

ERROR:epan/proto.c:9868:hfinfo_number_value_format_display: code should not be reached

Wireshark 3.2.6 (Git v3.2.6 packaged as 3.2.6-1) from Ubuntu repo and fresh compiled from git master both crash with the same assert.

GDB backtrace:

```
(gdb) bt
#0  __GI_raise (sig=11@entry=6) at ../sysdeps/unix/sysv/linux/raise.c:50
#1  0x00007ffff5d12859 in __GI_abort () at abort.c:79
#2  0x00007ffff603fb43 in () at /usr/lib/x86_64-linux-gnu/libglib-2.0.so.0
#3  0x00007ffff609cb2f in g_assert_message_expr () at /usr/lib/x86_64-linux-gnu/libglib-2.0.so.0
#4  0x00007ffff417f787 in hfinfo_number_value_format_display (hfinfo=hfinfo@entry=0x7ffff671b828 <hf+2088>, display=optimal,
proto_custom_set (tree=0x55555c2e4960, field_ids=0x555558258a90 = {...}, occurrence=0, result=0x5
0x00007ffff415092d in epan_custom_set (edt=edt@entry=0x7ffff67d470, field_ids=<optimized out>, occurrence=<optimized o
#7  0x00007ffff4152222 in col_custom_set_edt (edt=edt@entry=0x7ffff67d470, cinfo=0x55555d4cb90 <<file+304>) at ../epan/co
#8  0x00007ffff4126949 in epan_dissect_fill_in_columns (edt=edt@entry=0x7ffff67d470, fill_col_exprs=fill_col_exprs@entry=0
#9  0x000055555904a45c in PacketListRecord::dissect_capture_file*, bool) (this=this@entry=0x55555ab994cb, cap_file=0x5555
#10 0x000055555904a5ab in PacketListRecord::columnString(capture_file*, int, bool) (this=0x55555ab994cb, cap_file=optimal
#11 0x000055555904da3 in PacketListModel::ensureRowColorized(int) (row=18497, this=0x5555563fc600) at ../ui/qt/models/pack
#12 PacketListModel::ensureRowColorized(int) (this=0x5555563fc600, row=18497) at ../ui/qt/models/packet_list_model.cpp:755
#13 0x000055555904e624 in PacketListModel::dissectIdle(bool) (this=0x5555563fc600, reset=<optimized out>) at ../ui/qt/model
#14 0x00007ffff6d0d320 in () at /usr/lib/x86_64-linux-gnu/libQt5Core.so.5
#15 0x00007ffff6d10f03 in () at /usr/lib/x86_64-linux-gnu/libQt5Core.so.5
#16 0x00007ffff6d035f8 in QObject::event(QEvent*) () at /usr/lib/x86_64-linux-gnu/libQt5Core.so.5
#17 0x00007ffff6c15813 in QApplicationPrivate::notify_helper(QObject*, QEvent*) () at /usr/lib/x86_64-linux-gnu/libQt5Widge
#18 0x00007ffff6cfd71ca in QCoreApplication::notifyInternal2(QObject*, QEvent*) () at /usr/lib/x86_64-linux-gnu/libQt5Core.s
#19 0x00007ffff6d2db73 in QTimerInfoList::activateTimers() () at /usr/lib/x86_64-linux-gnu/libQt5Core.so.5
#20 0x00007ffff6d2e4b4 in () at /usr/lib/x86_64-linux-gnu/libQt5Core.so.5
#21 0x00007ffff673f9d in g_main_context_dispatch () at /usr/lib/x86_64-linux-gnu/libglib-2.0.so.0
#22 0x00007ffff674220 in () at /usr/lib/x86_64-linux-gnu/libglib-2.0.so.0
#23 0x00007ffff6742a3 in g_main_context_iteration () at /usr/lib/x86_64-linux-gnu/libglib-2.0.so.0
#24 0x00007ffff6d2e843 in QEventDispatcherGlib::processEvents(QFlags<QEventLoop::ProcessEventsFlag>) () at /usr/lib/x86_64
#25 0x00007ffff6dfe4a0 in QEventLoop::exec(QFlags<QEventLoop::ProcessEventsFlag>) () at /usr/lib/x86_64-linux-gnu/libQt5Core
#26 0x00007ffff6d8fc6 in QCoreApplication::exec() () at /usr/lib/x86_64-linux-gnu/libQt5Core.so.5
#27 0x0000555559562c31 in main(int, char**) (argc=<optimized out>, argv=<optimized out>) at ../ui/qt/main.cpp:933
(gdb) up
#1  0x00007ffff5d12859 in __GI_abort () at abort.c:79
79  abort.c: No such file or directory.
(gdb)
#2  0x00007ffff603fb43 in ?? () from /usr/lib/x86_64-linux-gnu/libglib-2.0.so.0
(gdb)
#3  0x00007ffff609cb2f in g_assert_message_expr () from /usr/lib/x86_64-linux-gnu/libglib-2.0.so.0
(gdb)
#4  0x00007ffff417f787 in hfinfo_number_value_format_display (hfinfo=hfinfo@entry=0x7ffff671b828 <hf+2088>, display=optimal,
g_assert_not_reached();
(gdb) print *hfinfo
$8 = {name = 0x7ffff4a07859 "Checksum Status", abbrev = 0x7ffff49f9319 "tcp.checksum.status", type = FT_UINT8, display = 0,
same_name_prev_id = -1, same_name_next = 0x0}
```

Edited 2 years 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 3

🔗 TCP: do not use an unknown status when the checksum is 0xffff

184



🔗 TCP: do not use an unknown status when the checksum is 0xffff

186



🔗 TCP: do not use an unknown status when the checksum is 0xffff

1102



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

Activity

- 🧑🏻 **Pascal Quantin** @pquantin · 2 years ago Developer

It seems like you have a custom column set for the tcp.checksum.status field triggering the error. So what's the content of your preferences file (more specifically the user interface columns section)?
- 🧑🏻 **Andreas Schultz** @aschultz · 2 years ago Author

The profile contains this:

```
gui.column.format:
  "Checksum Status", "%Cus:tcp.checksum.status:0:R",
  "No.", "%m",
  "Time", "%t",
  "Source", "%s",
  "Destination", "%d",
  "Protocol", "%p",
  "Length", "%L",
  "Response Time", "%Cus:gtpv2.response_time:0:R",
  "Time", "%Cus:gtp.time:0:R",
  "Sequence number", "%Cus:gtp.seq_number:0:R",
  "Recovery", "%Cus:gtp.recovery:0:R",
  "Info", "%i"
```

I have remove and readded the column, but the format of the Checksum Status entry remains the same.

Strange is also that it works for most PCAPs, but for a specific file it crashes.
- 🧑🏻 **Pascal Quantin** @pquantin · 2 years ago Developer


OK thanks for the clarification. If it's specific to a given file, we will need it for the investigation.
- 🧑🏻 **Andreas Schultz** @aschultz · 2 years ago Author

The file comes from a benchmark run. It is therefore too large to upload here.

Can you download it from <https://drive.google.com/file/d/1so9eNlK9hJL13DyqamZG5K-r37du117/view?usp=sharing> ??

And let me know once you have it, so I can disable the link again.
- 🧑🏻 **Pascal Quantin** @pquantin · 2 years ago Developer

Thanks, you can remove the link. Fix inbound.
- 🗨️ **Pascal Quantin** mentioned in merge request [184 \(merged\)](#) 2 years ago
- 🔒 **Pascal Quantin** closed via merge request [184 \(merged\)](#) 2 years ago
- 🗨️ **Pascal Quantin** mentioned in commit [eb727259](#) 2 years ago

 [Pascal Quantin](#) mentioned in merge request [186 \(merged\)](#) 2 years ago





[Valerii Zapodovnikov](#) @ValZapod · 2 years ago


Contributor

So this is a continuation of [#16334](#), packets 18498, 58165, etc have 0xFFFF in TCP, even though TCP CANNOT have such checksum, at all.
[@quqarris](#) What do you think about it? Is TCP in IPV4 in **GPFS** allowed to contain 0xFFFF? I doubt it.

Edited by [Valerii Zapodovnikov](#) 2 years ago

 [Pascal Quantin](#) mentioned in commit [9d7ab8b4](#) 2 years ago

 [Pascal Quantin](#) mentioned in merge request [1102 \(merged\)](#) 2 years ago

 [Pascal Quantin](#) mentioned in commit [7f3fed16](#) 2 years ago

 [Gerald Combs](#) changed title from **ERROR:span/proto.c:9868:hfinfo_number_value_format_display: code should not be reached B** to **ERROR:span/proto.c:9868:hfinfo_number_value_format_display: code should not be reached** 2 years ago



[Gerald Combs](#) @geraldcombs · 2 years ago

Owner

[CVE-2020-25862](#)

Please [register](#) or [sign in](#) to reply