Closed   Issue created 1 year ago by Doneing

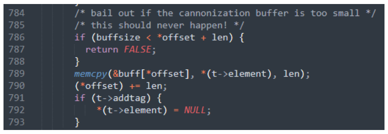
Heap-buffer-overflow in canonify_unencrypted_header at packet-c1222-template.c

Summary

In Wireshark-3.5.1rc0, the epsem dissector could cause out-of-bounds memory reads.

Bug information

In line **789** at packet-c1222-template.c

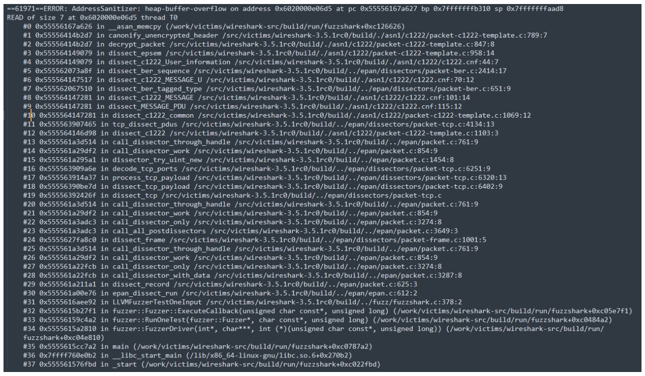


```
784         /* bail out if the cannonization buffer is too small */
785         /* this should never happen! */
786         if (buffsize < *offset + len) {
787           return FALSE;
788         }
789         memcpy(&buff[*offset], *(t->element), len);
790         (*offset) += len;
791         if (t->addtag) {
792             *(t->element) = NULL;
793         }
```

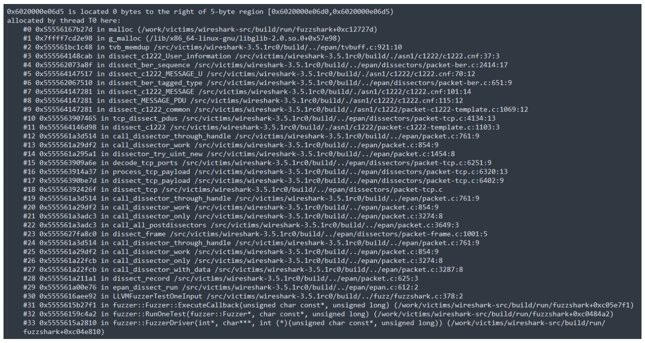
Steps to reproduce

- First, compile the program **fuzzshark** through ASAN. cmake -GNinja -DCMAKE_C_COMPILER=clang-12 -DCMAKE_CXX_COMPILER=clang++-12 -DDISABLE_WERROR=ON -DOSS_FUZZ=ON -DENABLE_STATIC=ON -DENABLE_PLUGINS=OFF -DENABLE_PCAP=OFF -DENABLE_GNUTLS=OFF -DBUILD_wireshark=OFF /wireshark-3.5.1rc0 && ninja all-fuzzers
- Second, set environment variables. `export FUZZSHARK_TARGET=tcp`
- Third, run the program with payload packet. `./fuzzshark tcp-crash-sample-001` tcp-crash-sample-001

Crash state with ASAN



```
==61971==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x6020000e06d5 at pc 0x55556167a627 bp 0x7fffffffb310 sp 0x7fffffffaad8
READ of size 7 at 0x6020000e06d5 thread T0
    #0 0x55556167a626 in __asan_memcpy (/work/victims/wireshark-src/build/run/fuzzshark+0xc126626)
    #1 0x55556414b2d7 in canonify_unencrypted_header /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/packet-c1222-template.c:789:7
    #2 0x55556414b2d7 in decrypt_packet /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/packet-c1222-template.c:847:8
    #3 0x555564149079 in dissect_epsem /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/packet-c1222-template.c:958:14
    #4 0x555564149079 in dissect_c1222_User_information /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/c1222.cnf:44:7
    #5 0x555562073a8f in dissect_ber_sequence /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-ber.c:2414:17
    #6 0x555564147517 in dissect_c1222_MESSAGE_U /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/c1222.cnf:70:12
    #7 0x555562067510 in dissect_ber_tagged_type /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-ber.c:651:9
    #8 0x555564147281 in dissect_c1222_MESSAGE /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/c1222.cnf:101:14
    #9 0x555564147281 in dissect_MESSAGE_PDU /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/c1222.cnf:115:12
    #10 0x555564147281 in dissect_c1222_common /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/packet-c1222-template.c:1069:12
    #11 0x555563907465 in tcp_dissect_pdus /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-tcp.c:4134:13
    #12 0x555564146d98 in dissect_c1222 /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/packet-c1222-template.c:1103:3
    #13 0x555561a3d514 in call_dissector_through_handle /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:761:9
    #14 0x555561a29df2 in call_dissector_work /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:854:9
    #15 0x555561a295a1 in dissector_try_uint_new /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:1454:8
    #16 0x555563909a6e in decode_tcp_ports /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-tcp.c:6251:9
    #17 0x555563914a37 in process_tcp_payload /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-tcp.c:6320:13
    #18 0x55556390be7d in dissect_tcp_payload /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-tcp.c:6402:9
    #19 0x55556392426f in dissect_tcp /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-tcp.c
    #20 0x555561a3d514 in call_dissector_through_handle /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:761:9
    #21 0x555561a29df2 in call_dissector_work /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:854:9
    #22 0x555561a3adc3 in call_dissector_only /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:3274:8
    #23 0x555561a3adc3 in call_all_postdissectors /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:3649:3
    #24 0x5555627fa8c0 in dissect_frame /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-frame.c:1001:5
    #25 0x555561a3d514 in call_dissector_through_handle /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:761:9
    #26 0x555561a29df2 in call_dissector_work /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:854:9
    #27 0x555561a22fcb in call_dissector_only /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:3274:8
    #28 0x555561a22fcb in call_dissector_with_data /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:3287:8
    #29 0x555561a211a1 in dissect_record /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:625:3
    #30 0x555561a00e76 in epan_dissect_run /src/victims/wireshark-3.5.1rc0/build/../epan/epan.c:612:2
    #31 0x5555616aee92 in LLVMFuzzerTestOneInput /src/victims/wireshark-3.5.1rc0/build/../fuzz/fuzzshark.c:378:2
    #32 0x5555615b27f1 in fuzzer::Fuzzer::ExecuteCallback(unsigned char const*, unsigned long) (/work/victims/wireshark-src/build/run/fuzzshark+0xc05e7f1)
    #33 0x55556159c4a2 in fuzzer::RunOneTest(fuzzer::Fuzzer*, char const*, unsigned long) (/work/victims/wireshark-src/build/run/fuzzshark+0xc0484a2)
    #34 0x5555615a2810 in fuzzer::FuzzerDriver(int*, char***, int (*)(unsigned char const*, unsigned long)) (/work/victims/wireshark-src/build/run/
    fuzzshark+0xc04e810)
    #35 0x5555615cc7a2 in main (/work/victims/wireshark-src/build/run/fuzzshark+0xc0787a2)
    #36 0x7ffff760e0b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x270b2)
    #37 0x555561576fbd in _start (/work/victims/wireshark-src/build/run/fuzzshark+0xc022fbd)
```



```
0x6020000e06d5 is located 0 bytes to the right of 5-byte region [0x6020000e06d0,0x6020000e06d5)
allocated by thread T0 here:
    #0 0x55556167b27d in malloc (/work/victims/wireshark-src/build/run/fuzzshark+0xc12727d)
    #1 0x7ffff7cd2e98 in g_malloc (/lib/x86_64-linux-gnu/libglib-2.0.so.0+0x57e98)
    #2 0x555561bc1c48 in tvb_memdup /src/victims/wireshark-3.5.1rc0/build/../epan/tvbuff.c:921:10
    #3 0x555564148cab in dissect_c1222_User_information /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/c1222.cnf:37:3
    #4 0x555562073a8f in dissect_ber_sequence /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-ber.c:2414:17
    #5 0x555564147517 in dissect_c1222_MESSAGE_U /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/c1222.cnf:70:12
    #6 0x555562067510 in dissect_ber_tagged_type /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-ber.c:651:9
    #7 0x555564147281 in dissect_c1222_MESSAGE /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/c1222.cnf:101:14
    #8 0x555564147281 in dissect_MESSAGE_PDU /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/c1222.cnf:115:12
    #9 0x555564147281 in dissect_c1222_common /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/packet-c1222-template.c:1069:12
    #10 0x555563907465 in tcp_dissect_pdus /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-tcp.c:4134:13
    #11 0x555564146d98 in dissect_c1222 /src/victims/wireshark-3.5.1rc0/build/./asn1/c1222/packet-c1222-template.c:1103:3
    #12 0x555561a3d514 in call_dissector_through_handle /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:761:9
    #13 0x555561a29df2 in call_dissector_work /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:854:9
    #14 0x555561a295a1 in dissector_try_uint_new /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:1454:8
    #15 0x555563909a6e in decode_tcp_ports /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-tcp.c:6251:9
    #16 0x555563914a37 in process_tcp_payload /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-tcp.c:6320:13
    #17 0x55556390be7d in dissect_tcp_payload /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-tcp.c:6402:9
    #18 0x55556392426f in dissect_tcp /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-tcp.c
    #19 0x555561a3d514 in call_dissector_through_handle /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:761:9
    #20 0x555561a29df2 in call_dissector_work /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:854:9
    #21 0x555561a3adc3 in call_dissector_only /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:3274:8
    #22 0x555561a3adc3 in call_all_postdissectors /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:3649:3
    #23 0x5555627fa8c0 in dissect_frame /src/victims/wireshark-3.5.1rc0/build/../epan/dissectors/packet-frame.c:1001:5
    #24 0x555561a3d514 in call_dissector_through_handle /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:761:9
    #25 0x555561a29df2 in call_dissector_work /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:854:9
    #26 0x555561a22fcb in call_dissector_only /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:3274:8
    #27 0x555561a22fcb in call_dissector_with_data /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:3287:8
    #28 0x555561a211a1 in dissect_record /src/victims/wireshark-3.5.1rc0/build/../epan/packet.c:625:3
    #29 0x555561a00e76 in epan_dissect_run /src/victims/wireshark-3.5.1rc0/build/../epan/epan.c:612:2
    #30 0x5555616aee92 in LLVMFuzzerTestOneInput /src/victims/wireshark-3.5.1rc0/build/../fuzz/fuzzshark.c:378:2
    #31 0x5555615b27f1 in fuzzer::Fuzzer::ExecuteCallback(unsigned char const*, unsigned long) (/work/victims/wireshark-src/build/run/fuzzshark+0xc05e7f1)
    #32 0x55556159c4a2 in fuzzer::RunOneTest(fuzzer::Fuzzer*, char const*, unsigned long) (/work/victims/wireshark-src/build/run/fuzzshark+0xc0484a2)
    #33 0x5555615a2810 in fuzzer::FuzzerDriver(int*, char***, int (*)(unsigned char const*, unsigned long)) (/work/victims/wireshark-src/build/run/
    fuzzshark+0xc04e810)
```

Sample capture file

tcp-crash-sample-001

Edited 1 year ago by Gerald Combs

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 4

C12.22: Track our allocation sizes
!4851

C12.22: Track our allocation sizes.
!4886

C12.22: Track our allocation sizes.
!4887

C12.22: Track our allocation sizes.
!4891

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

Activity

Gerald Combs @geraldcombs · 1 year ago   Owner

I'm having trouble replicating the issue here with the file provided. I'm also having trouble converting the raw payload to a usable capture file. @Lekensteyn, Jakub's samples_to_pcap doesn't appear to support arbitrary TCP ports. Is there a better tool for this?

Doneing @DoneingCK · 1 year ago   Author

I am sorry that the payload `tcp-crash-sample-001` provided earlier is wrong. I attached a new payload `tcp-crash-sample-new`.
tcp-crash-sample-new

Set environment `ASAN_OPTIONS` variables to output asan logs.

- export ASAN_OPTIONS=log_path= `path` /sanitize:log_exe_name=true:abort_on_error=1:symbolize=1

Doneing @DoneingCK · 1 year ago   Author

@geraldcombs Can you verify this bug?

Gerald Combs @geraldcombs · 1 year ago Owner

I can. Thank you for the detailed bug reports!

Doneing @DoneingCK · 1 year ago Author

Hi @geraldcombs , can I apply for a CVE ID for this vulnerability?

Gerald Combs @geraldcombs · 1 year ago Owner

It's been assigned CVE-2021-39922. See also wnpa-sec-2021-12.

Please register or sign in to reply

Gerald Combs mentioned in merge request !4851 (merged) 1 year ago

Gerald Combs closed via commit `b760c356` 1 year ago

Gerald Combs mentioned in merge request !4886 (merged) 1 year ago

Gerald Combs mentioned in merge request !4887 (merged) 1 year ago

Gerald Combs mentioned in commit `7ac1d5be` 1 year ago

Gerald Combs mentioned in merge request !4891 (merged) 1 year ago

Gerald Combs mentioned in commit `9a1ef88c` 1 year ago

Gerald Combs mentioned in commit `1b9972ae` 1 year ago

Gerald Combs changed title from **heap-buffer-overflow in canonify_unencrypted_header at packet-c1222-template.c** to **Heap-buffer-overflow in canonify_unencrypted_header at packet-c1222-template.c** 1 year ago

Please register or sign in to reply