

strongSwan - Bug #2425

`./utils/utils/memory.h:99:15: error: 'uintptr_t' undeclared (first use in this function); did you mean '__intptr_t'?`

12.09.2017 07:26 - Paul Wouters

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Tobias Brunner	Estimated time:	0.00 hour
Category:	libstrongswan	Resolution:	Fixed
Target version:	5.6.1		
Affected version:	5.6.0		

Description

This happens on the fedora rawhide gcc

```
/bin/sh ../../libtool --tag=CC --mode=compile gcc -DHAVE_CONFIG_H -I. -I../.. -I../..../src/libstrongswan -DIPSEC_DIR=\"/usr/libexec/strongswan\" -DIPSEC_LIB_DIR=\"/usr/lib64/strongswan\" -DPLUGIN_INDIR=\"/usr/lib64/strongswan/plugins\" -DSTRONGSWAN_CONF=\"/etc/strongswan/strongswan.conf\" -O2 -g -pipe -Wall -Werror=format-security -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -specs=/usr/lib/rpm/redhat/redhat-hardened-cc1 -m64 -mtune=generic -Wno-pointer-sign -include /builddir/build/BUILD/strongswan-5.6.0/config.h -c -o crypto/transform.lo crypto/transform.c
libtool: compile: gcc -DHAVE_CONFIG_H -I. -I../.. -I../..../src/libstrongswan -DIPSEC_DIR=\"/usr/libexec/strongswan\" -DIPSEC_LIB_DIR=\"/usr/lib64/strongswan\" -DPLUGIN_INDIR=\"/usr/lib64/strongswan/plugins\" -DSTRONGSWAN_CONF=\"/etc/strongswan/strongswan.conf\" -O2 -g -pipe -Wall -Werror=format-security -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -specs=/usr/lib/rpm/redhat/redhat-hardened-cc1 -m64 -mtune=generic -Wno-pointer-sign -include /builddir/build/BUILD/strongswan-5.6.0/config.h -c library.c -fPIC -DPIC -o .libs/library.o
In file included from utils/utils.h:55:0,
                 from library.h:101,
                 from library.c:17:
utils/utils/memory.h: In function 'memwipe_inline':
utils/utils/memory.h:99:15: error: 'uintptr_t' undeclared (first use in this function); did you mean '__intptr_t'?
   for (i = 0; (uintptr_t)&c[i] % sizeof(long) && i < n; i++)
                   ^~~~~~
                   __intptr_t
utils/utils/memory.h:99:15: note: each undeclared identifier is reported only once for each function it appears in
```

This is a bandaid I used:

```
diff -Naur strongswan-5.6.0-orig/src/libstrongswan/utils/utils/memory.h strongswan-5.6.0/src/libstrongswan/utils/utils/memory.h
---
strongswan-5.6.0-orig/src/libstrongswan/utils/utils/memory.h      2017-08-14 02:48:41.000000000 -0400
+++ strongswan-5.6.0/src/libstrongswan/utils/utils/memory.h      2017-09-12 01:15:29.690527667 -0400
@@ -14,6 +14,8 @@
 * for more details.
 */

+#include <stdint.h> /* for uintptr_t */
+
/**
 * @defgroup memory_i memory
 * @{ @ingroup utils_i
```

Associated revisions

Revision d4593353 - 19.09.2017 13:24 - Tobias Brunner

utils: Include stdint.h

Recent releases of glibc don't include the full stdint.h header in some network headers included by utils.h. So uintptr_t might not be defined. Since we use fixed width integers, including the latter, all over the place we make sure the complete file is included.

Fixes #2425.

History

#1 - 13.09.2017 22:59 - Noel Kuntze

This happens on Arch, too.

#2 - 14.09.2017 12:08 - Tobias Brunner

- *Tracker changed from Issue to Bug*
- *Description updated*
- *Status changed from New to Feedback*
- *Assignee set to Tobias Brunner*
- *Target version set to 5.6.1*

Looks like recent glibc releases changed some stuff concerning the stdint.h header ([Fix network headers stdint.h namespace](#)). stdint.h was previously pulled in via several network headers e.g. netinet/in.h or netdb.h, which utils.h includes. Since we use fixed width integers all over the place we could probably just include it in utils.h. I pushed a commit to the *2425-stdint* branch.

#3 - 19.09.2017 13:26 - Tobias Brunner

- *Status changed from Feedback to Closed*
- *Resolution set to Fixed*