

## DragonFlyBSD - Bug #2372

### segfault correct detection failure

05/20/2012 03:18 AM - marino

<b>Status:</b>	New	<b>Start date:</b>	05/20/2012
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			
<b>Description</b>			
<p>The package devel/libsigsegv fails its final diagnostic test on DragonFly. For information, this does pass on FreeBSD 9.</p> <p>To repeat:</p> <pre>&gt; cd /usr/pkgsrc/devel/libsigsegv &gt; bmake &gt; cd \${WRKOBJDIR}/devel/libsigsegv/work/libsigsegv-2.10/ &gt; gmake check</pre> <p>The output of the fifth test (out of five) should be:</p> <pre>  Starting recursion pass 1.   Stack overflow 1 caught.   Starting recursion pass 2.   Stack overflow 2 caught.   Segmentation violation correctly detected.   Segmentation violation correctly detected.   Test passed.   PASS: stackoverflow2</pre> <p>The actual output is:</p> <pre>  Starting recursion pass 1.   Stack overflow 1 caught.   Starting recursion pass 2.   Stack overflow 2 caught.   Segmentation violation misdetected as stack overflow.   Test passed.   FAIL: stackoverflow2</pre> <p>The code of the test is here: <a href="http://fossies.org/dox/libsigsegv-2.10/stackoverflow2_8c_source.html">http://fossies.org/dox/libsigsegv-2.10/stackoverflow2_8c_source.html</a></p> <p>What seems to be happening is that when the stack is exhausted, accessing an illegal memory location triggers the stack overflow handler before the sigsegv handler. I think it's a DragonFly bug.</p>			