Suricata - Bug #3907

http2 rust - 'index out of bounds: the len is 2 but the index is 63'

09/06/2020 06:38 AM - Peter Manev

Status: In Review
Priority: Normal
Assignee: Philippe Antoine
Category: Target version: 6.0.0rc1
Affected Versions: 6.0.0beta1, git master

Description
Affected versions: gitmaster only
Running normal/usual AFPv3.

thread '<unnamed>' panicked at 'index out of bounds: the len is 2 but the index is 63', src/http2/parser.rs:298:23

note: run with `RUST_BACKTRACE=1` environment variable to display a backtrace
fatal runtime error: failed to initiate panic, error 5
Aborted (core dumped)

Suri info:

/usr/local/bin/suricata --build-info
This is Suricata version 6.0.0-dev (1be294718 2020-09-05)
Features: PCAP_SET_BUFF AF_PACKET HAVE_PACKET_FANOUT LIBCAP_NG LIBNET1.1 HAVE_HTP_URI_NORMALIZE_HOSTNAME_OK PCRE_JIT HAVE_NSS HAVE LUA HAVE LUAJIT HAVE_LIBJANSSON TLS TLS_C11 MAGIC RUST
SIMD support: SSE_4_2 SSE_4_1 SSE_3
Atomic intrinsics: 1 2 4 8 16 byte(s)
64-bits, Little-endian architecture
GCC version 4.2.1 Compatible Clang 7.0.1 (tags/RELEASE_701/final), C version 201112
compiled with _FORTIFY_SOURCE=0
L1 cache line size (CLS)=64
thread local storage method: _Thread_local
compiled with LibHTP v0.5.33, linked against LibHTP v0.5.33

Suricata Configuration:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>AF_PACKET support</td>
<td>yes</td>
</tr>
<tr>
<td>eBPF support</td>
<td>yes</td>
</tr>
<tr>
<td>XDP support</td>
<td>yes</td>
</tr>
<tr>
<td>PF_RING support</td>
<td>no</td>
</tr>
<tr>
<td>NFQueue support</td>
<td>no</td>
</tr>
<tr>
<td>NFLog support</td>
<td>no</td>
</tr>
<tr>
<td>IPFW support</td>
<td>no</td>
</tr>
<tr>
<td>Netmap support</td>
<td>no</td>
</tr>
<tr>
<td>DAG enabled</td>
<td>no</td>
</tr>
<tr>
<td>Napatech enabled</td>
<td>no</td>
</tr>
<tr>
<td>WinDivert enabled</td>
<td>no</td>
</tr>
<tr>
<td>Unix socket enabled</td>
<td>yes</td>
</tr>
<tr>
<td>Detection enabled</td>
<td>yes</td>
</tr>
<tr>
<td>Libmagic support</td>
<td>yes</td>
</tr>
<tr>
<td>libnss support</td>
<td>yes</td>
</tr>
<tr>
<td>libnspr support</td>
<td>yes</td>
</tr>
<tr>
<td>libjanssson support</td>
<td>yes</td>
</tr>
<tr>
<td>hiredis support</td>
<td>no</td>
</tr>
<tr>
<td>hiredis async with libevent</td>
<td>no</td>
</tr>
<tr>
<td>Prelude support</td>
<td>no</td>
</tr>
</tbody>
</table>
PCRE jit: yes
LUA support: yes, through luajit
libluajit: yes
GeoIP2 support: yes
Non-bundled htp: no
Old barnyard2 support: yes
Hyperscan support: yes
Libnet support: yes
liblz4 support: yes

Rust support: yes
Rust strict mode: yes
Rust compiler path: /root/.cargo/bin/rustc
Rust compiler version: rustc 1.46.0 (0448afe3 2020-08-24)
Cargo path: /root/.cargo/bin/cargo
Cargo version: cargo 1.46.0 (149022b1d 2020-07-17)
Cargo vendor: yes

Python support: yes
Python path: /usr/bin/python3
Python distutils: yes
Python yaml: yes
Install suricatactl: yes
Install suricatasc: yes
Install suricata-update: not bundled

Profiling enabled: no
Profiling locks enabled: no

Plugin support (experimental): yes

Development settings:
Coccinelle / spatch: no
Unit tests enabled: no
Debug output enabled: no
Debug validation enabled: no

Generic build parameters:
Installation prefix: /usr/local
Configuration directory: /usr/local/etc/suricata/
Log directory: /usr/local/var/log/suricata/

--prefix /usr/local
--sysconfdir /usr/local/etc
--localstatedir /usr/local/var
--datarootdir /usr/local/share

Host: x86_64-pc-linux-gnu
Compiler: clang (exec name) / g++ (real)
GCC Protect enabled: no
GCC march native enabled: yes
GCC Profile enabled: no
Position Independent Executable enabled: no

SECCFLAGS -I/usr/include

#2 0x00000000000ec3be7 in std::sys::unix::abort_internal () at src/libstd/sys/unix/mod.rs:167
#3 0x00000000000eb2a05 in std::sys_common::util1::abort () at src/libstd/sys_common/util1.rs:19
#4 0x00000000000e0a03e in std::panicking::rust_panic () at src/libstd/panicking.rs:577
#5 0x00000000000e07ca5 in std::panicking::rust_panic_with_hook () at src/libstd/panicking.rs:545
#6 0x00000000000e0b33b in rust_begin_unwind () at src/libstd/panicking.rs:437
#7 0x00000000000ebb481 in core::panicking::panic_fmt () at src/libcore/panicking.rs:85
#8 0x00000000000ebb442 in core::panicking::panic_bounds_check () at src/libcore/panicking.rs:62
History

#1 - 09/08/2020 02:03 PM - Victor Julien
- Status changed from New to Assigned
- Assignee set to Philippe Antoine
- Target version set to 6.0.0rc1
- Affected Versions 6.0.0beta1, git master added
- Affected Versions deleted (6.0.0rc1)

#2 - 09/08/2020 02:28 PM - Philippe Antoine
- Status changed from Assigned to In Review

https://github.com/OISF/suricata/pull/5383

Thanks for finding this

I should run some fuzz testing, as this bug was pretty shallow...