

## Ruby master - Bug #7756

### clang 4.2 sees through UNINITIALIZED\_VAR macro, gives warning

01/30/2013 05:05 AM - drbrain (Eric Hodel)

<b>Status:</b>	Closed	
<b>Priority:</b>	Normal	
<b>Assignee:</b>	mame (Yusuke Endoh)	
<b>Target version:</b>	2.6	
<b>ruby -v:</b>	-	
<b>Backport:</b>		
<b>Description</b>		
=begin		
In clang 3.2, the UNINITIALIZED_VAR macro does not prevent the compiler from producing a warning.		
Apple clang 3.2:		
\$ clang -v		
Apple LLVM version 4.2 (clang-425.0.24) (based on LLVM 3.2svn)		
Target: x86_64-apple-darwin12.2.1		
Thread model: posix		
When compiling thread.c:		
compiling ../trunk/thread.c		
../trunk/thread.c:3261:34: warning: variable 'orig_read' is uninitialized when used within its own initialization [-Wuninitialized]		
rb_fdset_t UNINITIALIZED_VAR(orig_read);		
~~~~~		
../trunk/vm_core.h:119:34: note: expanded from macro 'UNINITIALIZED_VAR'		
#define UNINITIALIZED_VAR(x) x = x		
^		
../trunk/thread.c:3262:34: warning: variable 'orig_write' is uninitialized when used within its own initialization [-Wuninitialized]		
rb_fdset_t UNINITIALIZED_VAR(orig_write);		
~~~~~		
../trunk/vm_core.h:119:34: note: expanded from macro 'UNINITIALIZED_VAR'		
#define UNINITIALIZED_VAR(x) x = x		
^		
../trunk/thread.c:3263:34: warning: variable 'orig_except' is uninitialized when used within its own initialization [-Wuninitialized]		
rb_fdset_t UNINITIALIZED_VAR(orig_except);		
~~~~~		
../trunk/vm_core.h:119:34: note: expanded from macro 'UNINITIALIZED_VAR'		
#define UNINITIALIZED_VAR(x) x = x		
^		
3 warnings generated.		
=end		

#### Associated revisions

##### Revision 69d5f4ab - 03/21/2013 04:50 AM - nobu (Nobuyoshi Nakada)

vm\_core.h: suppress warnings

- vm\_core.h (UNINITIALIZED\_VAR): suppress warnings by clang 4.2. [ruby-core:51742] [Bug #7756]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@39855 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

##### Revision 39855 - 03/21/2013 04:50 AM - nobu (Nobuyoshi Nakada)

vm\_core.h: suppress warnings

- vm\_core.h (UNINITIALIZED\_VAR): suppress warnings by clang 4.2. [ruby-core:51742] [Bug #7756]

**Revision 39855 - 03/21/2013 04:50 AM - nobu (Nobuyoshi Nakada)**

vm\_core.h: suppress warnings

- vm\_core.h (UNINITIALIZED\_VAR): suppress warnings by clang 4.2. [ruby-core:51742] [Bug #7756]

**Revision 39855 - 03/21/2013 04:50 AM - nobu (Nobuyoshi Nakada)**

vm\_core.h: suppress warnings

- vm\_core.h (UNINITIALIZED\_VAR): suppress warnings by clang 4.2. [ruby-core:51742] [Bug #7756]

**Revision 39855 - 03/21/2013 04:50 AM - nobu (Nobuyoshi Nakada)**

vm\_core.h: suppress warnings

- vm\_core.h (UNINITIALIZED\_VAR): suppress warnings by clang 4.2. [ruby-core:51742] [Bug #7756]

**Revision 39855 - 03/21/2013 04:50 AM - nobu (Nobuyoshi Nakada)**

vm\_core.h: suppress warnings

- vm\_core.h (UNINITIALIZED\_VAR): suppress warnings by clang 4.2. [ruby-core:51742] [Bug #7756]

**Revision 39855 - 03/21/2013 04:50 AM - nobu (Nobuyoshi Nakada)**

vm\_core.h: suppress warnings

- vm\_core.h (UNINITIALIZED\_VAR): suppress warnings by clang 4.2. [ruby-core:51742] [Bug #7756]

**History**

---

**#1 - 01/30/2013 05:32 AM - kosaki (Motohiro KOSAKI)**

Hm, maybe we need another workaround.  
Does **attribute** (unused) works on your compiler?

likes

```
#ifdef GNUC
#define UNINITIALIZED_VAR(x) x attribute (unused) = x
#else
....
```

**#2 - 01/30/2013 08:53 AM - drbrain (Eric Hodel)**

=begin  
This patch works:

```
Index: vm_core.h
=====
--- vm_core.h (revision 38980)
+++ vm_core.h (working copy)
@@ -116,7 +116,7 @@
#end if GNUC >= 3 */

# if GNUC >= 3
-#define UNINITIALIZED_VAR(x) x = x
+#define UNINITIALIZED_VAR(x) x attribute((unused))
# else
# define UNINITIALIZED_VAR(x) x
# end if
= end
```

**#3 - 02/17/2013 01:50 PM - ko1 (Koichi Sasada)**

- Assignee set to *mame* (Yusuke Endoh)

mame-san, how about it?

**#4 - 02/17/2013 02:44 PM - mame (Yusuke Endoh)**

- Target version changed from 2.0.0 to 2.6

It emits just a warning, not an error, right?

Looks not significant.

--

Yusuke Endoh [mame@tsg.ne.jp](mailto:mame@tsg.ne.jp)

**#5 - 02/18/2013 02:31 AM - drbrain (Eric Hodel)**

It is only a warning

**#6 - 03/12/2013 09:23 AM - nobu (Nobuyoshi Nakada)**

=begin

Unfortunately, it doesn't shut up gcc-4.7.

thread.c: In function 'do\_select':

```
thread.c:3155:26: warning: 'orig_except.fdset' may be used uninitialized in this function [-Wmaybe-uninitialized]
thread.c:3352:16: note: 'orig_except.fdset' was declared here
thread.c:3229:19: warning: 'orig_except.maxfd' may be used uninitialized in this function [-Wmaybe-uninitialized]
thread.c:3352:16: note: 'orig_except.maxfd' was declared here
thread.c:3155:26: warning: 'orig_write.fdset' may be used uninitialized in this function [-Wmaybe-uninitialized]
thread.c:3351:16: note: 'orig_write.fdset' was declared here
thread.c:3229:19: warning: 'orig_write.maxfd' may be used uninitialized in this function [-Wmaybe-uninitialized]
thread.c:3351:16: note: 'orig_write.maxfd' was declared here
thread.c:3155:26: warning: 'orig_read.fdset' may be used uninitialized in this function [-Wmaybe-uninitialized]
thread.c:3350:16: note: 'orig_read.fdset' was declared here
thread.c:3229:19: warning: 'orig_read.maxfd' may be used uninitialized in this function [-Wmaybe-uninitialized]
thread.c:3350:16: note: 'orig_read.maxfd' was declared here
=end
```

**#7 - 03/12/2013 03:10 PM - naruse (Yui NARUSE)**

GCC 4.6 and clang (2.8?) supports pragma diagnostic, but its syntax is different from current UNINITIALIZED\_VAR().

```
#pragma GCC diagnostic push
#pragma GCC diagnostic ignored "-Wuninitialized"
blah blah
#pragma GCC diagnostic pop
```

<http://gcc.gnu.org/gcc-4.6/changes.html>

<http://gcc.gnu.org/onlinedocs/gcc/Diagnostic-Pragmas.html>

<http://clang.lvm.org/docs/UsersManual.html#controlling-diagnostics-via-pragmas>

**#8 - 03/20/2013 02:11 AM - naruse (Yui NARUSE)**

A macro can be following, but it still changes syntax from current UNINITIALIZED\_VAR.

```
#if GCC_VERSION_SINCE(4,6,0) || defined(clang)
```

**define SUPPRESS\_DIAGNOSTIC(arg,rest) \**

```
  _Pragma("GCC diagnostic push") \
  _Pragma(Stringize(GCC diagnostic ignored arg)) \
  rest \
  _Pragma("GCC diagnostic pop")
```

#else

**define SUPPRESS\_DIAGNOSTIC(arg,rest) rest**

#endif

**#9 - 03/21/2013 01:50 PM - nobu (Nobuyoshi Nakada)**

- Subject changed from clang 3.2 sees through UNINITIALIZED\_VAR macro, gives warning to clang 4.2 sees through UNINITIALIZED\_VAR macro, gives warning

**#10 - 03/21/2013 01:50 PM - nobu (Nobuyoshi Nakada)**

- Status changed from Open to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r39855.

Eric, thank you for reporting this issue.  
Your contribution to Ruby is greatly appreciated.  
May Ruby be with you.

---

vm\_core.h: suppress warnings

- vm\_core.h (UNINITIALIZED\_VAR): suppress warnings by clang 4.2. [ruby-core:51742] [Bug [#7756](#)]