# ELFUTILS - elflint

## Source Code Version

Affected Version : Release 0.168
Source Code URL : ftp://sourceware.org/pub/elfutils/0.168/elfutils-0.168.tar.bz2

ELFUTILS is managed privately. We are only able to submit bugs through bugzilla. The team uses git version control under git://sourceware.org/git/elfutils.git . We are unable to provide commit ID as it is manages internally with the develop team.

## Whether the PoC is downloadable from Internet

Yes. It is downloadable at https://github.com/asarubbo/poc/blob/master/00234-elfutils-heapoverflow-check_symtab_shndx?raw=true

## CVE ID

CVE-2017-7611
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-7611

## The detailed procedures that trigger the crash

### How the project programs are compiled :

1. Decompress the tarball using `tar xvf elfutils-0.168.tar.bz2`
2. Go to the decompressed directory
3. Make a build folder using `mkdir build`
4. Go into the *build* directory
5. Invoke the configure script, points compiler to afl compiler and enable the use of Address Sanitizer. This can be done by this line `AFL_USE_ASAN=1 ./../configure CC=afl-gcc CXX=afl-g++ LD=afl-gcc--disable-shared`
6. Compile/Make the code by using `AFL_USE_ASAN=1 make`

Note: `AFL_USE_ASAN=1` is to enable the use of Address Sanitizer

### The exact running arguments

1. Go to the *src* directory after compiling the code

2. Recreate the crash by running the following code `./elflint -d $FILE` where the *$FILE* is the input provided

# Description about the crashes (program locations of crash, program locations of the root cause)

The program crashes due to a heap-based overflow at line 1996, `Elf32_Word xndx = ((Elf32_Word *) data->d_buf)[cnt];`, under *src/elflint.c*

# Explanation about the bug fixes

Rather than only checking for null pointer at line 1961. The program should also check is symbol table data is big enough.
By changing from:
`const char *name = data->d_buf + name_offset;`
to:
`if (data->d_size < sizeof (Elf32_Word)|| *((Elf32_Word *) data->d_buf) != 0)`
will prevent the heap-based overflow issue.