

Ruby - Bug #21167

Visual Studio 2022 17.13.x couldn't build ruby.exe

03/03/2025 05:30 AM - hsbt (Hiroshi SHIBATA)

Status:Third Party's Issue Priority:Normal Assignee: Target version: ruby -v:		Backport:3.1: UNKNOWN, 3.2: UNKNOWN, 3.3: UNKNOWN, 3.4: UNKNOWN
Description miniruby.exe built by Visual Studio 2022 17.13.0-2 is not working with our build process. <pre>user32.lib advapi32.lib shell32.lib ws2_32.lib iphlapi.lib gmp.lib imagehlp.lib shlwapi.lib bcrypt.lib linking miniruby.exe Creating library miniruby.lib and object miniruby.exp [BUG] heap_idx_for_size: allocation size too large (size=56u, heap_idx=891288545u) ruby 3.5.0dev (2025-03-03T02:17:14Z master a47e686cb6) [x64-mswin64_140] -- Control frame information ----- c:0001 p:---- s:0003 e:000002 DUMMY [FINISH] -- Threading information ----- Total ractor count: 0 Ruby thread count for this ractor: 0 -- C level backtrace information ----- C:\WINDOWS\SYSTEM32\ntdll.dll (ZwWaitForSingleObject+0x14) [0x00007FF9057BF854] C:\WINDOWS\System32\KERNELBASE.dll (WaitForSingleObjectEx+0xaf) [0x00007FF902CFCE1F] C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (rb_print_backtrace+0x3e) [0x00007FF63DC37E7E] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\vm_dump.c:859 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (rb_vm_bugreport+0x1ba) [0x00007FF63DC3803E] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\vm_dump.c:1170 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (rb_bug_without_die_internal+0x72) [0x00007FF63DB0255A] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\error.c:1097 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (rb_bug+0x20) [0x00007FF63DB02448] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\error.c:1117 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (newobj_of+0x1e7) [0x00007FF63DB186FB] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\gc.c:1007 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (typed_data_alloc+0xc3) [0x00007FF63DB212CF] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\gc.c:1084 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (rb_data_typed_object_zalloc+0x3b) [0x00007FF63DB19797] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\gc.c:1107 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (pin_array_list_new+0x25) [0x00007FF63DC1B5D9] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\vm.c:4387 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (Init_vm_objects+0x17) [0x00007FF63DC1665B] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\vm.c:4446 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (ruby_setup+0x3f) [0x00007FF63DB08C1B] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\eval.c:83 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (ruby_init+0x9) [0x00007FF63DB089AD] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\eval.c:98 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (wmain+0x3a) [0x00007FF63DA8103A] C:\Users\hsbt\DevDrive\github.com\ruby\ruby\main.c:51 C:\Users\hsbt\DevDrive\github.com\ruby\ruby.build\2022\miniruby.exe (__scrt_common_main_seh+0x10c) [0x00007FF63DC9DE88] D:\a_work\1\s\src\vctools\crt\vcstartup\src\startup\exe_common.inl:288 C:\WINDOWS\System32\KERNEL32.DLL (BaseThreadInitThunk+0x17) [0x00007FF90409E8D7] -- Other runtime information ----- NMAKE : fatal error U1077: '@ miniruby.exe -v' : return code '0x3' Stop.</pre> I disabled that version on our CI temporary. https://github.com/ruby/ruby/pull/12830 https://github.com/ruby/actions/pull/81		

Related issues:

Has duplicate Ruby - Bug #21213: [BUG] heap_idx_for_size: allocation size too...

Third Party's Issue

History

#1 - 03/03/2025 09:59 AM - nobu (Nobuyoshi Nakada)

I think it is a bug in VC 19.43.34808 for x64.

This is the code compiled heap_idx_for_size with VC 19.43.34808 for x64

```
mov r8d, 63      ; 0000003fH
sub r8d, DWORD PTR r$2[rsp]
lea rcx, QWORD PTR [r10+39]
mul rcx
shr rdx, 5
lea rax, QWORD PTR [rdx-1]
bsr rcx, rax
mov DWORD PTR r$2[rsp], ecx
mov ecx, 64      ; 00000040H
mov eax, ecx
cmovne eax, r8d
```

If bsr reset the zero flag, cmovne at the end copies r8d which is calculated prior to bsr, (probably to reduce use of intermediate registers). However r8d is calculated from r\$2[rsp] which should be the result of bsr and has not been set at the time of that calculation.

#2 - 03/03/2025 11:29 AM - nobu (Nobuyoshi Nakada)

Reported at <https://developercommunity.visualstudio.com/t/Use-of-the-result-of-bsr-before-that-i/10862351>

#3 - 03/06/2025 02:08 AM - hsbt (Hiroshi SHIBATA)

[@nobu \(Nobuyoshi Nakada\)](#) found miniruby.exe is working with -O2 instead of -O2s. But we faced another issue with:

```
linking shared-library x64-vcruntime140-ruby350.dll
x64-vcruntime140-ruby350.def : error LNK2001: unresolved external symbol Avx2WmemEnabledWeakValue
x64-vcruntime140-ruby350.def : error LNK2001: unresolved external symbol _isa_available_default
dummy.lib : fatal error LNK1120: 2 unresolved externals
```

<https://github.com/ruby/ruby/pull/12853>

It happened with vs2019/2015 too.

#4 - 03/07/2025 09:51 AM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Third Party's Issue

According to the Microsoft engineering team, already this bug has been fixed internally.

<https://developercommunity.visualstudio.com/t/Use-of-the-result-of-bsr-before-that-i/10862351#T-N10863092-N10864280>

#5 - 04/02/2025 05:53 AM - nobu (Nobuyoshi Nakada)

- Has duplicate Bug #21213: [BUG] heap_idx_for_size: allocation size too large (size=145u, heap_idx=146u) added

#6 - 04/09/2025 01:36 AM - hsbt (Hiroshi SHIBATA)

vs2022 17.14.0 preview 2.0 also has this issue.

#7 - 04/22/2025 08:05 AM - hsbt (Hiroshi SHIBATA)

vs2022 17.14.0 preview 3.0 resolved this issue.