Suricata - Bug #1076
Tcp assembly sliding window left_edge could be unexpected value.
01/04/2014 02:27 AM - Song Liu

Status: Closed
Priority: Normal
Assignee: Song Liu
Category:
Target version:
Affected Versions:
Effort:
Description
In stream-tcp-reassemble.c, the function StreamTcpReassembleInlineRaw() has the following code:

/* determine the left edge and right edge */
uint32_t right_edge = TCP_GET_SEQ(p) + p->payload_len;
uint32_t left_edge = right_edge - chunk_size;
......
......
left_edge = (ra_base_seq + 1) - chunk_size;

In theory, (right_edge - chunk_size) could be less than zero. This will cause uint32_t left_edge to be an unexpected value.
Therefore, it will be necessary to compare right_edge and chunk_size before doing (right_edge - chunk_size).

History
#1 - 01/05/2014 09:34 PM - Song Liu
I debugged it, and it's like SEG_GT and SEG_LEQ already convert unsigned integer to signed integer and handle this case and would not cause problem.

#2 - 01/10/2014 12:59 AM - Song Liu
- Status changed from New to Closed

#3 - 03/14/2014 07:03 AM - Victor Julien
- Target version deleted (1.4.8)