Suricata - Bug #373

suricata 1.1 crash for Illegal instruction

11/11/2011 02:05 AM - delta yeh

<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></td>
</tr>
<tr>
<td>Category:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
<tr>
<td>Affected Versions:</td>
<td></td>
</tr>
<tr>
<td>Difficulty:</td>
<td></td>
</tr>
<tr>
<td>Label:</td>
<td></td>
</tr>
</tbody>
</table>

**Description**

set args -c /etc/suricata/suricata.yaml --pfring
run

[Thread debugging using libthread_db enabled]
[New process 22627]
[New Thread -1219356096 (LWP 22627)]

Program received signal SIGILL, Illegal instruction.
[Switching to Thread -1219356096 (LWP 22627)]
0x0816414c in DefragInit () at defrag.c:1238
1238     defrag.c: No such file or directory.
 in defrag.c
(gdb) bt
#0 0x0816414c in DefragInit () at defrag.c:1238
 #1 0x0804d7ff in main (argc=4, argv=0xbfe69504) at suricata.c:1161
 (gdb) frame 0
#0 0x0816414c in DefragInit () at defrag.c:1238
1238     in defrag.c
==>
Line 1238 is :    defrag_hash_rand = (int)( DEFAULT_DEFRAG_HASH_SIZE * (rand_r(&seed) / RAND_MAX + 1.0));

(gdb) info args
No arguments.
(gdb) info locals
seed = 4059467122

FUNCTION = "DefragInit"
(gdb) call rand_r(&seed)
$1 = 1408555743
(gdb) call rand_r(&seed)
$2 = 1238339832
(gdb) call rand_r(&seed)
$3 = 1866603133
(gdb) call rand_r(&seed)
$4 = 1128750617

1. uname -a
   Linux test 2.6.39.4 #21 SMP Wed Oct 26 21:30:22 CST 2011 i686 GNU/Linux
2. cat /proc/cpuinfo
   processor : 0
   vendor_id : GenuineIntel
cpu family : 6
   model       : 13
   model name  : Intel(R) Pentium(R) M processor 2.13GHz
   stepping : 8
cpu MHz     : 2127.573
   cache size : 2048 KB
fdiv_bug : no
hit_bug : no
f00f_bug : no
coma_bug : no
fpu : yes
fpu_exception : yes
cpuid level : 2
wp : yes
flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov cflush dts acpi mmx fxsr sse sse2 ss tm pbe nx
up bts est tm2
bogomips : 4255.14
clflush size : 64
cache_alignment : 64
address sizes : 32 bits physical, 32 bits virtual
power management:

History

#1 - 11/11/2011 02:51 AM - Victor Julien

Did you compile your Suricata on this same box? If not, the --march=native option we use by default with gcc might be the issue. Try adding --disable-gccmarch-native to ./configure.

#2 - 11/11/2011 06:04 AM - delta yeh

You are right, add --disable-gccmarch-native fix this issue.

I think we should not turn on --march=native on, because

in most time, suri is not run on the box where suri is compiled.

#3 - 11/11/2011 06:54 AM - Victor Julien

- Status changed from New to Closed

I disagree. We want the best out of the box performance. Distributions and others that distribute binaries can use the --disable-gccmarch-native option, as they are doing already.