Ruby trunk - Bug #4320

Bus Error in digest/sha2 on sparc

01/25/2011 10:31 PM - Meik (Meik Nienaber)

Status:	Closed		
Priority:	Normal		
Assignee:	nahi (Hiroshi Nakamura)		
Target version:	1.9.3		
ruby -v:	ruby 1.9.2p136 (2010-12-25 revision 30365) [sparc-solaris2.10]	**Backport:**	

Description

=begin
Most likely this is caused due to misaligned memory. Any comment is greatly appreciated.

This bug can reproduce at Ruby 1.8, too.

```
ruby -e "require 'digest/sha2'; d= Digest::SHA256.new; ['a' * 97, 'a' * 97].each {|i| d.update(i)}; p d"
-e:1: [BUG] Bus Error
ruby 1.9.2p136 (2010-12-25 revision 30365) [sparc-solaris2.10]

-- control frame ----------
c:0007 p:---- s:0019 b:0019 l:000018 d:000018 CFUNC  :update
c:0006 p:0014 s:0015 b:0015 l:0015ac d:000014 BLOCK  -e:1
c:0005 p:---- s:0012 b:0012 l:000011 d:000011 FINISH
c:0004 p:---- s:0010 b:0010 l:000009 d:000009 CFUNC  :each
c:0003 p:0054 s:0007 b:0007 l:0015ac d:000ed0 EVAL   -e:1
c:0002 p:---- s:0004 b:0004 l:000003 d:000003 FINISH
c:0001 p:0000 s:0002 b:0002 l:0015ac d:0015ac TOP
```

=end

Associated revisions

Revision e56f2abe - 07/15/2011 03:03 AM - nahi (Hiroshi Nakamura)

- ext/digest/sha2/sha2.c (SHA256_Update, SHA512_Update): avoid Bus Error caused by unalignment access on Sparc-Solaris (and possibly on other similar environment.) This patch just do memcpy always instead of checking architecture. I see no perf drop on my 64bit env. For more details, see #4320.
- test/digest/test_digest.rb: add test for unalignment access.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@32546 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 32546 - 07/15/2011 03:03 AM - nahi (Hiroshi Nakamura)

- ext/digest/sha2/sha2.c (SHA256_Update, SHA512_Update): avoid Bus Error caused by unalignment access on Sparc-Solaris (and possibly on other similar environment.) This patch just do memcpy always instead of checking architecture. I see no perf drop on my 64bit env. For more details, see #4320.
- test/digest/test_digest.rb: add test for unalignment access.

Revision 32546 - 07/15/2011 03:03 AM - nahi (Hiroshi Nakamura)

- ext/digest/sha2/sha2.c (SHA256_Update, SHA512_Update): avoid Bus Error caused by unalignment access on Sparc-Solaris (and possibly on other similar environment.) This patch just do memcpy always instead of checking architecture. I see no perf drop on my 64bit env. For more details, see #4320.
- test/digest/test_digest.rb: add test for unalignment access.

Revision 32546 - 07/15/2011 03:03 AM - nahi (Hiroshi Nakamura)

- ext/digest/sha2/sha2.c (SHA256_Update, SHA512_Update): avoid Bus Error caused by unalignment access on Sparc-Solaris (and possibly on other similar environment.) This patch just do memcpy always instead of checking architecture. I see no perf drop on my 64bit env. For more details, see #4320.
- test/digest/test_digest.rb: add test for unalignment access.

Revision 32546 - 07/15/2011 03:03 AM - nahi (Hiroshi Nakamura)

- ext/digest/sha2/sha2.c (SHA256_Update, SHA512_Update): avoid Bus Error caused by unalignment access on Sparc-Solaris (and possibly on other similar environment.) This patch just do memcpy always instead of checking architecture. I see no perf drop on my 64bit env. For more details, see #4320.
- test/digest/test_digest.rb: add test for unalignment access.

Revision 32546 - 07/15/2011 03:03 AM - nahi (Hiroshi Nakamura)

- ext/digest/sha2/sha2.c (SHA256_Update, SHA512_Update): avoid Bus Error caused by unalignment access on Sparc-Solaris (and possibly on other similar environment.) This patch just do memcpy always instead of checking architecture. I see no perf drop on my 64bit env. For more details, see #4320.
- test/digest/test_digest.rb: add test for unalignment access.

Revision 32546 - 07/15/2011 03:03 AM - nahi (Hiroshi Nakamura)

- ext/digest/sha2/sha2.c (SHA256_Update, SHA512_Update): avoid Bus Error caused by unalignment access on Sparc-Solaris (and possibly on other similar environment.) This patch just do memcpy always instead of checking architecture. I see no perf drop on my 64bit env. For more details, see #4320.
- test/digest/test_digest.rb: add test for unalignment access.

Revision b03e3cc5 - 07/15/2011 03:05 AM - nahi (Hiroshi Nakamura)

- backport r32546 from trunk.
- ext/digest/sha2/sha2.c (SHA256_Update, SHA512_Update): avoid Bus Error caused by unalignment access on Sparc-Solaris (and possibly on other similar environment.) This patch just do memcpy always instead of checking architecture. I see no perf drop on my 64bit env. For more details, see #4320.
- test/digest/test_digest.rb: add test for unalignment access.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_1_9_3@32547 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 01/28/2011 11:57 AM - usa (Usaku NAKAMURA)

- Status changed from Open to Assigned

- Assignee set to knu (Akinori MUSHA)

=begin

=end

#2 - 04/19/2011 08:52 PM - slink (Nils Goroll)

=begin
This indeed is an alignment issue, SHA256_Update calls SHA256_Transform with possibly unaligned data, but the latter needs its data argument be aligned on platforms which do not support unaligned word access. The same bug exists for SHA384 and SHA512

I have a fix ready, currently testing it.

here's a failing test case for SHA512, which will also hit for SHA384 (which uses the same code).

```
ruby -e "require 'digest/sha2'; d= Digest::SHA512.new; ['a' * 57, 'b' * 199].each {|i| d.update(i)}; p d"
-e:1: [BUG] Bus Error
ruby 1.9.2p136 (2010-12-25 revision 30365) [sparc-solaris2.10]


-- control frame ----------
c:0007 p:---- s:0019 b:0019 l:000018 d:000018 CFUNC  :update
c:0006 p:0014 s:0015 b:0015 l:001504 d:000014 BLOCK  -e:1
c:0005 p:---- s:0012 b:0012 l:000011 d:000011 FINISH
c:0004 p:---- s:0010 b:0010 l:000009 d:000009 CFUNC  :each
c:0003 p:0054 s:0007 b:0007 l:001504 d:000db0 EVAL   -e:1
c:0002 p:---- s:0004 b:0004 l:000003 d:000003 FINISH
c:0001 p:0000 s:0002 b:0002 l:001504 d:001504 TOP
```

```
-- Ruby level backtrace information ----------------------------------------
-e:1:in <main>'
-e:1:ineach'
-e:1:in block in <main>'
-e:1:inupdate'

[NOTE]
You may have encountered a bug in the Ruby interpreter or extension libraries.
Bug reports are welcome.
For details: http://www.ruby-lang.org/bugreport.html

Abort (core dumped)
```

=end

#3 - 04/19/2011 11:49 PM - slink (Nils Goroll)

- File ruby_issue_4320__digest_sha2_alignment.patch added

=begin
I'll attach a proposed fix:

- make ((|context->buffer|)) an array of the type being expected by (({SHAXXX_Transform})) (rather than a byte array), so our compiler will align it,

if necessary
- remove now unneeded casts when passing the buffer to (({SHAXXX_Transform}))
- use a cast-to-uint8 version of the buffer for byte access

the actual fix:

- for platforms which are not known to accept unaligned access to words (conditions taken from ((%regint.h%))), use existing buffering code in (({SHAXXX_Update})) to align data by copying

I am not too happy about the (({PLAT_NEED_ALIGNED_WORD_ACCESS})), it appears to me that checking architecture alignment requirements should be done in ((%autoconf%)) or similar. Also, the simplistic (({ALIGNOF()})) macro will not return the minimal alignment requirement, but rather the alignment the compiler has chosen for a struct (which may be more than minimal).

At any rate, the patch does the job.

=== Test

I've done basic regression testing by

- comparing the output of the following commands between x86 with unpatched 1.8.7-p175 i386-solaris2.11 and patched 1.9.2-p180:

ruby -e "require 'digest/sha2'; 1.upto(2*90+20) { |i| 1.upto(2*90+20) { |j| d= Digest::SHA256.new; p i.to_s+' '+j.to_s; ['a' * i, 'b' * j].each {|c| d.update(c)}; p d}}" >/tmp/sha_256.out &
ruby -e "require 'digest/sha2'; 1.upto(2*90+20) { |i| 1.upto(2*90+20) { |j| d= Digest::SHA512.new; p i.to_s+' '+j.to_s; ['a' * i, 'b' * j].each {|c| d.update(c)}; p d}}" >/tmp/sha_512.out &

- checksumming the contents of Solaris 10 SPARC ((%/usr/bin%)) with the vesions given above usind:

find . -type f | ruby -e "require 'digest/sha2'; ARGF.each_line { |fname| fname = fname.chomp; begin; p fname+':
'+Digest::SHA256.file(fname).hexdigest; rescue; end; } " >/tmp/ruby_hash_n

So I presume SHA2 is working OK.

((%make check%)) returns

8063 tests, 1870484 assertions, 21 failures, 35 errors, 2 skips

but none of the failures/errors is related to digest.
=end

#4 - 06/26/2011 04:12 PM - nahi (Hiroshi Nakamura)

- Assignee changed from knu (Akinori MUSHA) to nahi (Hiroshi Nakamura)

- Target version changed from 1.9.2 to 1.9.3

#5 - 07/14/2011 06:47 PM - nahi (Hiroshi Nakamura)

- File sha2.c.diff added

Nils, thanks for the patch, and sorry for late reply.

Since sha2.c has an upstream version http://www.aarongifford.com/computers/sha.html (which does not support sparc-solaris I think), I want to minimize the patch. Here's my version based on yours. Can you try it? It passes 'make test-all TESTS=digest' on my SunOS 5.8 on sparc.

#6 - 07/15/2011 01:05 AM - nahi (Hiroshi Nakamura)

This looks much simpler. I'll check this patch tomorrow.

Index: ext/digest/sha2/sha2.c

--- ext/digest/sha2/sha2.c (revision 32536)
+++ ext/digest/sha2/sha2.c (working copy)
@@ -559,7 +559,8 @@
}
while (len >= SHA256_BLOCK_LENGTH) {
/* Process as many complete blocks as we can */

- SHA256_Transform(context, (sha2_word32*)data);
- MEMCPY_BCOPY(context->buffer, data, SHA256_BLOCK_LENGTH);
- SHA256_Transform(context, (sha2_word32*)context->buffer);   context->bitcount += SHA256_BLOCK_LENGTH << 3;   len -= SHA256_BLOCK_LENGTH;   data += SHA256_BLOCK_LENGTH; @@ -880,7 +881,8 @@ } while (len >= SHA512_BLOCK_LENGTH) {   /* Process as many complete blocks as we can */
- SHA512_Transform(context, (sha2_word64*)data);

- MEMCPY_BCOPY(context->buffer, data, SHA512_BLOCK_LENGTH);
- SHA512_Transform(context, (sha2_word64*)context->buffer); ADDINC128(context->bitcount, SHA512_BLOCK_LENGTH << 3); len -= SHA512_BLOCK_LENGTH; data += SHA512_BLOCK_LENGTH;

#7 - 07/15/2011 01:18 PM - nahi (Hiroshi Nakamura)

- Status changed from Assigned to Closed

I applied the last patch to trunk at r32546 and ruby_1_9_3 at r32547.

Nils, sorry for not merging your patch directly. The reason I didn't apply yours is almost from the size of the patch. Would you please report the issue to the upstream (http://www.aarongifford.com/computers/sha.html) for other users, and of course for us?

Files

ruby_issue_4320__digest_sha2_alignment.patch	10.5 KB	04/19/2011	slink (Nils Goroll)
sha2.c.diff	1.54 KB	07/14/2011	nahi (Hiroshi Nakamura)