Suricata - Optimization #5127

Bug # 5120 (Closed): alerts: 5.0.8/6.0.4 count noalert sigs towards built-in alert limit (6.0.x backport)
alerts: use alert queing in DetectEngineThreadCtx (6.0.x backport)

02/18/2022 01:56 PM - Jeff Lucovsky

Status: Closed  Priority: Normal
Assignee: Juliana Fajardini Reichow  Category:
Target version: 6.0.6  Effort:
Difficulty:

Description
Currently each alert is written directly to Packet::alerts during rule evaluation. Then at the end of the detection run for a packet, PacketAlertFinalize removes entries again, when applying thresholding, suppression and noalert. This leads to the issue in #4941 but is often also not very efficient esp when there are multiple rules to remove.

The idea of this ticket is to use a per DetectEngineThreadCtx specific queue of some sort to store the alert "candidates" and have PacketAlertFinalize only write the final alerts to the Packet structure.

Related issues:
Copied from Optimization #4943: alerts: use alert queing in DetectEngineThread... Closed

History
#1 - 02/18/2022 01:56 PM - Jeff Lucovsky
- Copied from Optimization #4943: alerts: use alert queing in DetectEngineThreadCtx added

#2 - 05/02/2022 11:38 PM - Juliana Fajardini Reichow
- Status changed from New to In Progress
- Assignee changed from OISF Dev to Juliana Fajardini Reichow
- Target version set to 6.0.6

#3 - 05/02/2022 11:59 PM - Juliana Fajardini Reichow
- Status changed from In Progress to In Review

Backports PR for review: https://github.com/OISF/suricata/pull/7366

#4 - 05/04/2022 09:01 PM - Juliana Fajardini Reichow
- Status changed from In Review to Closed

Merged PR: https://github.com/OISF/suricata/pull/7369

#5 - 05/10/2022 12:37 PM - Juliana Fajardini Reichow
- Subject changed from alerts: use alert queing in DetectEngineThreadCtx to alerts: use alert queing in DetectEngineThreadCtx (6.0.x backport)

#6 - 06/03/2022 09:52 AM - Victor Julien
- Parent task changed from #5124 to #5120