termrlen should be counted with it because embedded strings are also processed by TERM_FILL.
- string.c (rb_str_capacity, str_shared_replace, str_buf_cat): ditto.
- string.c (rb_str_drop_bytes, rb_str_setbyte, str_byte_substr): ditto.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55547 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision f2ee2237 - 06/30/2016 10:20 AM - ngoto (Naohisa Goto)

- string.c: Fix memory corruptions when using UTF-16/32 strings.
[Bug #12536] [ruby-dev:49699]
- string.c (TERM_LEN_MAX): Macro for the longest TERM_FILL length, the same as largest value of rb_enc_mbminlen(enc) among encodings.
- string.c (str_new, rb_str_buf_new, str_shared_replace): Allocate

+TERM_LEN_MAX bytes instead of +1. This change may increase memory usage.

- string.c (rb_str_new_with_class): Use TERM_LEN of the "obj".
- string.c (rb_str_plus, rb_str_justify): Use str_new0 which is aware of termilen.
- string.c (str_shared_replace): Copy +termilen bytes instead of +1.
- string.c (rb_str_times): termilen should not be included in capa.
- string.c (RESIZE_CAPA_TERM): When using RSTRING_EMBED_LEN_MAX, termilen should be counted with it because embedded strings are also processed by TERM_FILL.
- string.c (rb_str_capacity, str_shared_replace, str_buf_cat): ditto.
- string.c (rb_str_drop_bytes, rb_str_setbyte, str_byte_substr): ditto.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55547 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 6734a0c3d9cf46d3c7e2b0368fd42ab40e74a5c8 - 07/01/2016 03:58 AM - ngoto (Naohisa Goto)

string.c: Add parentheses to avoid C source code ambiguity. [Bug #12536]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55551 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 6734a0c3 - 07/01/2016 03:58 AM - ngoto (Naohisa Goto)

string.c: Add parentheses to avoid C source code ambiguity. [Bug #12536]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55551 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 10e28726a149d7d01e0c3dedb9694ec312cd8a9d - 07/01/2016 04:50 AM - ngoto (Naohisa Goto)

- string.c (rb_str_subseq, str_substr): When RSTRING_EMBED_LEN_MAX is used, TERM_LEN(str) should be considered with it because embedded strings are also processed by TERM_FILL. Additional fix for [Bug #12536] [ruby-dev:49699].

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55552 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 10e28726 - 07/01/2016 04:50 AM - ngoto (Naohisa Goto)

- string.c (rb_str_subseq, str_substr): When RSTRING_EMBED_LEN_MAX is used, TERM_LEN(str) should be considered with it because embedded strings are also processed by TERM_FILL. Additional fix for [Bug #12536] [ruby-dev:49699].

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55552 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision a92a537bf4ada05a763ad3e43c9ebef0038643d8 - 07/01/2016 11:24 AM - ngoto (Naohisa Goto)

- string.c: Specify termilen as far as possible. Additional fix for [Bug #12536] [ruby-dev:49699].
- string.c (rb_usascii_str_new, rb_utf8_str_new): Specify termilen which is apparently 1 for the encodings.
- string.c (str_new0_cstr): New static function to create a String object from a C string with specifying termilen.
- string.c (rb_usascii_str_new_cstr, rb_utf8_str_new_cstr): Specify termilen by using new str_new0_cstr().
- string.c (str_new_static): Specify termilen from the given encoding when creating a new String object is needed.

- `string.c (rb_tainted_str_new_with_enc)`: New function to create a tainted String object with the given encoding. This means that the termien is correctly specified. Curretly static function. The function name might be renamed to `rb_tainted_enc_str_new` or `rb_enc_tainted_str_new`.
- `string.c (rb_external_str_new_with_enc)`: Use encoding by using the above `rb_tainted_str_new_with_enc()`.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55555 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision a92a537b - 07/01/2016 11:24 AM - ngoto (Naohisa Goto)

- `string.c`: Specify termien as far as possible. Additional fix for [Bug #12536] [ruby-dev:49699].
- `string.c (rb_usascii_str_new, rb_utf8_str_new)`: Specify termien which is apparently 1 for the encodings.
- `string.c (str_new0_cstr)`: New static function to create a String object from a C string with specifying termien.
- `string.c (rb_usascii_str_new_cstr, rb_utf8_str_new_cstr)`: Specify termien by using `str_new0_cstr()`.
- `string.c (str_new_static)`: Specify termien from the given encoding when creating a new String object is needed.
- `string.c (rb_tainted_str_new_with_enc)`: New function to create a tainted String object with the given encoding. This means that the termien is correctly specified. Curretly static function. The function name might be renamed to `rb_tainted_enc_str_new` or `rb_enc_tainted_str_new`.
- `string.c (rb_external_str_new_with_enc)`: Use encoding by using the above `rb_tainted_str_new_with_enc()`.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55555 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 61f2ee0d90e017e387c537bba6322075fbf4884c - 07/01/2016 05:32 PM - ngoto (Naohisa Goto)

- `string.c (str_fill_term)`: When termien increases, re-allocation of memory for termien should always be needed. In this fix, if possible, decrease capa instead of realloc. [Bug #12536] [ruby-dev:49699]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55557 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 61f2ee0d - 07/01/2016 05:32 PM - ngoto (Naohisa Goto)

- `string.c (str_fill_term)`: When termien increases, re-allocation of memory for termien should always be needed. In this fix, if possible, decrease capa instead of realloc. [Bug #12536] [ruby-dev:49699]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55557 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 3418a277d8359ac10ba0ccc047b9189d8f5f6fe3 - 07/01/2016 06:11 PM - ngoto (Naohisa Goto)

- `string.c`: Partially reverts r55547 and r55555. ChangeLog about the reverted changes are also deleted in this file. [Bug #12536] [ruby-dev:49699] [ruby-dev:49702]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55559 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 3418a277 - 07/01/2016 06:11 PM - ngoto (Naohisa Goto)

- string.c: Partially reverts r55547 and r55555.
ChangeLog about the reverted changes are also deleted in this file.
[Bug #12536] [ruby-dev:49699] [ruby-dev:49702]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55559 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 5eff15d1bdda4efcdd8aa59c91c804e345394a1c - 07/05/2016 10:45 AM - ngoto (Naohisa Goto)

- string.c (rb_str_change_terminator_length): New function to change termilen and resize heap for the terminator. This is split from rb_str_fill_terminator (str_fill_term) because filling terminator and changing terminator length are different things. [Bug #12536]
- internal.h: declaration for rb_str_change_terminator_length.
- string.c (str_fill_term): Simplify only to zero-fill the terminator. For non-shared strings, it assumes that (capa + termilen) bytes of heap is allocated. This partially reverts r55557.
- encoding.c (rb_enc_associate_index): rb_str_change_terminator_length is used, and it should be called whenever the termilen is changed.
- string.c (str_capacity): New static function to return capacity of a string with the given termilen, because the termilen may sometimes be different from TERM_LEN(str) especially during changing termilen or filling terminator with specific termilen.
- string.c (rb_str_capacity): Use str_capacity.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55575 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 5eff15d1 - 07/05/2016 10:45 AM - ngoto (Naohisa Goto)

- string.c (rb_str_change_terminator_length): New function to change termilen and resize heap for the terminator. This is split from rb_str_fill_terminator (str_fill_term) because filling terminator and changing terminator length are different things. [Bug #12536]
- internal.h: declaration for rb_str_change_terminator_length.
- string.c (str_fill_term): Simplify only to zero-fill the terminator. For non-shared strings, it assumes that (capa + termilen) bytes of heap is allocated. This partially reverts r55557.
- encoding.c (rb_enc_associate_index): rb_str_change_terminator_length is used, and it should be called whenever the termilen is changed.
- string.c (str_capacity): New static function to return capacity of a string with the given termilen, because the termilen may sometimes be different from TERM_LEN(str) especially during changing termilen or filling terminator with specific termilen.
- string.c (rb_str_capacity): Use str_capacity.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55575 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 2bb292fccf9560b2c885b4368e5c5fc3fe2a2bda - 07/15/2016 12:35 PM - ngoto (Naohisa Goto)

- string.c (str_buf_cat): Fix capa size for embed string.
Fix bug in r55547. [Bug #12536]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55691 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 2bb292fc - 07/15/2016 12:35 PM - ngoto (Naohisa Goto)

- string.c (str_buf_cat): Fix capa size for embed string.
Fix bug in r55547. [Bug #12536]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55691 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 6f122a4f10512bb60795faa0021e498800a7f1b8 - 08/22/2016 06:18 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) 55547,55551,55552,55555,55557,55559,55575,55691,55568: [Backport #12536]

```
* string.c: Fix memory corruptions when using UTF-16/32 strings.
[Bug #12536] [ruby-dev:49699]

* string.c (TERM_LEN_MAX): Macro for the longest TERM_FILL length,
the same as largest value of rb_enc_mbminlen(enc) among encodings.

* string.c (str_new, rb_str_buf_new, str_shared_replace): Allocate
+TERM_LEN_MAX bytes instead of +1. This change may increase memory
usage.

* string.c (rb_str_new_with_class): Use TERM_LEN of the "obj".

* string.c (rb_str_plus, rb_str_justify): Use str_new0 which is aware
of termilen.

* string.c (str_shared_replace): Copy +termilen bytes instead of +1.

* string.c (rb_str_times): termilen should not be included in capa.

* string.c (RESIZE_CAPA_TERM): When using RSTRING_EMBED_LEN_MAX,
termilen should be counted with it because embedded strings are
also processed by TERM_FILL.

* string.c (rb_str_capacity, str_shared_replace, str_buf_cat): ditto.

* string.c (rb_str_drop_bytes, rb_str_setbyte, str_byte_substr): ditto.

* string.c (rb_str_subseq, str_substr): When RSTRING_EMBED_LEN_MAX
is used, TERM_LEN(str) should be considered with it because
embedded strings are also processed by TERM_FILL.
Additional fix for [Bug #12536] [ruby-dev:49699].
Additional fix for [Bug #12536] [ruby-dev:49699].

* string.c (rb_usascii_str_new, rb_utf8_str_new): Specify termilen
which is apparently 1 for the encodings.

* string.c (str_new0_cstr): New static function to create a String
object from a C string with specifying termilen.

* string.c (rb_usascii_str_new_cstr, rb_utf8_str_new_cstr): Specify
termilen by using new_str_new0_cstr().

* string.c (str_new_static): Specify termilen from the given encoding
when creating a new String object is needed.

* string.c (rb_tainted_str_new_with_enc): New function to create a
tainted String object with the given encoding. This means that
the termilen is correctly specified. Curretly static function.
The function name might be renamed to rb_tainted_enc_str_new
or rb_enc_tainted_str_new.

* string.c (rb_external_str_new_with_enc): Use encoding by using the
above rb_tainted_str_new_with_enc().

* string.c (str_fill_term): When termilen increases, re-allocation
of memory for termilen should always be needed.
In this fix, if possible, decrease capa instead of realloc.
[Bug #12536] [ruby-dev:49699]

* string.c: Partially reverts r55547 and r55555.
ChangeLog about the reverted changes are also deleted in this file.
[Bug #12536] [ruby-dev:49699] [ruby-dev:49702]

* string.c (rb_str_change_terminator_length): New function to change
termilen and resize heap for the terminator. This is split from
```

rb_str_fill_terminator (str_fill_term) because filling terminator and changing terminator length are different things. [Bug #12536]

* internal.h: declaration for rb_str_change_terminator_length.

* string.c (str_fill_term): Simplify only to zero-fill the terminator. For non-shared strings, it assumes that (capa + termilen) bytes of heap is allocated. This partially reverts r55557.

* encoding.c (rb_enc_associate_index): rb_str_change_terminator_length is used, and it should be called whenever the termilen is changed.

* string.c (str_capacity): New static function to return capacity of a string with the given termilen, because the termilen may sometimes be different from TERM_LEN(str) especially during changing termilen or filling terminator with specific termilen.

* string.c (rb_str_capacity): Use str_capacity.

* string.c (str_buf_cat): Fix capa size for embed string. Fix bug in r55547. [Bug #12536]

* string.c: Specify termilen as far as possible. the termilen is correctly specified. Currently static function.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_3@55988 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 6f122a4f - 08/22/2016 06:18 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) 55547,55551,55552,55555,55557,55559,55575,55691,55568: [Backport #12536]

* string.c: Fix memory corruptions when using UTF-16/32 strings. [Bug #12536] [ruby-dev:49699]

* string.c (TERM_LEN_MAX): Macro for the longest TERM_FILL length, the same as largest value of rb_enc_mbminlen(enc) among encodings.

* string.c (str_new, rb_str_buf_new, str_shared_replace): Allocate +TERM_LEN_MAX bytes instead of +1. This change may increase memory usage.

* string.c (rb_str_new_with_class): Use TERM_LEN of the "obj".

* string.c (rb_str_plus, rb_str_justify): Use str_new0 which is aware of termilen.

* string.c (str_shared_replace): Copy +termilen bytes instead of +1.

* string.c (rb_str_times): termilen should not be included in capa.

* string.c (RESIZE_CAPA_TERM): When using RSTRING_EMBED_LEN_MAX, termilen should be counted with it because embedded strings are also processed by TERM_FILL.

* string.c (rb_str_capacity, str_shared_replace, str_buf_cat): ditto.

* string.c (rb_str_drop_bytes, rb_str_setbyte, str_byte_substr): ditto.

* string.c (rb_str_subseq, str_substr): When RSTRING_EMBED_LEN_MAX is used, TERM_LEN(str) should be considered with it because embedded strings are also processed by TERM_FILL. Additional fix for [Bug #12536] [ruby-dev:49699]. Additional fix for [Bug #12536] [ruby-dev:49699].

* string.c (rb_usascii_str_new, rb_utf8_str_new): Specify termilen which is apparently 1 for the encodings.

* string.c (str_new0_cstr): New static function to create a String object from a C string with specifying termilen.

* string.c (rb_usascii_str_new_cstr, rb_utf8_str_new_cstr): Specify termilen by using new str_new0_cstr().

* string.c (str_new_static): Specify termilen from the given encoding when creating a new String object is needed.

- * string.c (rb_tainted_str_new_with_enc): New function to create a tainted String object with the given encoding. This means that the termilen is correctly specified. Currently static function. The function name might be renamed to rb_tainted_enc_str_new or rb_enc_tainted_str_new.
- * string.c (rb_external_str_new_with_enc): Use encoding by using the above rb_tainted_str_new_with_enc().
- * string.c (str_fill_term): When termilen increases, re-allocation of memory for termilen should always be needed. In this fix, if possible, decrease capa instead of realloc. [Bug #12536] [ruby-dev:49699]
- * string.c: Partially reverts r55547 and r55555. ChangeLog about the reverted changes are also deleted in this file. [Bug #12536] [ruby-dev:49699] [ruby-dev:49702]
- * string.c (rb_str_change_terminator_length): New function to change termilen and resize heap for the terminator. This is split from rb_str_fill_terminator (str_fill_term) because filling terminator and changing terminator length are different things. [Bug #12536]
- * internal.h: declaration for rb_str_change_terminator_length.
- * string.c (str_fill_term): Simplify only to zero-fill the terminator. For non-shared strings, it assumes that (capa + termilen) bytes of heap is allocated. This partially reverts r55557.
- * encoding.c (rb_enc_associate_index): rb_str_change_terminator_length is used, and it should be called whenever the termilen is changed.
- * string.c (str_capacity): New static function to return capacity of a string with the given termilen, because the termilen may sometimes be different from TERM_LEN(str) especially during changing termilen or filling terminator with specific termilen.
- * string.c (rb_str_capacity): Use str_capacity.
- * string.c (str_buf_cat): Fix capa size for embed string. Fix bug in r55547. [Bug #12536]
- * string.c: Specify termilen as far as possible. the termilen is correctly specified. Currently static function.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_3@55988 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 63e4f2120ce8b80b5782dbf13a691925aea7f3e2 - 09/30/2016 03:42 PM - U.Nakamura

merge revision(s) 55547,55551,55552,55555,55557,55559,55568,55575,55691: [Backport #12536]

- * string.c: Fix memory corruptions when using UTF-16/32 strings. [Bug #12536] [ruby-dev:49699]
- * string.c (TERM_LEN_MAX): Macro for the longest TERM_FILL length, the same as largest value of rb_enc_mbminlen(enc) among encodings.
- * string.c (str_new, rb_str_buf_new, str_shared_replace): Allocate +TERM_LEN_MAX bytes instead of +1. This change may increase memory usage.
- * string.c (rb_str_new_with_class): Use TERM_LEN of the "obj".
- * string.c (rb_str_plus, rb_str_justify): Use str_new0 which is aware of termilen.
- * string.c (str_shared_replace): Copy +termilen bytes instead of +1.
- * string.c (rb_str_times): termilen should not be included in capa.
- * string.c (RESIZE_CAPA_TERM): When using RSTRING_EMBED_LEN_MAX, termilen should be counted with it because embedded strings are also processed by TERM_FILL.

- * string.c (rb_str_capacity, str_shared_replace, str_buf_cat): ditto.
- * string.c (rb_str_drop_bytes, rb_str_setbyte, str_byte_substr): ditto.
- * string.c (rb_str_subseq, str_substr): When RSTRING_EMBED_LEN_MAX is used, TERM_LEN(str) should be considered with it because embedded strings are also processed by TERM_FILL. Additional fix for [Bug #12536] [ruby-dev:49699]. Additional fix for [Bug #12536] [ruby-dev:49699].
- * string.c (rb_usascii_str_new, rb_utf8_str_new): Specify termilen which is apparently 1 for the encodings.
- * string.c (str_new0_cstr): New static function to create a String object from a C string with specifying termilen.
- * string.c (rb_usascii_str_new_cstr, rb_utf8_str_new_cstr): Specify termilen by using new str_new0_cstr().
- * string.c (str_new_static): Specify termilen from the given encoding when creating a new String object is needed.
- * string.c (rb_tainted_str_new_with_enc): New function to create a tainted String object with the given encoding. This means that the termilen is correctly specified. Currently static function. The function name might be renamed to rb_tainted_enc_str_new or rb_enc_tainted_str_new.
- * string.c (rb_external_str_new_with_enc): Use encoding by using the above rb_tainted_str_new_with_enc().
- * string.c (str_fill_term): When termilen increases, re-allocation of memory for termilen should always be needed. In this fix, if possible, decrease capa instead of realloc. [Bug #12536] [ruby-dev:49699]
- * string.c: Partially reverts r55547 and r55555. ChangeLog about the reverted changes are also deleted in this file. [Bug #12536] [ruby-dev:49699] [ruby-dev:49702]
- * string.c: Specify termilen as far as possible. the termilen is correctly specified. Currently static function.
- * string.c (rb_str_change_terminator_length): New function to change termilen and resize heap for the terminator. This is split from rb_str_fill_terminator (str_fill_term) because filling terminator and changing terminator length are different things. [Bug #12536]
- * internal.h: declaration for rb_str_change_terminator_length.
- * string.c (str_fill_term): Simplify only to zero-fill the terminator. For non-shared strings, it assumes that (capa + termilen) bytes of heap is allocated. This partially reverts r55557.
- * encoding.c (rb_enc_associate_index): rb_str_change_terminator_length is used, and it should be called whenever the termilen is changed.
- * string.c (str_capacity): New static function to return capacity of a string with the given termilen, because the termilen may sometimes be different from TERM_LEN(str) especially during changing termilen or filling terminator with specific termilen.
- * string.c (rb_str_capacity): Use str_capacity.
- * string.c (str_buf_cat): Fix capa size for embed string. Fix bug in r55547. [Bug #12536]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_2@56301 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 63e4f212 - 09/30/2016 03:42 PM - U.Nakamura

merge revision(s) 55547,55551,55552,55555,55557,55559,55568,55575,55691: [Backport #12536]

- * string.c: Fix memory corruptions when using UTF-16/32 strings. [Bug #12536] [ruby-dev:49699]

- * string.c (TERM_LEN_MAX): Macro for the longest TERM_FILL length, the same as largest value of rb_enc_mbminlen(enc) among encodings.
- * string.c (str_new, rb_str_buf_new, str_shared_replace): Allocate +TERM_LEN_MAX bytes instead of +1. This change may increase memory usage.
- * string.c (rb_str_new_with_class): Use TERM_LEN of the "obj".
- * string.c (rb_str_plus, rb_str_justify): Use str_new0 which is aware of termrlen.
- * string.c (str_shared_replace): Copy +termrlen bytes instead of +1.
- * string.c (rb_str_times): termrlen should not be included in capa.
- * string.c (RESIZE_CAPA_TERM): When using RSTRING_EMBED_LEN_MAX, termrlen should be counted with it because embedded strings are also processed by TERM_FILL.
- * string.c (rb_str_capacity, str_shared_replace, str_buf_cat): ditto.
- * string.c (rb_str_drop_bytes, rb_str_setbyte, str_byte_substr): ditto.
- * string.c (rb_str_subseq, str_substr): When RSTRING_EMBED_LEN_MAX is used, TERM_LEN(str) should be considered with it because embedded strings are also processed by TERM_FILL. Additional fix for [Bug #12536] [ruby-dev:49699]. Additional fix for [Bug #12536] [ruby-dev:49699].
- * string.c (rb_usascii_str_new, rb_utf8_str_new): Specify termrlen which is apparently 1 for the encodings.
- * string.c (str_new0_cstr): New static function to create a String object from a C string with specifying termrlen.
- * string.c (rb_usascii_str_new_cstr, rb_utf8_str_new_cstr): Specify termrlen by using new str_new0_cstr().
- * string.c (str_new_static): Specify termrlen from the given encoding when creating a new String object is needed.
- * string.c (rb_tainted_str_new_with_enc): New function to create a tainted String object with the given encoding. This means that the termrlen is correctly specified. Currently static function. The function name might be renamed to rb_tainted_enc_str_new or rb_enc_tainted_str_new.
- * string.c (rb_external_str_new_with_enc): Use encoding by using the above rb_tainted_str_new_with_enc().
- * string.c (str_fill_term): When termrlen increases, re-allocation of memory for termrlen should always be needed. In this fix, if possible, decrease capa instead of realloc. [Bug #12536] [ruby-dev:49699]
- * string.c: Partially reverts r55547 and r55555. ChangeLog about the reverted changes are also deleted in this file. [Bug #12536] [ruby-dev:49699] [ruby-dev:49702]
- * string.c: Specify termrlen as far as possible. the termrlen is correctly specified. Currently static function.
- * string.c (rb_str_change_terminator_length): New function to change termrlen and resize heap for the terminator. This is split from rb_str_fill_terminator (str_fill_term) because filling terminator and changing terminator length are different things. [Bug #12536]
- * internal.h: declaration for rb_str_change_terminator_length.
- * string.c (str_fill_term): Simplify only to zero-fill the terminator. For non-shared strings, it assumes that (capa + termrlen) bytes of heap is allocated. This partially reverts r55557.

```
* encoding.c (rb_enc_associate_index): rb_str_change_terminator_length
is used, and it should be called whenever the termilen is changed.

* string.c (str_capacity): New static function to return capacity
of a string with the given termilen, because the termilen may
sometimes be different from TERM_LEN(str) especially during
changing termilen or filling terminator with specific termilen.

* string.c (rb_str_capacity): Use str_capacity.

* string.c (str_buf_cat): Fix capa size for embed string.
Fix bug in r55547. [Bug #12536]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_2@56301 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 06/30/2016 10:20 AM - ngoto (Naohisa Goto)

- Status changed from Open to Closed

Applied in changeset r55547.

- string.c: Fix memory corruptions when using UTF-16/32 strings.
[Bug #12536] [ruby-dev:49699]
- string.c (TERM_LEN_MAX): Macro for the longest TERM_FILL length,
the same as largest value of rb_enc_mbmminlen(enc) among encodings.
- string.c (str_new, rb_str_buf_new, str_shared_replace): Allocate
+TERM_LEN_MAX bytes instead of +1. This change may increase memory
usage.
- string.c (rb_str_new_with_class): Use TERM_LEN of the "obj".
- string.c (rb_str_plus, rb_str_justify): Use str_new0 which is aware
of termilen.
- string.c (str_shared_replace): Copy +termilen bytes instead of +1.
- string.c (rb_str_times): termilen should not be included in capa.
- string.c (RESIZE_CAPA_TERM): When using RSTRING_EMBED_LEN_MAX,
termilen should be counted with it because embedded strings are
also processed by TERM_FILL.
- string.c (rb_str_capacity, str_shared_replace, str_buf_cat): ditto.
- string.c (rb_str_drop_bytes, rb_str_setbyte, str_byte_substr): ditto.

#2 - 07/01/2016 04:34 PM - ngoto (Naohisa Goto)

```
rb_str_new() rb_str_buf_new() capa+1
```

```
string.c
rb_enc_associate_index(), rb_enc_associate(), rb_enc_copy()
API
```

```
rb_str_fill_terminator()
string.c str_fill_term()
```

```
termilen termilen
```

```
termilen
→rb_str_fill_terminator()capa
→rb_str_catcaparealloc
→capaTERM_FILL
→termileno termilen
→
```

000000

```
000000000000000000000000000000000000 revert 0000
```

EMBED

#3 - 07/05/2016 01:03 PM - ngoto (Naohisa Goto)

```
rb str fill terminator()
```

[illegible]

#4 - 07/05/2016 01:04 PM - ngoto (Naohisa Goto)

- Backport changed from 2.1: UNKNOWN, 2.2: UNKNOWN, 2.3: UNKNOWN to 2.1: UNKNOWN, 2.2: REQUIRED, 2.3: REQUIRED

#5 - 08/22/2016 06:18 PM - nagachika (Tomoyuki Chikanaga)

- Backport changed from 2.1: UNKNOWN, 2.2: REQUIRED, 2.3: REQUIRED to 2.1: UNKNOWN, 2.2: REQUIRED, 2.3: DONE

ruby_2_3 r55988 merged revision(s) 55547,55551,55552,55555,55557,55559,55575,55691,55568.

#6 - 09/30/2016 03:42 PM - usa (Usaku NAKAMURA)

- Backport changed from 2.1: UNKNOWN, 2.2: REQUIRED, 2.3: DONE to 2.1: UNKNOWN, 2.2: DONE, 2.3: DONE

ruby 2 2 r56301 merged revision(s) 55547,55551,55552,55555,55557,55559,55568,55575,55691.

Files

20160630-string.c-term-len.patch	5.66 KB	06/30/2016	ngoto (Naohisa Goto)
----------------------------------	---------	------------	----------------------