Suricata - Bug #632

tcp.reassembly_memuse incorrectly reported

11/16/2012 01:47 PM - Ludovico Cavedon

| Status:       | Closed
| Priority:     | Normal
| Assignee:     | category
| Category:     | Difficulty:
| Target version: | Label:
| Affected Versions: | Difficulty:
| Effort:       | Label:

Description

tcp.reassembly_memuse seems to be incorrectly multiplied by the number of detector threads.

The static variable ra_memuse holds the amount of memory used by the TcpSegmentPool, which is shared across all threads. This global variable is used to set the value of the rtv->counter_tcp_reass_memuse counter (in StreamTcpReassembleMemuseCounter()), which is in the per-thread context.

However, when tcp.reassembly_memuse is printed on stats.log it is summed across all threads, therefore reporting the actual value of ra_memuse multiplied by the number of detector threads.

Verified on version suricata 1.3.4.

History

#1 - 12/07/2012 09:10 AM - Victor Julien
  - Status changed from New to Assigned
  - Assignee set to Anoop Saldanha

#2 - 10/26/2013 10:21 AM - Victor Julien
  - Target version set to TBD

#3 - 11/12/2014 01:12 PM - Ken Steele
  Now (suricat 2.0.x) stats.log reports the value of ra_memuse for each thread. So it is no longer summed incorrectly, but it no implies that threads * ra_memuse amount of memory is being used, which is wrong.

#4 - 06/15/2016 05:55 AM - Victor Julien
  - Status changed from Assigned to Closed
  - Assignee deleted (Anoop Saldanha)
  - Target version deleted (TBD)

This has been addressed in 3.0 I believe.