Suricata - Optimization #4066
Add a PASS_IF_NULL macro to the FAIL/PASS API
10/12/2020 09:46 AM - Shivani Bhardwaj

Status: Closed
Priority: Normal
Assignee: Janani Ramjee
Category: 
Target version: 
Effort: 
Difficulty: 
Label: Beginner, C, Outreachy

Description
src/util-unittests.h contains a few macro definitions that are being used in the new tests. There is PASS, PASS_IF for the success paths. Add a 'PASS_IF_NULL' macro. Its function should be as the name suggests. It should pass if the value provided to it is NULL.

Related issues:
Related to Optimization #4795: Remove PASS_IF macro from the FAIL/PASS API

History
#1 - 10/12/2020 10:28 AM - Janani Ramjee
- Subject changed from Add a FAIL_IF_NULL macro to the FAIL/PASS API to Add a PASS_IF_NULL macro to the FAIL/PASS API
- Assignee changed from Community Ticket to Janani Ramjee

#2 - 10/19/2020 12:53 PM - Janani Ramjee
- Status changed from New to In Review

#3 - 10/29/2021 11:07 PM - Juliana Fajardini Reichow
- Related to Optimization #4795: Remove PASS_IF macro from the FAIL/PASS API added

#4 - 10/29/2021 11:12 PM - Juliana Fajardini Reichow
This idea was discontinued after some further discussion (see https://github.com/OISF/suricata/pull/5485).

One of the reasons is that having PASS_IF macros could end up breaking the guideline of having PASS to be the last thing called in a unit tests (as the PASS_IF macros could end up implicitly calling FAIL, internally).

The other is because it requires reverting the logic of the FAIL calls in order to be written, therefore making the process more convoluted.

#5 - 11/19/2021 11:59 AM - Juliana Fajardini Reichow
- Status changed from In Review to Closed

Commit https://github.com/OISF/suricata/pull/6582/commits/d581fc82b18219d9c5bef8f0a0c0c42545c7b054, merged in https://github.com/OISF/suricata/pull/6582, removed the PASS_IF macro, as per the decision shared in the related issue.

Closing this one, then...