Ruby - Bug #2640

signal code only needs slight changes to make Ruby compile on Haiku-os

01/25/2010 10:30 AM - kallisti5 (Alexander von Gluck)

Status:	Third Party's Issue		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:	1.9.1	**Backport:**	

Description

=begin
gcc -O2 -g -Wall -Wno-parentheses -I. -I.ext/include/i586-haiku -I./include -I. -DRUBY_EXPORT   -o signal.o -c signal.c
signal.c: In function 'sigsegv':
signal.c:593: error: 'info' undeclared (first use in this function)
signal.c:593: error: (Each undeclared identifier is reported only once
signal.c:593: error: for each function it appears in.)
make: *** [signal.o] Error 1

commenting out the sigsegv handling code lets ruby 1.9 compile successfully on Haiku-os. This is not a valid solution but does show how close 1.9 is to running on the Haiku platform.
=end

History

#1 - 01/25/2010 10:51 AM - naruse (Yui NARUSE)

- Category set to build

=begin
Please try the patch in following page.
http://znz.s1.xrea.com/t/?date=20090926
=end

#2 - 01/25/2010 10:54 AM - nobu (Nobuyoshi Nakada)

=begin
Hi,

At Mon, 25 Jan 2010 10:30:30 +0900,
Alexander von Gluck wrote in [ruby-core:27768]:

> signal.c: In function 'sigsegv':
> signal.c:593: error: 'info' undeclared (first use in this function)
> signal.c:593: error: (Each undeclared identifier is reported only once
> signal.c:593: error: for each function it appears in.)
> make: *** [signal.o] Error 1

It means sigaltstack() is found but SA_SIGINFO isn't. Haiku-OS
doesn't provide the flag? Or does it need another header?

--
Nobu Nakada

=end

#3 - 01/25/2010 11:13 AM - kallisti5 (Alexander von Gluck)

=begin
@Yui

That patch definitely looks like it would resolve this issue. Since that info structure is a todo for the Haiku project the code should probably be left as-is until Haiku completes it.

Thanks!
-- Alex
=end

#4 - 01/25/2010 11:35 AM - naruse (Yui NARUSE)

- Status changed from Open to Third Party's Issue

=begin
Hmm, if so, this is not Ruby's issue.
If Haiku 1.0 still doesn't have them, please reopen this; we may apply the patch.
=end

#5 - 01/25/2010 12:16 PM - kallisti5 (Alexander von Gluck)

=begin
The patch suggested at http://znz.s1.xrea.com/t/?date=20090926 causes the bus error for me too and seems like an invalid work-around.

The overall solution for this issue is on Haiku's side and not Ruby's fault.
http://dev.haiku-os.org/ticket/2695

Until Haiku fixes this I'll just comment out the broken call which seems to work and make a miniruby that works :). Just a quick note that this patch may cause some bad mojo on a sigsegv and should not be used in production apps.

Index: signal.c

--- signal.c (revision 26395)
+++ signal.c (working copy)
@@ -34,7 +34,7 @@

define ATOMIC_DEC(var) (--(var))

#endif

-#ifdef **BEOS**
+#if defined(**BEOS**) || defined(**HAIKU**)
#undef SIGBUS
#endif

@@ -597,6 +597,7 @@
static RETSIGTYPE
sigsegv(int sig SIGINFO_ARG)
{
+/*
#ifdef USE_SIGALTSTACK
int ruby_stack_overflowed_p(const rb_thread_t *, const void *);
NORETURN(void ruby_thread_stack_overflow(rb_thread_t *th));
@@ -605,6 +606,7 @@
ruby_thread_stack_overflow(th);
}
#endif
+/
if (segv_received) {
fprintf(stderr, "SEGV received in SEGV handler\n");
exit(EXIT_FAILURE);
=end

#6 - 01/25/2010 12:41 PM - kallisti5 (Alexander von Gluck)

=begin
Looks like Nobuyoshi Nakada fixed this in r26399

logic: if no SIGSEGV info, don't USE_SIGALTSTACK

That should be generic, way to be smarter then the Operating system :)

can be closed.
=end