### Suricata - Bug #2230

#### valgrind memcheck - 4.0.0-dev (rev 1180687)

10/12/2017 03:09 PM - Peter Manev

<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>Peter Manev</td>
</tr>
<tr>
<td>Category</td>
<td></td>
</tr>
<tr>
<td>Target version</td>
<td>4.0.2/4.0.3</td>
</tr>
<tr>
<td>Affected Versions</td>
<td>4.0.2/4.0.3</td>
</tr>
<tr>
<td>Difficulty</td>
<td></td>
</tr>
</tbody>
</table>

**Description**

```
==8084== Syscall param ioctl(SIOCETHTOOL) points to uninitialised byte(s)
==8084==    at 0x7C28F07: ioctl (syscall-template.S:84)
==8084==    by 0x5AE96E: GetEthtoolValue (util-ioctl.c:286)
==8084==    by 0x5AF3B4: DisableIfaceOffloadingLinux (util-ioctl.c:434)
==8084==    by 0x558BE6: ParseAFPConfig (runmode-af-packet.c:468)
==8084==    by 0x55E9BC1: RunModeSetLiveCaptureWorkers (util-runmodes.c:373)
==8084==    by 0x55B329: RunModeIdsAFPWorkers (runmode-af-packet.c:693)
==8084==    by 0x563D97: RunModeDispatch (runmodes.c:384)
==8084==    by 0x413512: main (suricata.c:2909)
==8084==  Address 0xffeffcd85 is on thread 1's stack
==8084==  in frame #1, created by GetEthtoolValue (util-ioctl.c:273)
```

```
==8084== Syscall param ioctl(SIOCETHTOOL) points to uninitialised byte(s)
==8084==    at 0x7C28F07: ioctl (syscall-template.S:84)
==8084==    by 0x5AE96E: GetEthtoolValue (util-ioctl.c:286)
==8084==    by 0x5AF46C: DisableIfaceOffloadingLinux (util-ioctl.c:441)
==8084==    by 0x558BE6: ParseAFPConfig (runmode-af-packet.c:468)
==8084==    by 0x55E9BC1: RunModeSetLiveCaptureWorkers (util-runmodes.c:373)
==8084==    by 0x55B329: RunModeIdsAFPWorkers (runmode-af-packet.c:693)
==8084==    by 0x563D97: RunModeDispatch (runmodes.c:384)
==8084==    by 0x413512: main (suricata.c:2909)
==8084==  Address 0xffeffcd85 is on thread 1's stack
==8084==  in frame #1, created by GetEthtoolValue (util-ioctl.c:273)
```

```
==8084== Syscall param ioctl(SIOCETHTOOL) points to uninitialised byte(s)
==8084==    at 0x7C28F07: ioctl (syscall-template.S:84)
==8084==    by 0x5AE96E: GetEthtoolValue (util-ioctl.c:286)
==8084==    by 0x5AF52A: DisableIfaceOffloadingLinux (util-ioctl.c:448)
==8084==    by 0x558BE6: ParseAFPConfig (runmode-af-packet.c:468)
==8084==    by 0x55E9BC1: RunModeSetLiveCaptureWorkers (util-runmodes.c:373)
==8084==    by 0x55B329: RunModeIdsAFPWorkers (runmode-af-packet.c:693)
==8084==    by 0x563D97: RunModeDispatch (runmodes.c:384)
==8084==    by 0x413512: main (suricata.c:2909)
==8084==  Address 0xffeffcd85 is on thread 1's stack
==8084==  in frame #1, created by GetEthtoolValue (util-ioctl.c:273)
```

```
==8084== Syscall param ioctl(SIOCETHTOOL) points to uninitialised byte(s)
```

---

09/02/2022

1/4
Details attached.

History

#1 - 10/12/2017 04:39 PM - Andreas Herz
- Assignee set to OISF Dev
- Target version set to TBD

#2 - 10/13/2017 02:25 AM - Victor Julien
I don't think we can fix these. Can you see if you can generate suppressions for it? Valgrind as a way to output warnings in their suppression format. Think it's in the -h or manpage how to do it.

#3 - 10/19/2017 04:37 PM - Peter Manev
Seems we have those cases (with slight variations) with AFP and pcap - so would investigate a bit further for completeness then will summarize all suppressions and report back.

#4 - 10/27/2017 02:43 AM - Victor Julien
- Status changed from New to Assigned
- Assignee changed from OISF Dev to Peter Manev
- Target version changed from TBD to 70

You can add all the unique ones you find to this ticket. If you add suppressions, please update the suppressions file in the source and issue a pull request.

#5 - 10/27/2017 02:55 AM - Peter Manev
These are the suppressions I have confirmed so far.

```c
{
    AFP - NIC GET offloading
    Memcheck:Param
    ioctl(SIOCETHTOOL)
    fun:ioctl
    fun:GetEthtoolValue
    fun:DisableIfaceOffloadingLinux
    fun:DisableIfaceOffloading
    fun:ParseAEFPConfig
    fun:RunModeSetLiveCaptureWorkers
    fun:RunModeIdsAFPWorkers
    fun:RunModeDispatch
    fun:main
}
{
    AFP - NIC SET offloading
    Memcheck:Param
    ioctl(SIOCETHTOOL)
    fun:ioctl
    fun:SetEthtoolValue
    fun:DisableIfaceOffloadingLinux
    fun:DisableIfaceOffloading
    fun:ParseAEFPConfig
    fun:RunModeSetLiveCaptureWorkers
    fun:RunModeIdsAFPWorkers
    fun:RunModeDispatch
    fun:main
}
```
Not sure if valgrind supports wildcards, if it does it would be good to use them here.

#6 - 10/27/2017 03:06 AM - Victor Julien
Memcheck:Cond
  fun:DisableIfaceOffloadingLinux
  fun:DisableIfaceOffloading

#7 - 11/04/2017 03:59 AM - Peter Manev
Done by - https://github.com/OISF/suricata/pull/2963

#8 - 11/06/2017 11:13 AM - Victor Julien
  - Status changed from Assigned to Closed
  - Target version changed from 70 to 4.0.2/4.0.3

Merged pr/2963, thanks Peter.

Files

<table>
<thead>
<tr>
<th>File</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>bugreport-valgrind</td>
<td>43.8 KB</td>
<td>10/12/2017</td>
<td>Peter Manev</td>
</tr>
</tbody>
</table>