Suricata - Bug #42
SMB app layer assumes int for pointer arith

01/04/2010 11:36 AM - Brian Rectanus

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>Kirby Kuehl</td>
</tr>
<tr>
<td>Category:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
<tr>
<td>Affected Versions:</td>
<td></td>
</tr>
<tr>
<td>Difficulty:</td>
<td></td>
</tr>
<tr>
<td>Effort:</td>
<td></td>
</tr>
<tr>
<td>Label:</td>
<td></td>
</tr>
</tbody>
</table>

Description

app-layer-smb.c assumes that pointer arith results in "int", but on 64-bit linux this is "long int".

EX: static int SMBGetByteCount(Flow *f, void *smb_state, AppLayerParserState *pstate, uint8_t *input, uint32_t input_len,
AppLayerParserResult *output)

uint8_t *p = input;
...
SCReturnInt(p - input);

Should return "int", not "long int". On 64-bit Linux and causes compile to fail with debug on (due to debugging output around SCReturnInt() macro) when using `-Wall -Werror -Wextra -Wno-unused-parameter` on 64-bit linux:

if gcc -DHAVE_CONFIG_H -l -l -l -l/home/brectanus/projects/oisf/deps/include -Wall -Weror -Wextra -Wno-unused-parameter
-g -Wall -no-strict-aliasing -D_BSD_SOURCE -D_BSD_SOURCE -D__FAVOR_BSD -DHAVE_NET_EHTERNET_H -I/usr/include
-DLIBPCAP_VERSION_MAJOR=1 -DUNITTESTS -DDEBUG -MT app-layer-smb.o -MD -MP -MF ".deps/app-layer-smb.Tpo" -c -o
app-layer-smb.o app-layer-smb.c; \nthen mv -f ".deps/app-layer-smb.Tpo" ".deps/app-layer-smb.Po"; else rm -f ".deps/app-layer-smb.Tpo"; exit 1; fi
cc1: warnings being treated as errors
app-layer-smb.c: In function 'SMBGetByteCount':
app-layer-smb.c:425: error: format '%d' expects type 'int', but argument 4 has type 'long int'
app-layer-smb.c:425: error: format '%d' expects type 'int', but argument 4 has type 'long int'
app-layer-smb.c: In function 'SMBParseWordCount':
app-layer-smb.c:458: error: format '%d' expects type 'int', but argument 4 has type 'long int'
app-layer-smb.c:458: error: format '%d' expects type 'int', but argument 4 has type 'long int'
app-layer-smb.c: In function 'SMBParseByteCount':
app-layer-smb.c:499: error: format '%d' expects type 'int', but argument 4 has type 'long int'
app-layer-smb.c:499: error: format '%d' expects type 'int', but argument 4 has type 'long int'
app-layer-smb.c: In function 'NBSSParseHeader':
app-layer-smb.c:543: error: format '%d' expects type 'int', but argument 4 has type 'long int'
app-layer-smb.c:543: error: format '%d' expects type 'int', but argument 4 has type 'long int'
app-layer-smb.c: In function 'SMBParseHeader':
app-layer-smb.c:701: error: format '%d' expects type 'int', but argument 4 has type 'long int'

app-layer-smb.c:701: error: format '%d' expects type 'int', but argument 4 has type 'long int'
make

2: *** [app-layer-smb.o] Error 1

History

#1 - 01/04/2010 12:50 PM - Kirby Kuehl

In util-debug.h
Could you modify
#define SCReturnInt(x) do {                                           
  if (sc_log_global_log_level >= SC_LOG_DEBUG) { 
    SCLogDebug("Returning: %d ... <<", x); 
  } 
  return x; 
} while(0)

To look like this (NOTE the only difference is the inclusion of the '\'.

06/13/2020 1/3
#define SCReturnInt(x)  \
  do {  \
  if (sc_log_global_log_level >= SC_LOG_DEBUG) {  \
    SCLogDebug("Returning: %td ... <<", x);  \
    SCLogCheckFDFilterExit(FUNCTION);  \
  }  \
  return x;  \
  } while(0)

In C99, such code can use %ld; the t size modifier corresponds to the ptrdiff_t type.

#2 - 01/04/2010 12:52 PM - Kirby Kuehl

If that works, let me know and I will do a submit a correct patch. If it doesn't work, I will create a SCReturnLongInt() macro and cast the difference to a long and use %ld

#3 - 01/04/2010 02:21 PM - Brian Rectanus

It does solve the issue there, but there is the other issue of passing an int:

c1: warnings being treated as errors
flow.c: In function 'FlowClearMemory':
flow.c:747: error: format '%td' expects type 'ptrdiff_t', but argument 4 has type 'int'
flow.c:747: error: format '%td' expects type 'ptrdiff_t', but argument 4 has type 'int'

While this may solve the warning for the debug output, I think the real issue is that the function returns "int" and you are passing a "long int" in the return macro. Probably better would be to do proper casting instead of assuming that it is an int? Though the issue is not there w/o debug because the compiler sees that it needs to be an int in the return. I prefer casts, which shows that the programmer knows the issue is there.

#4 - 01/04/2010 03:14 PM - Kirby Kuehl

- File 0001-64-bit-portability.patch added

Please try the attached patch. It introduces a new SCReturnLongInt() macro. Rather than use ptrdiff_t or the t format specifier, I casted everything to long int.

I use pointer arithmetic throughout smb, smb2, and dcerpc decoding so I have also made the appropriate modifications there as well.

If this works well for you, we can close the ticket.

#5 - 01/04/2010 03:15 PM - Kirby Kuehl

Grr. Wait. Another patch coming soon :)

#6 - 01/04/2010 03:35 PM - Kirby Kuehl

- File 0001-64-bit-fixes-second-try.patch added

Try this one. Note to Victor, I also compile cleanly on Fedora 12.

-Wall -Werror -Wextra -Wno-unused-parameter

#7 - 01/04/2010 04:11 PM - Victor Julien

Wouldn't it make more sense to use a fixed size type like uint32_t?

#8 - 01/04/2010 04:47 PM - Brian Rectanus

Kirby: That patch does indeed remove the warnings for me, however...

Victor: Agreed, uint32_t seems adequate (providing no negative returns). Though "int" seems fine as well, given proper handling/casts. Or maybe just use ptrdiff_t, but there is no return Macro for this.

#9 - 01/04/2010 04:52 PM - Kirby Kuehl

- File 0001-64-bit-portability.patch added

Here is a modified version that uses uint32_t.

#10 - 01/04/2010 05:41 PM - Brian Rectanus

That last patch works for me as well (I like it better).
#11 - 01/05/2010 02:45 AM - Victor Julien
After applying the patch the unittest DCERPCParserTest01 gets stuck an an endless loop.

#12 - 01/05/2010 09:05 AM - Kirby Kuehl
- File 0001-64-bit-portability.patch added

#13 - 01/07/2010 06:08 AM - Victor Julien
- Status changed from New to Closed

Patch applied, thanks guys!

**Files**

<table>
<thead>
<tr>
<th>File Name</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001-64-bit-portability.patch</td>
<td>22.2 KB</td>
<td>01/04/2010</td>
<td>Kirby Kuehl</td>
</tr>
<tr>
<td>0001-64-bit-fixes-second-try.patch</td>
<td>31.8 KB</td>
<td>01/04/2010</td>
<td>Kirby Kuehl</td>
</tr>
<tr>
<td>0001-64-bit-portability.patch</td>
<td>29.7 KB</td>
<td>01/04/2010</td>
<td>Kirby Kuehl</td>
</tr>
<tr>
<td>0001-64-bit-portability.patch</td>
<td>33.5 KB</td>
<td>01/05/2010</td>
<td>Kirby Kuehl</td>
</tr>
<tr>
<td>0001-64-bit-portability.patch</td>
<td>33.5 KB</td>
<td>01/05/2010</td>
<td>Kirby Kuehl</td>
</tr>
<tr>
<td>0001-64-bit-portability.patch</td>
<td>33.5 KB</td>
<td>01/05/2010</td>
<td>Kirby Kuehl</td>
</tr>
</tbody>
</table>