**Suricata - Bug #3846**

**Infinite loop if the sniffing interface temporarily goes down**

07/27/2020 03:10 PM - Dan Moinescu

<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>Juliana Fajardini Reichow</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></td>
</tr>
<tr>
<td>Difficulty:</td>
<td></td>
</tr>
<tr>
<td>Effort:</td>
<td></td>
</tr>
<tr>
<td>Label:</td>
<td>Needs backport to 6.0</td>
</tr>
</tbody>
</table>

**Description**

How to reproduce: start suricata in live-pcap mode. Bring the sniffing interface down with ip link set(interface_name) down, then bring it up again.

Suricata doesn't exit, but it no longer produces events either.

The problem appears to be that function ReceivePcapLoop keeps calling PcapTryReopen in a loop, which in turn keeps returning PCAP_ERROR_ACTIVATED, resulting in an infinite loop.

The following patch fixes it:

```c
--- a/src/source-pcap.c
+++ b/src/source-pcap.c
@@ -250,6 +250,7 @@ TmEcode ReceivePcapLoop(ThreadVars *tv, void *data, void *slot)
     "the operation can't be performed on already activated captures" - permanent error, exit
+    /*
     -     
+        dbreak = 1;
+        retval = TM_ECODE_FAILED;
+        break;
+    
+    
+    } while (r < 0); 
+    if (dbreak) {
+        break;
+    }
@@ -292,7 +302,7 @@ TmEcode ReceivePcapLoop(ThreadVars *tv, void *data, void *slot)

PcapDumpCounters(ptv);
StatsSyncCountersIfSignalled(tv);
-   SCReturnInt(TM_ECODE_OK);
+   SCReturnInt(retval);
```
Related issues:
Copied to Bug #5436: Infinite loop if the sniffing interface temporarily goes...  New

History

#1 - 07/28/2020 06:21 PM - Roland Fischer
The proposed patch is one way of fixing it but it'll exit Suricata.

The question is whether Suricata should try to recover in this use case, or if it should exit and leave it up to some external process control, e.g. systemd if you fancy that, to restart the whole thing. Each solution has its tos and fros.

From what I can tell, retry behaviour was added in https://github.com/OISF/suricata/commit/fb36c0af126883f91f7bd45ad162323a9efaf031 quite a bit ago with the goal of “Prior to this patch, a suricata listening to an interface was leaving when the interface goes down. This patch modifies the behaviour to automatically reconnect.”.

This probably points more towards fixing up PcapTryReopen() to handle this use case as well, but the suricata devs would better know their preference.

#2 - 07/28/2020 06:24 PM - Roland Fischer
Fixing PcapTryReopen() would probably require to pcap_close() the pcap handle in this case and re-create it similar to what is done in ReceivePcapThreadInit().

#3 - 09/08/2021 06:35 PM - Philippe Antoine
- Status changed from New to In Review
- Target version set to 7.0rc1

https://github.com/OISF/suricata/pull/6274

#4 - 06/28/2022 06:49 PM - Juliana Fajardini Reichow
- Status changed from In Review to In Progress
- Assignee set to Juliana Fajardini Reichow

This work had been started already on GH but the contributor, unfortunately, hasn't signed the CLA, so we can't accept their contribution.

Claiming it, so this bug can be resolved.

#5 - 06/28/2022 08:48 PM - Juliana Fajardini Reichow
- Status changed from In Progress to In Review

New PR for review: https://github.com/OISF/suricata/pull/7580

#6 - 06/29/2022 08:28 PM - Juliana Fajardini Reichow
- Status changed from In Review to Closed

Merged PR: https://github.com/OISF/suricata/pull/7580

#7 - 07/04/2022 02:11 PM - Juliana Fajardini Reichow
- Label Needs backport to 6.0 added

#8 - 07/04/2022 02:12 PM - Juliana Fajardini Reichow
- Copied to Bug #5436: Infinite loop if the sniffing interface temporarily goes down (6.0.x backports) added