Suricata - Security #5408
filestore: Segfault with filestore enabled and forced
06/25/2022 01:40 PM - Jeff Lucovsky

<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>Philippe Antoine</td>
</tr>
<tr>
<td>Category:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td>7.0rc1</td>
</tr>
<tr>
<td>Affected Versions:</td>
<td>6.0.5</td>
</tr>
<tr>
<td>Git IDs:</td>
<td></td>
</tr>
<tr>
<td>Label:</td>
<td></td>
</tr>
<tr>
<td>CVE:</td>
<td></td>
</tr>
<tr>
<td>Severity:</td>
<td>MODERATE</td>
</tr>
</tbody>
</table>

**Description**

Suricata will crash when filestore is enabled and the bigFlows.pcap is used. This was reported in the forum: https://forum.suricata.io/t/file-store-core-dumping-on-specific-pcap/2587

I've confirmed that the crash occurs on

- master
- master-6.0.x

The crash does not occur on master-5.0.x

The bigFlows.pcap is available here: https://tcpreplay.appneta.com/wiki/captures.html#bigflows-pcap

```
master: args -c suricata.yaml -l /tmp/ll -r /home/jlucovsky/bigFlows.pcap
```

Thread 15 "W#13" received signal SIGSEGV, Segmentation fault.
[Switching to Thread 0x7fff9d9d13700 (LWP 241346)]
0x0000000000000000 in ?? ()
(gdb) bt
#0 0x0000000000000000 in ?? ()
#1 0x0000005555551cb37 in AppLayerParserGetTx (ipproto=1 \001', alproto=1, alstate=0x60c001eca3c0 , tx_id=0) at app-layer-parser.c:1126
#2 0x000000555556106875 in CloseFile (p=0x61d00606f80, f=0x612003090940, file=0x61100035ac40) at output-filedata.c:128
#3 0x000000555556107161 in OutputFiledataLogFfc (tv=0x612001d36ec0, td=0x602002e7d670, p=0x61d00668f1080, ffc=0x602002e62050, call_flags=8 "\b", file_close=false, file_trunc=false, dir=8 "\b") at output-filedata.c:129
#4 0x000000555556107493 in OutputFiledataLog (tv=0x612001d36ec0, p=0x61d00668f1080, thread_data=0x602002e7d670) at output-filedata.c:244
#5 0x0000005555561073c4 in OutputLoggerLog (tv=0x612001d36ec0, p=0x61d00668f1080, thread_data=0x602002e7d670) at output-filedata.c:129
#6 0x0000005555560ecf19 in FlowWorker (tv=0x612001d36ec0, p=0x61d00668f1080, data=0x61000016b040) at flow-worker.c:565
#7 0x0000005555558953a8 in TmThreadsSlotVarRun (tv=0x612001d36ec0, p=0x61d00668f1080, thread_data=0x602002e7d670) at tm-threads.c:457
#8 0x000000555555897682 in TmThreadsSlotVar (td=0x602002e7d670) at tm-threads.c:457
#9 0x0000007fff730d609 in start_thread (arg=optimized out) at pthread_create.c:477
#10 0x0000007fff69ab133 in clone () at ../sysdeps/unix/sysv/linux/x86_64/clone.S:95
(gdb) fr 1
#1 0x0000005555551cb37 in AppLayerParserGetTx (ipproto=1 \001', alproto=1, alstate=0x60c001eca3c0 , tx_id=0) at app-layer-parser.c:1126
1126 void *r = alp_ctx.ctxs[FlowGetProtoMapping(ipproto)][alproto].StateGetTx(alstate, tx_id);
(gdb) p alp_ctx.ctxs[FlowGetProtoMapping(ipproto)][alproto]
$1 = {Parser = {0x0, 0x0}, logger = false, first_data_dir = 0 '\000', logger_bits = 0, StateAlloc = 0x0, StateFree = 0x0, StateTransactionFree = 0x0, LocalStorageAlloc = 0x0, LocalStorageFree = 0x0

07/14/2022
(gdb) p FlowGetProtoMapping(ipproto)
$2 = 2 '\002'
(gdb) p alproto
$3 = 1
(gdb) p ipproto
$4 = 1 '\001'

master-6.0.x: args -c suricata.yaml -l /tmp/ll -r /home/jlucovsky/bigFlows.pcap

(gdb) bt
#0 0x0000000000000000 in ?? ()
#1 0x00005555555561f3383 in AppLayerParserGetTx (ipproto=1 '\001', alproto=1, alstate=0x6080004cd920, tx_id=0) at app-layer-parser.c:1095
#2 0x00005555555561f3383 in CloseFile (p=0x61d00026de80, f=0x612000d83640, file=0x6110007dc5c0) at output-filedata.c:137
#3 0x00005555555561f3383 in OutputFiledataLogFfc (tv=0x612001526be0, ff=0x61200099e0de70, call_flags=8 '\b', file_close=false, file_trunc=false, dir=8 '\b') at output-filedata.c:218
#4 0x00005555555561f3383 in OutputFiledataLog (tv=0x612001526be0, p=0x61d00026de80, thread_data=0x6020009de70, call_flags=8 '\b', file_close=false, file_trunc=false, dir=8 '\b') at output-filedata.c:253
#5 0x00005555555561f3383 in OutputLoggerLog (tv=0x612001526be0, p=0x61d00026de80, thread_data=0x6020009de70, call_flags=8 '\b', file_close=false, file_trunc=false, dir=8 '\b') at output-filedata.c:282
#6 0x00005555555561f3383 in FlowWorker (tv=0x612001526be0, p=0x61d00026de80, data=0x60d00073f1b0) at flow-worker.c:556
#7 0x00005555555561f3383 in TmThreadsSlotVarRun (tv=0x612001526be0, p=0x61d00026de80, slot=0x60600022, 5f0eo) at tm-threads.c:117
#8 0x00005555555561f3383 in TmThreadsSlotVar (td=0x612001526be0) at tm-threads.c:463
#9 0x00005555555561f3383 in start_thread (arg=optimized out) at pthread_create.c:477
#10 0x00005555555561f3383 in clone () at ../sysdeps/unix/sysv/linux/x86_64/clone.S:95
(gdb) fr 1
#1 0x0000000000000000 in AppLayerParserGetTx (ipproto=1 '\001', alproto=1, alstate=0x6080004cd920, tx_id=0) at app-layer-parser.c:1095
1095 r = alp_ctx.ctxs[FlowGetProtoMapping(ipproto)][alproto].
(gdb) list 1090
1091 void *AppLayerParserGetTx(uint8_t ipproto, AppProto alproto, void *alstate, uint64_t tx_id)
1092 {
1093 SCEnter();
1094 void * r = NULL;
1095 r = alp_ctx.ctxs[FlowGetProtoMapping(ipproto)][alproto].
1096 SCReturnPtr(r, "void *");
1097 }
1098
(gdb) p alp_ctx.ctxs[FlowGetProtoMapping(ipproto)][alproto]
$2 = {Parser = 0x0, logger = false, logger_bits = 0, StateAlloc = 0x0, StateFree = 0x0, StateTransactionFree = 0x0, LocalStorageAlloc = 0x0, LocalStorageFree = 0x0, Truncate = 0x0, StateFiles = 0x0, StateEvents = 0x0, StateGetProgress = 0x0, StateGetTxCnt = 0x0, StateGetTx = 0x0, StateGetTxIterator = 0x0, StateGetProgressCompletionStatus = 0x0, StateGetEventInfoById = 0x0, StateGetEventInfo = 0x0, GetTxDetectState = 0x0, SetTxDetectState = 0x0, GetTxData = 0x0, ApplyTxConfig = 0x0, SetStreamDepthFlag = 0x0, stream_depth = 1048576, first_data_dir = 0 '000', option_flags = 0, internal_flags = 0, RegisterUnittests = 0x0}
(gdb) p FlowGetProtoMapping(ipproto)
$3 = 2 '002'
(gdb) p alproto
$4 = 1
(gdb) p ipproto
$5 = 1 '001'

07/14/2022 2/4
Subtasks:
Security # 5431: filestore: Segfault with filestore enabled and forced (6.0.x backport)  Closed

History

#1 - 06/26/2022 01:22 PM - Jeff Lucovsky
Both crashes (master, master-6.0.x) occur at the same point in the pcap

{gdb} p p->pcap_cnt
$q = 291867

capdump (and wireshark, et al) show this to be an ICMP packet.

ICMP doesn't have a TX accessor:

$ git grep -l -e AppLayerParserRegisterGetTx
rust/src/applayer.rs
src/app-layer-dnp3.c
src/app-layer-enip.c
src/app-layer-ftp.c
src/app-layer-htp.c
src/app-layer-parser.c
src/app-layer-parser.h
src/app-layer-register.c
src/app-layer-smtp.c
src/app-layer-ssl.c
src/app-layer-template.c
src/app-layer-tftp.c

#2 - 06/26/2022 01:33 PM - Jeff Lucovsky
- Subject changed from Segfault with filestore enabled to Segfault with filestore enabled and forces

#3 - 06/26/2022 01:51 PM - Victor Julien
Certain types of ICMP errors are considered part of a TCP/UDP flow.

#4 - 06/26/2022 01:52 PM - Jeff Lucovsky
- Subject changed from Segfault with filestore enabled and forces to Segfault with filestore enabled and forced

#5 - 06/30/2022 02:52 PM - JP J
- File bad_icmp_test4.3.pcap added

Hello,

Please find attached a narrower pcap extracted from BigFlows.pcap

We have been able to reproduce 100% with the joined pcap.

Setting force-filestore to 'no' makes harder to reproduce the issue.

#6 - 07/01/2022 05:59 AM - Victor Julien
- Priority changed from Normal to Urgent

#7 - 07/01/2022 06:31 AM - Shivani Bhardwaj
- Affected Versions 6.0.5 added
- Affected Versions deleted (6.0.6, 7.0rc1)

#8 - 07/01/2022 11:03 AM - Victor Julien
- Target version changed from TBD to 7.0rc1

#9 - 07/01/2022 11:03 AM - Victor Julien
- Subject changed from Segfault with filestore enabled and forced to filestore: Segfault with filestore enabled and forced

07/14/2022
#10 - 07/01/2022 11:03 AM - Victor Julien
- Tracker changed from Bug to Security
- Severity set to MODERATE

#11 - 07/01/2022 11:04 AM - Victor Julien
- Label deleted (Needs backport to 6.0)

#12 - 07/01/2022 11:06 AM - Philippe Antoine
- Assignee changed from OISF Dev to Philippe Antoine

#13 - 07/01/2022 01:47 PM - Philippe Antoine
- Status changed from New to In Review

https://github.com/OISF/suricata/pull/7598

#14 - 07/07/2022 02:34 PM - Jason Ish
- Status changed from In Review to Resolved

#15 - 07/07/2022 02:36 PM - Jason Ish
- Status changed from Resolved to Closed

Files

<table>
<thead>
<tr>
<th>File Name</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>bad_icmp_test4.3.pcap</td>
<td>3.07 MB</td>
<td>06/30/2022</td>
<td>JP J</td>
</tr>
</tbody>
</table>