

[erlang-questions] can nodes fail/recover too fast to be seen?

Per Hedeland <per@hedeland.org>

Fri Jul 5 21:23:35 CEST 2013

- Previous message: [\[erlang-questions\] can nodes fail/recover too fast to be seen?](#)
 - Next message: [\[erlang-questions\] can nodes fail/recover too fast to be seen?](#)
 - Messages sorted by: [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#)
-

Gleb Peregud <gleber.p@gmail.com> wrote:

>

>If it was an intermittent network issue, TCP can mask the problem and

>Erlang would never know about it.

This is a completely different (non-)problem, the question was about a node failing and recovering. If both nodes keep running and there is an "intermittent network issue" that "TCP can mask", there *is* no problem - TCP deals with it, that's arguably its main purpose in life.

>On Fri, Jul 5, 2013 at 5:24 PM, Sergej Jurecko <sergej.jurecko@gmail.com> wrote:

>> Well yes erlang does reconnect, but you still get a nodedown/nodeup message no?

Yes, and 'DOWN' for the process(es). But in both cases only if you monitor (or link => exit signal).

>> On Jul 5, 2013, at 5:22 PM, Tim Watson wrote:

>>

>>> As i understand it, this can and does happen, because erlang does automatic reconnect in order to provide reliable communications.

No.

>>> On 5 Jul 2013, at 15:49, Jonathan Leivent <jleivent@comcast.net> wrote:

>>>

>>>> In Erlang, is it possible for a monitored node to fail and recover so quickly that nodes monitoring it won't detect the failure?

No. The TCP connection to the old node instance cannot be used for communication with the new node instance, i.e. there is no way that communication with the new node instance can be established without the local VM generating node_down/'DOWN'/exit messages for the old instance.

>>>> *Or, is there some kind of internal persistent state that prevents this?*

This is where it potentially gets interesting - i.e. assuming **no** monitoring or linking - and that's where the "creation" part of a node identifier comes into play. If a distributed node restarts, it will get a new "creation" value courtesy of epmd, and any any pid() values referring to the old node instance will be invalid.

However the "creation" is only 2 bits, so if a node restarts frequently, old pid() values may become "valid" once again, i.e. referring to a new, different process. Is it a problem in practice, outside academic research papers? I've never heard of such a case. If you really have this problem, you should probably look at why the node keeps restarting...

--Per Hedeland

-
- Previous message: [\[erlang-questions\] can nodes fail/recover too fast to be seen?](#)
 - Next message: [\[erlang-questions\] can nodes fail/recover too fast to be seen?](#)
 - **Messages sorted by:** [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#)

[More information about the erlang-questions mailing list](#)