Closed Issue created 2 months ago by Qiuhao Li

Stack Overflow Write - OPUS dissector - dissect_opus() frames

Summary

In dissect_opus() at wireshark-3.6.8/epan/dissectors/packet-opus.c:254, frame_count is truncated below 0x3F (63) and checked if framesize * frame_count > 120 * MAX_FRAMES_COUNT. But when stack variable frames[MAX_FRAMES_COUNT] is indexed with frame_count later, **we didn't make sure frame_count won't excess MAX_FRAMES_COUNT, leading to stack-over-flow write when the begin/size values are put into frames later.**

This flaw affects the current released stable wireshark/tshark/... on Linux/Windows/Mac/..., previous versions may also be affected. This issue hasn't been reported elsewhere.

Steps to reproduce

Open the attachment rtp_opus_stackoverflow_poc.pcap with wireshark or tshark. For wireshark GUI it will crash. For tshark it will complain "*** stack smashing detected ***: terminated Aborted (core dumped)":

```
qiuhao@VBox:~/wireshark_src/wireshark-3.6.8$ tshark -r ./rtp_opus_stackoverflow_poc.pcap
    1   0.000000 10.100.147.143 →10.100.148.44 SIP/SDP 2799 Request: INVITE sip:10.100.148.44 |
*** stack smashing detected ***: terminated
Aborted (core dumped)
```

I reproduced this flaw on the current stable version (wireshark-3.6.8) and the latest release version (wireshark-4.0.0rc2). You can view the ASAN report below for more details.

What is the current bug behavior?

CWE-121: Stack-based Buffer Overflow (4.8).

What is the expected correct behavior?

Make sure frame_count is not bigger than MAX_FRAMES_COUNT.

Relevant logs and/or screenshots

ASAN Report:

```
qiuhao@VBox:~/wireshark_src/wireshark-3.6.8$ ./run/tshark -r ./rtp_opus_stackoverflow_poc.pcap

    1   0.000000 10.100.147.143 → 10.100.148.44 SIP/SDP 2799 Request: INVITE sip:10.100.148.44 |

=================================================================

==83225==ERROR: AddressSanitizer: stack-buffer-overflow on address 0x7ffc7ffcd4e2 at pc 0x7fe14f6117

WRITE of size 2 at 0x7ffc7ffcd4e2 thread T0

    #0 0x7fe14f611704 in parse_size_field /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors

    #1 0x7fe14f612764 in dissect_opus /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors/pac

    #2 0x7fe14e3e557d in call_dissector_through_handle /home/qiuhao/wireshark_src/wireshark-3.6.8/ep

    #3 0x7fe14e3e5b2f in call_dissector_work /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet.

    #4 0x7fe14e3e8c05 in dissector_try_string_new /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/pa

    #5 0x7fe14e3e8c9a in dissector_try_string /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet

    #6 0x7fe14f875146 in process_rtp_payload /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissect

    #7 0x7fe14f875671 in dissect_rtp_data /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors

    #8 0x7fe14f879c92 in dissect_rtp /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors/pack

    #9 0x7fe14e3e557d in call_dissector_through_handle /home/qiuhao/wireshark_src/wireshark-3.6.8/ep
```

```
#10 0x7fe14e3e5b2f in call_dissector_work /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet
#11 0x7fe14e3ee90e in call_dissector_only /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet
#12 0x7fe14e3b2b70 in try_conversation_call_dissector_helper /home/qiuhao/wireshark_src/wireshar
#13 0x7fe14e3b2d60 in try_conversation_dissector /home/qiuhao/wireshark_src/wireshark-3.6.8/epan
#14 0x7fe14fc6d8c6 in decode_udp_ports /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissector
#15 0x7fe14fc722cf in dissect /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors/packet-
#16 0x7fe14fc7239f in dissect_udp /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors/pac
#17 0x7fe14e3e557d in call_dissector_through_handle /home/qiuhao/wireshark_src/wireshark-3.6.8/e
#18 0x7fe14e3e5b2f in call_dissector_work /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet
#19 0x7fe14e3e80a1 in dissector_try_uint_new /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/pac
#20 0x7fe14f07e700 in ip_try_dissect /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors/
#21 0x7fe14f081668 in dissect_ip_v4 /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors/p
#22 0x7fe14e3e557d in call_dissector_through_handle /home/qiuhao/wireshark_src/wireshark-3.6.8/e
#23 0x7fe14e3e5b2f in call_dissector_work /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet
#24 0x7fe14e3e80a1 in dissector_try_uint_new /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/pac
#25 0x7fe14e3e813c in dissector_try_uint /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet.
#26 0x7fe14ece5264 in dissect_ethertype /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissecto
#27 0x7fe14e3e557d in call_dissector_through_handle /home/qiuhao/wireshark_src/wireshark-3.6.8/e
#28 0x7fe14e3e5b2f in call_dissector_work /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet
#29 0x7fe14e3ee90e in call_dissector_only /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet
#30 0x7fe14e3ee955 in call_dissector_with_data /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/p
#31 0x7fe14f9cbc9f in dissect_payload /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors
#32 0x7fe14f9cc142 in dissect_sll_common /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissect
#33 0x7fe14f9cc2c0 in dissect_sll_v1 /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors/
#34 0x7fe14e3e557d in call_dissector_through_handle /home/qiuhao/wireshark_src/wireshark-3.6.8/e
#35 0x7fe14e3e5b2f in call_dissector_work /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet
#36 0x7fe14e3ee90e in call_dissector_only /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet
#37 0x7fe14ed6003d in dissect_frame /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors/p
#38 0x7fe14e3e557d in call_dissector_through_handle /home/qiuhao/wireshark_src/wireshark-3.6.8/e
#39 0x7fe14e3e5b2f in call_dissector_work /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet
#40 0x7fe14e3ee90e in call_dissector_only /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet
#41 0x7fe14e3ee955 in call_dissector_with_data /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/p
#42 0x7fe14e3e3dfe in dissect_record /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/packet.c:62
#43 0x7fe14e3c2448 in epan_dissect_run_with_taps /home/qiuhao/wireshark_src/wireshark-3.6.8/epan
#44 0x5650067314cc in process_packet_single_pass /home/qiuhao/wireshark_src/wireshark-3.6.8/tsha
#45 0x56500672f645 in process_cap_file_single_pass /home/qiuhao/wireshark_src/wireshark-3.6.8/ts
```

```
    #46 0x565006730633 in process_cap_file /home/qiuhao/wireshark_src/wireshark-3.6.8/tshark.c:3674
    #47 0x56500672a574 in main /home/qiuhao/wireshark_src/wireshark-3.6.8/tshark.c:2103
    #48 0x7fe145f5fd8f in __libc_start_call_main ../sysdeps/nptl/libc_start_call_main.h:58
    #49 0x7fe145f5fe3f in __libc_start_main_impl ../csu/libc-start.c:392
    #50 0x565006723c94 in _start (/home/qiuhao/wireshark_src/wireshark-3.6.8/run/tshark+0x33c94)

Address 0x7ffc7ffcd4e2 is located in stack of thread T0 at offset 258 in frame
    #0 0x7fe14f611a1d in dissect_opus /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dissectors/pac

  This frame has 4 object(s):
    [32, 33) 'toc' (line 161)
    [48, 50) 'framesize' (line 164)
    [64, 256) 'frames' (line 168) <== Memory access at offset 258 overflows this variable
    [320, 322) 'octet' (line 161)
HINT: this may be a false positive if your program uses some custom stack unwind mechanism, swapcont
      (longjmp and C++ exceptions *are* supported)
SUMMARY: AddressSanitizer: stack-buffer-overflow /home/qiuhao/wireshark_src/wireshark-3.6.8/epan/dis
Shadow bytes around the buggy address:
  0x10000fff1a40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10000fff1a50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10000fff1a60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10000fff1a70: 00 00 00 00 00 00 00 00 00 00 00 00 f1 f1 f1 f1
  0x10000fff1a80: 01 f2 02 f2 00 00 00 00 00 00 00 00 00 00 00 00
=>0x10000fff1a90: 00 00 00 00 00 00 00 00 00 00 00 00[f2]f2 f2 f2
  0x10000fff1aa0: f2 f2 f2 f2 02 f3 f3 f3 00 00 00 00 00 00 00 00
  0x10000fff1ab0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10000fff1ac0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10000fff1ad0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10000fff1ae0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Shadow byte legend (one shadow byte represents 8 application bytes):
  Addressable:           00
  Partially addressable: 01 02 03 04 05 06 07
  Heap left redzone:       fa
  Freed heap region:       fd
  Stack left redzone:      f1
  Stack mid redzone:       f2
```

```
  Stack right redzone:     f3
  Stack after return:      f5
  Stack use after scope:   f8
  Global redzone:          f9
  Global init order:       f6
  Poisoned by user:        f7
  Container overflow:      fc
  Array cookie:            ac
  Intra object redzone:    bb
  ASan internal:           fe
  Left alloca redzone:     ca
  Right alloca redzone:    cb
  Shadow gap:              cc
==83225==ABORTING
```

Build information

```
TShark (Wireshark) 3.6.8 (Git commit d25900c51508)

Copyright 1998-2022 Gerald Combs <gerald@wireshark.org> and contributors.
License GPLv2+: GNU GPL version 2 or later <https://www.gnu.org/licenses/gpl-2.0.html>
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

Compiled (64-bit) using GCC 11.2.0, with libpcap, with POSIX capabilities
(Linux), with libnl 3, with GLib 2.72.1, with zlib 1.2.11, with Lua 5.2.4, with
GnuTLS 3.7.3 and PKCS #11 support, with Gcrypt 1.9.4, with MIT Kerberos, with
MaxMind DB resolver, with nghttp2 1.43.0, with brotli, with LZ4, with Zstandard,
with Snappy, with libxml2 2.9.13, with libsmi 0.4.8.

Running on Linux 5.15.0-47-generic, with 11th Gen Intel(R) Core(TM) i7-11850H @
2.50GHz (with SSE4.2), with 7949 MB of physical memory, with GLib 2.72.1, with
zlib 1.2.11, with libpcap 1.10.1 (with TPACKET_V3), with c-ares 1.18.1, with
GnuTLS 3.7.3, with Gcrypt 1.9.4, with nghttp2 1.43.0, with brotli 1.0.9, with
LZ4 1.9.3, with Zstandard 1.4.8, with libsmi 0.4.8, with LC_TYPE=en_US.UTF-8,
binary plugins supported (0 loaded).
```

Omit 4.0.0 and Windows/Linux/Mac GUI Version...

To upload designs, you'll need to enable LFS and have an admin enable hashed storage. More information

Tasks 0

No tasks are currently assigned. Use tasks to break down this issue into smaller parts.

Linked items 0

Link issues together to show that they're related or that one is blocking others. Learn more.

Related merge requests 3

opus: Don't overflow a signed 16-bit integer
!8248

opus: Don't overflow a signed 16-bit integer
!8250

opus: Don't overflow a signed 16-bit integer
!8251

When these merge requests are accepted, this issue will be closed automatically.

Activity

John Thacker @johnthacker · 2 months ago — Developer

The problem seems to stem from ecd1ab5b

`opus_packet_get_samples_per_frame(const unsigned char *data, int16_t Fs)` is called with 48000U as the second parameter, which is too large to fit into a *signed* 16 bit integer, making all the calculations wrong.

John Thacker mentioned in merge request !8248 (merged) 1 month ago

John Thacker closed via commit 749a8d09 1 month ago

A Wireshark GitLab Utility closed via merge request !8248 (merged) 1 month ago

John Thacker mentioned in commit 0e1a66ec 1 month ago

John Thacker mentioned in merge request !8250 (merged) 1 month ago

John Thacker mentioned in commit 5db46d3a 1 month ago

John Thacker mentioned in merge request !8251 (merged) 1 month ago

Qiuhao Li @QiuhaoLi · 1 month ago — Author

@geraldcombs Should this bug be in Security Advisories?

Gerald Combs @geraldcombs · 1 month ago — Owner

Hi @QiuhaoLi, it should. I typically write up advisories and request CVEs one or two days prior to each release. In this case that will probably be on October 24th or 25th.

How would you prefer to be credited in the advisory & CVE?

Qiuhao Li @QiuhaoLi · 1 month ago — Author

@geraldcombs thanks. Credit: Qiuhao Li of Zoom Video Communications, Inc.

Please register or sign in to reply

Gerald Combs @geraldcombs · 4 weeks ago

Owner

CVE-2022-3725.

Please register or sign in to reply