Ruby master - Bug #8337

Test failure and memory leak with OpenSSL::BN

04/27/2013 03:29 PM - h.shirosaki (Hiroshi Shirosaki)

Status:	Closed		
Priority:	Normal		
Assignee:	h.shirosaki (Hiroshi Shirosaki)		
Target version:	2.1.0		
ruby -v:	ruby 2.1.0dev (2013-04-27 trunk 40468) [x86_64-darwin12.3.0]	**Backport:**	1.9.3: UNKNOWN, 2.0.0: UNKNOWN

Description

I noticed test failure of test_to_bn.

http://ci.rubyinstaller.org/job/ruby-trunk-x64-test-all/1137/console
1) Failure:
test_to_bn(OpenSSL::TestBN) [C:/Users/Worker/Jenkins/workspace/ruby-trunk-x64-build/test/openssl/test_bn.rb:36]:
<#OpenSSL::BN:0x00000017d20188> expected but was
<#OpenSSL::BN:0x00000017d18460>.

sizeof(VALUE) of the following code looks wrong because sizeof(VALUE) > sizeof(long) on x64 Windows.

https://github.com/ruby/ruby/blob/8b29525dadeaba1ba6dc2a9ea5e590aa9d1d825a/ext/openssl/ossl_bn.c#L130

And ALLOC_N() may need xfree().
I've confirmed memory leak with the following script.

$ ruby -rOpenSSL -e 'loop { 1.to_bn }'

Here is a patch to fix above issues.

diff --git a/ext/openssl/ossl_bn.c b/ext/openssl/ossl_bn.c
index 4e9734e..3d8e095 100644
--- a/ext/openssl/ossl_bn.c
+++ b/ext/openssl/ossl_bn.c
@@ -127,15 +127,17 @@ ossl_bn_initialize(int argc, VALUE *argv, VALUE self)
long n = FIX2LONG(str);
unsigned long un = labs(n);

- for (i = sizeof(VALUE) - 1; 0 <= i; i--) {

- for (i = sizeof(long) - 1; 0 <= i; i--) {
  bin[i] = un&0xff;
  un >>= 8;
  }

  GetBN(self, bn);
  if (!BN_bin2bn(bin, sizeof(long), bn)) {

- [redacted]
  ossl_raise(eBNError, NULL);

  }

- xfree(bin);
  if (n < 0) BN_set_negative(bn, 1);
  return self;
  }
  @@ -154,8 +156,10 @@ ossl_bn_initialize(int argc, VALUE *argv, VALUE self)

  GetBN(self, bn);
  if (!BN_bin2bn(bin, (int)sizeof(BDIGIT)*RBIGNUM_LENINT(str), bn)) {

- [redacted]
  ossl_raise(eBNError, NULL);

  }

- xfree(bin);
  if (!RBIGNUM_SIGN(str)) BN_set_negative(bn, 1);
  return self;
  }

Associated revisions

Revision be4aa330 - 04/28/2013 01:20 PM - shirosaki

ossl_bn.c: fix ossl_bn_initialize bug with integer

- ext/openssl/ossl_bn.c (ossl_bn_initialize): fix buffer overflow on x64 Windows and memory leak when initializing with integer. [ruby-core:54615] [Bug #8337]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@40513 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 40513 - 04/28/2013 01:20 PM - shirosaki

ossl_bn.c: fix ossl_bn_initialize bug with integer

- ext/openssl/ossl_bn.c (ossl_bn_initialize): fix buffer overflow on x64 Windows and memory leak when initializing with integer. [ruby-core:54615] [Bug #8337]

Revision 40513 - 04/28/2013 01:20 PM - shirosaki

ossl_bn.c: fix ossl_bn_initialize bug with integer

- ext/openssl/ossl_bn.c (ossl_bn_initialize): fix buffer overflow on x64 Windows and memory leak when initializing with integer. [ruby-core:54615] [Bug #8337]

Revision 40513 - 04/28/2013 01:20 PM - shirosaki

ossl_bn.c: fix ossl_bn_initialize bug with integer

- ext/openssl/ossl_bn.c (ossl_bn_initialize): fix buffer overflow on x64 Windows and memory leak when initializing with integer. [ruby-core:54615] [Bug #8337]

Revision 40513 - 04/28/2013 01:20 PM - shirosaki

ossl_bn.c: fix ossl_bn_initialize bug with integer

- ext/openssl/ossl_bn.c (ossl_bn_initialize): fix buffer overflow on x64 Windows and memory leak when initializing with integer. [ruby-core:54615] [Bug #8337]

Revision 40513 - 04/28/2013 01:20 PM - shirosaki

ossl_bn.c: fix ossl_bn_initialize bug with integer

- ext/openssl/ossl_bn.c (ossl_bn_initialize): fix buffer overflow on x64 Windows and memory leak when initializing with integer. [ruby-core:54615] [Bug #8337]

Revision 40513 - 04/28/2013 01:20 PM - shirosaki

ossl_bn.c: fix ossl_bn_initialize bug with integer

- ext/openssl/ossl_bn.c (ossl_bn_initialize): fix buffer overflow on x64 Windows and memory leak when initializing with integer. [ruby-core:54615] [Bug #8337]

History

#1 - 04/27/2013 05:23 PM - naruse (Yui NARUSE)

Sure, commit please.

#2 - 04/27/2013 05:24 PM - naruse (Yui NARUSE)

naruse (Yui NARUSE) wrote:

> Sure, commit please.

Additionally, ALLOC_N for Fixnum can be simply ALLOCA_N.

#3 - 04/28/2013 10:20 PM - Anonymous

- % Done changed from 0 to 100

- Status changed from Open to Closed

This issue was solved with changeset r40513.
Hiroshi, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

---

ossl_bn.c: fix ossl_bn_initialize bug with integer

- ext/openssl/ossl_bn.c (ossl_bn_initialize): fix buffer overflow on x64 Windows and memory leak when initializing with integer. [ruby-core:54615] [Bug #8337]

#4 - 04/29/2013 12:27 AM - nagachika (Tomoyuki Chikanaga)

- Status changed from Closed to Assigned

- Assignee set to h.shirosaki (Hiroshi Shirosaki)

Hello,

I think ALLOCA_N() uses alloca() to allocate memory from machine stack and xfree() is not necessary.

#5 - 04/29/2013 01:11 AM - h.shirosaki (Hiroshi Shirosaki)

- Status changed from Assigned to Closed

nagachika (Tomoyuki Chikanaga) wrote:

> Hello,
>
> I think ALLOCA_N() uses alloca() to allocate memory from machine stack and xfree() is not necessary.

I use ALLOCA_N only for Fixnum and ALLOC_N is used for Bignum that needs xfree().

https://github.com/ruby/ruby/blob/be4aa330374d42cdead52a94144be189b5054e67/ext/openssl/ossl_bn.c#L145

#6 - 04/29/2013 07:03 AM - nagachika (Tomoyuki Chikanaga)

oh...
I'm sorry for bother you.