Bug 67394 - crash due to null pointer deref in demangle_signature()

Status:	NEW	**Reported:**	2015-08-29 17:08 UTC by geeknik
Alias:	None	**Modified:**	2015-08-29 22:06 UTC (History)
Product:	gcc	**CC List:**	1 user (show)
Component:	c++ (show other bugs)		
Version:	4.9.2	**See Also:**	
		Host:	
Importance:	P3 normal	**Target:**	
Target Milestone:	---	**Build:**	
		Known to work:	
Assignee:	Not yet assigned to anyone	**Known to fail:**	6.0
		Last reconfirmed:	2015-08-29 00:00:00
URL:			
Keywords:			
Depends on:			
Blocks:			

Attachments

Add an attachment (proposed patch, testcase, etc.)

Note

You need to log in before you can comment on or make changes to this bug.

geeknik 2015-08-29 17:08:43 UTC **Description**

While fuzzing binutils/cxxfilt with AFL (http://lcamtuf.coredump.cx/afl/), I discovered a crash due to a null ptr deref in demangle_signature(). This is with GCC 4.9.2 and Debian 7 (x64).

```
./cxxfilt _Q.__0

Valgrind:
==4253== Invalid write of size 8
==4253==    at 0x7AD3A0: register_Btype (cplus-dem.c:4319)
==4253==    by 0x7AD3A0: demangle_class (cplus-dem.c:2594)
==4253==    by 0x7AD3A0: demangle_signature (cplus-dem.c:1490)
==4253==    by 0x7BB869: internal_cplus_demangle (cplus-dem.c:1203)
==4253==    by 0x7825B2: cplus_demangle (cplus-dem.c:886)
==4253==    by 0x408192: demangle_it (cxxfilt.c:62)
==4253==    by 0x407618: main (cxxfilt.c:227)
==4253==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==4253==
==4253==
==4253== Process terminating with default action of signal 11 (SIGSEGV)
==4253==  Access not within mapped region at address 0x0
==4253==    at 0x7AD3A0: register_Btype (cplus-dem.c:4319)
==4253==    by 0x7AD3A0: demangle_class (cplus-dem.c:2594)
==4253==    by 0x7AD3A0: demangle_signature (cplus-dem.c:1490)
==4253==    by 0x7BB869: internal_cplus_demangle (cplus-dem.c:1203)
==4253==    by 0x7825B2: cplus_demangle (cplus-dem.c:886)
==4253==    by 0x408192: demangle_it (cxxfilt.c:62)
```

```
==4253==    by 0x407618: main (cxxfilt.c:227)
==4253==  If you believe this happened as a result of a stack
==4253==  overflow in your program's main thread (unlikely but
==4253==  possible), you can try to increase the size of the
==4253==  main thread stack using the --main-stacksize= flag.
==4253==  The main thread stack size used in this run was 8388608.
Segmentation fault


GDB:
Program received signal SIGSEGV, Segmentation fault.
0x00000000007ad3a0 in demangle_signature ()
(gdb) bt
#0  0x00000000007ad3a0 in demangle_signature ()
#1  0x00000000007bb86a in internal_cplus_demangle ()
#2  0x00000000007825b3 in cplus_demangle ()
#3  0x0000000000408193 in demangle_it () at cxxfilt.c:62
#4  0x0000000000407619 in main () at cxxfilt.c:227
(gdb) i R
rax            0x0      0
rbx            0x0      0
rcx            0x0      0
rdx            0x7fffffffe110   140737488347408
rsi            0x7fffffffe108   140737488347400
rdi            0x0      0
rbp            0x7fffffffe108   0x7fffffffe108
rsp            0x7fffffffdfe0   0x7fffffffdfe0
r8             0xabe000 11264000
r9             0x0      0
r10            0x20     32
r11            0x1e     30
r12            0x7fffffffe110   140737488347408
r13            0x0      0
r14            0x7fffffffe180   140737488347520
r15            0x1      1
rip            0x7ad3a0 0x7ad3a0 <demangle_signature+9248>
eflags         0x10293  [ CF AF SF IF RF ]
cs             0x33     51
ss             0x2b     43
ds             0x0      0
es             0x0      0
fs             0x0      0
gs             0x0      0
```

Mikhail Maltsev 2015-08-29 22:06:34 UTC Comment 1

Reproduces on trunk (the bug is in pre-v3 demangler, cplus-dem.c, I did not fuzz it). Something like this should fix it:

```
diff --git a/libiberty/cplus-dem.c b/libiberty/cplus-dem.c
index c68b981..7ab46dd 100644
--- a/libiberty/cplus-dem.c
+++ b/libiberty/cplus-dem.c
@@ -1237,11 +1237,13 @@ squangle_mop_up (struct work_stuff *work)
     {
       free ((char *) work -> btypevec);
       work->btypevec = NULL;
+      work->bsize = 0;
     }
   if (work -> ktypevec != NULL)
     {
       free ((char *) work -> ktypevec);
       work->ktypevec = NULL;
+      work->ksize = 0;
     }
 }
```

Format For Printing - XML - Clone This Bug - Top of page