

More Hanging Jrun and Networking

Posted At : June 2, 2006 6:03 PM | Posted By : Mark Kruger

Related Categories: Coldfusion Troubleshooting

For those of you following this issue I've had a breakthrough - an epiphany brought to me by the inestimable Cameron Childress. Cameron was musing over a post I made to a mutual list we both monitor. In the post I laid out the 3 instances where network issues (particularly auto-sensing NICs or Switches) had caused Jrun to hang. I think he must be a better googler than I am because he came up with this link to an article on setting up a [Win2003 server cluster](#) that contains some excellent information regarding our issue. Among the instructions on setting up a multi-home network for clustering was this item:

NOTE: This post is a follow up of these 2 posts:

[Queued Requests Hanging Coldfusion](#)
[Network Issues and Hanging Threads](#)

The autosense mode of some 10/100 Ethernet adapters might automatically detect the speed of the connected network. However, during the detection process, packets can't be handled by the adapter and must be queued. Some adapters might inadvertently trigger the auto detection process to reoccur intermittently. As a result, communications are queued and delayed. Delays of this nature might cause cluster nodes not to receive critical packets in a timely fashion and might cause premature failover of cluster resources. This is why it's important to manually set the correct speed and duplex to avoid any possible autosense-related problems.

Is this the fix?

Sure enough - that sounds like my issue. Packets are queuing as low level protocols go about optimizing the connection periodically. I would add that it is not only NICs with this issue. Many unmanaged switches are auto-sensing, and even managed switches might be set to auto-sense by default. That makes this a magic bullet for anyone afflicted with my problem (I mean my *network problem*, not the issue with Jello). The secret for connectivity would be to ensure point to point *static configuration* for devices along the path between the web server and the database server, JMS server or other external network resource.

Another Tidbit

In my research I also ran across this [tech note](#) from Microsoft related to it's own JDBC driver (which I believe is a repackaged version of the Data Direct Driver). The note indicates that using a "setQueryTimeout()" function can cause thread issues because an additional thread is spawned to monitor the query - causing memory loss and death (much like daytime TV) in a high load environment. This got me to wondering if this method is implemented in the drivers that ship with Coldfusion. If it is, how is it implemented? Cameron had suggested using the "replyTimeout" value of the JDBC driver to cut down on network issues. I wonder if that value would result in more threads rather than less.

Anyway, I think I now have a procedure to cover to eliminate this as a potential issue.