Suricata - Bug #4503
Buffer overflow in "by_rule" threshold context
05/28/2021 09:34 AM - Mats Klepsland

| Status:     | Closed       |
| Priority:   | Normal       |
| Assignee:   | Mats Klepsland |
| Category:   |             |
| Target version: | 7.0rc1       |
| Affected Versions: | needs backport to 5.0, needs backport to 6.0 |
| Difficulty: |             |
| Label:      |             |

**Description**

Several servers running Suricata has been crashing occasionally. I managed to get a PCAP file reproducing the bug on the same server, but had a hard time reproducing on my test rig. It turned out that the ordering of rules mattered, so after trying for a while I got my test rig to segfault as well.

The bug is connected to using "by_rule" tracking in thresholds in signatures.

When a new signature with "by_rule" tracking is parsed th_entry is resized to signature number plus one using ThresholdHashRealloc(). This ensures that the "buffer" is large enough to hold state for every rule that use "by_rule" tracking in the ruleset. The issue is that the rules are reordered after they are parsed, and then all the rules are looped over and assigned new signature numbers based on the new order! Because of this, a buffer overflow could occur if we are unlucky enough that a signature with "by_rule" tracking has been given a signature number greater than the size of th_entry after the reordering, and that this rule triggers.

I'm suggesting to fix this by allocating th_entry after all the signatures have been parsed and loaded to ensure that it is large enough to hold all the entries needed.

Program terminated with signal SIGSEGV, Segmentation fault.

```
#0 0x000000000051b381 in ThresholdHandlePacket (p=p@entry=0x7fb0080f3960, lookup_tsh=0x51, new_tsh=new_tsh@entry=0x7fb016c316e0, td=td@entry=0x14adedf0, sid=9800979, gid=1, pa=0x7fb0080f3b18)
   at detect-engine-threshold.c:415
415>---- if (TIMEVAL_DIFF_SEC(p->ts, lookup_tsh->tv1) < td->seconds) {
```

**Related issues:**

- Has duplicate Bug #4514: Suricata 6.0.2 segfault
- Copied to Bug #4518: Buffer overflow in "by_rule" threshold context
- Copied to Bug #4519: Buffer overflow in "by_rule" threshold context

**History**

- **#1 - 05/31/2021 03:40 PM - Philippe Antoine**
  Just adding the stack trace

```
==16985==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x60200002bf68 at pc 0x00010f057b66 bp 0x700000266828 sp 0x700000266828
READ of size 8 at 0x60200002bf68 thread T3
  #0 0x10f057b65 in ThresholdHandlePacketRule detect-engine-threshold.c:580
  #1 0x10f05706e in PacketAlertThreshold detect-engine-threshold.c:639
  #2 0x10f013028 in PacketAlertHandle detect-engine-alert.c:116
  #3 0x10f01259c in PacketAlertFinalize detect-engine-alert.c:260
  #4 0x10efcfb2d in DetectRunPostRules detect.c:944
  #5 0x10efc1d2 in DetectRun detect.c:141
  #6 0x10efcb727 in Detect detect.c:1672
  #7 0x0f0f11e54 in FlowWorker flow-worker.c:540
  #8 0x0f0f128b2e in TmThreadsSlotVarRun tm-threads.c:117
  #9 0x0f0f12c03 in TmThreadsSlotVar tm-threads.c:452
  #10 0x7ff8500e660 in _pthread_body (libsystem_pthread.dylib:libx86_64+0x3660)
  #11 0x7ff8500e50c in _pthread_start (libsystem_pthread.dylib:libx86_64+0x350c)
  #12 0x7ff8500e6f8 in thread_start (libsystem_pthread.dylib:libx86_64+0x2bf8)
```
#2 - 06/07/2021 06:41 AM - Victor Julien
- Status changed from New to Closed
- Target version set to 7.0rc1
- Label Needs backport to 5.0, Needs backport to 6.0 added

https://github.com/OISF/suricata/pull/6151

#3 - 06/07/2021 12:01 PM - Jeff Lucovsky
- Copied to Bug #4518: Buffer overflow in "by_rule" threshold context added

#4 - 06/07/2021 12:01 PM - Jeff Lucovsky
- Copied to Bug #4519: Buffer overflow in "by_rule" threshold context added

#5 - 07/02/2021 12:53 PM - Victor Julien
- Has duplicate Bug #4514: Suricata 6.0.2 segfault added