Backport187 - Bug #1131

String#unpack("V") does not work correctly is linux on s390x

02/08/2009 10:32 PM - ittayd (Ittay Dror)

Status:	Closed	
Priority:	Normal	
Assignee:	nobu (Nobuyoshi Nakada)	
Target version:		
ruby -v:	ruby 1.8.7 (2008-08-11 patchlevel 72) [s390x-linux]	

Description

```
=begin
irb(main):008:0> [17138429].pack("V").unpack("V")
=> [736089920598179842]

ruby --version:
ruby 1.8.7 (2008-08-11 patchlevel 72) [s390x-linux]

uname -a:
Linux etpgloi 2.6.5-7.244-s390x #1 SMP Mon Dec 12 18:32:25 UTC 2005 s390x s390x s390x GNU/Linux

cat /etc/issue:
Welcome to SUSE LINUX Enterprise Server 9 (s390x) - Kernel \r (\l).

ruby was compiled from sources.
=end
```

History

#1 - 02/09/2009 12:05 AM - ittayd (Ittay Dror)

```
=begin
the following workaround fixes the issue:
if  [1].pack('V').unpack('V')[0] != 1
class String
alias_method :broken_unpack, :unpack
def unpack(spec)
broken_unpack(spec).each_with_index.map do |x,i|
if spec[i] == ?V
x = x / 4294967296 # 256 ^ 4
end
x
end
end
end
end
=end
```

#2 - 02/09/2009 12:21 AM - nobu (Nobuyoshi Nakada)

=begin
Hi,

At Sun, 8 Feb 2009 22:31:10 +0900,
Ittay Dror wrote in [ruby-core:21937]:

```
    ruby -v: ruby 1.8.7 (2008-08-11 patchlevel 72) [s390x-linux]

    irb(main):008:0> [17138429].pack("V").unpack("V")
    => [736089920598179842]

    ruby --version:
    ruby 1.8.7 (2008-08-11 patchlevel 72) [s390x-linux]
```

System Z is 64bit big endian? Could you show the config.h?

--
Nobu Nakada

=end

#3 - 04/16/2009 03:41 PM - ittayd (Ittay Dror)

- File config.h added

=begin

=end

#4 - 10/28/2009 03:02 AM - naruse (Yui NARUSE)

- Status changed from Open to Assigned

- Assignee set to nobu (Nobuyoshi Nakada)

=begin

=end

#5 - 02/15/2010 08:05 PM - darix (Marcus Rückert)

=begin
any update on this bug?
=end

#6 - 02/16/2010 10:15 AM - matz (Yukihiro Matsumoto)

=begin
Hi,

In message "Re: [ruby-core:28176] [Bug #1131] String#unpack("V") does not work correctly is linux on s390x"
on Mon, 15 Feb 2010 20:05:55 +0900, Marcus Rückert redmine@ruby-lang.org writes:

|any update on this bug?

Since no core developer has s390x machine, we need information asked
in [ruby-core:21940] to solve the problem. Or patch from the owner is
welcome.

```
                                matz.
```

=end

#7 - 02/16/2010 07:51 PM - darix (Marcus Rückert)

- File s390x-config.h added

=begin
another s390x config.h
=end

#8 - 02/16/2010 07:52 PM - darix (Marcus Rückert)

=begin
On 2010-02-16 10:15:28 +0900, Yukihiro Matsumoto wrote:

> In message "Re: [ruby-core:28176] [Bug #1131] String#unpack("V") does not work correctly is linux on s390x"
> on Mon, 15 Feb 2010 20:05:55 +0900, Marcus Rückert redmine@ruby-lang.org writes:
>
> |any update on this bug?
>
> Since no core developer has s390x machine, we need information asked
> in [ruby-core:21940] to solve the problem. Or patch from the owner is
> welcome.

04/16/2009 03:41 PM - Ittay Dror
File config.h added

it just seems that event didnt trigger an email.

i now also attached our config.h.

If you need any testing for patches let me know.

```
darix
```

--
openSUSE - SUSE Linux is my linux
openSUSE is good for you
www.opensuse.org

=end

#9 - 05/10/2010 03:48 AM - kkaempf (Klaus Kämpf)

- File bnc603914_string_unpack.patch added

=begin
unpack("V") broken on 64bit big endian systems, ppc64 is also affected.

It is caused by the following code (pack.c, pack_unpack(), case 'V')

```
            unsigned long tmp = 0;
            memcpy(OFF32(&tmp), s, NATINT_LEN(long,4));
            s += NATINT_LEN(long,4);
            rb_ary_push(ary, ULONG2NUM(vtohl(tmp)));
```

The use of OFF32 in memcpy copies the int32 value into the correct half of the 8-byte target value (tmp).
The call to vtohl() swaps the complete 8-byte value (acting like swap64), thereby bringing the 4 bytes of the int32 into the right endianess but also into the wrong half of the 8-byte target.

The fix is to either drop of OFF32 call or to replace the vtohl with swap32. I choose the latter since swap32 is faster than vtohl.

Proposed patch attached
=end

#10 - 05/12/2010 11:12 PM - kkaempf (Klaus Kämpf)

=begin
A similar problem exists in "Array#pack"ing 'i_' values in big endian system.

[1].pack("i_")
=> "\000\000\000\000"

Again, the wrong use of OFF32() causes this bug together with the use of "long" when handling 'i' type in pack.c

I have opened a new issue for this: http://redmine.ruby-lang.org/issues/show/3297
=end

#11 - 05/12/2010 11:20 PM - akr (Akira Tanaka)

=begin
ruby 1.8.8dev doesn't use OFF32() now.
=end

#12 - 06/08/2010 05:47 PM - shyouhei (Shyouhei Urabe)

- Status changed from Assigned to Closed

- % Done changed from 0 to 100

=begin
This issue was solved with changeset r28220.
Ittay, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

=end

Files

config.h	5.12 KB	04/16/2009	ittayd (Ittay Dror)
s390x-config.h	4.89 KB	02/16/2010	darix (Marcus Rückert)

bnc603914_string_unpack.patch	485 Bytes	05/10/2010	kkaempf (Klaus Kämpf)