

Re: Too much multicasting in Linux

Source: <http://linux.derkeiler.com/Newsgroups/comp.os.linux.networking/2004-04/0258.html>

From: Juanjo Aparicio Jara (*bnt_at_fastpace.org*)

Date: 04/07/04

Date: Wed, 7 Apr 2004 12:11:31 +0000 (UTC)

That is what I expected. However, data is being received by sockets that didn't join the multicast address where the data is being sent. That is, data sent to 224.0.1.84 is being received by a socket that joined ONLY group 224.0.1.85. I assume this is not the expected behaviour.

I don't think Java is the culprit. As stated before, code in C is doing the same.

A previous poster suggested that the fact that the sockets are tied to the interface 0.0.0.0 causes them to receive all data. If the solution means to bind the sockets to different interfaces I guess they will not receive the data, which is what I want since they have not joined the same multicast group. But then, if I have a socket that joins the SAME multicast group but is tied to a different interface it will not receive the multicast data. The software I'm working on should work on different machines in the same LAN.

Anyone have any more ideas?

Thank you very much to all that responded.

In article <4073b747\$0\$64610\$4a441750@news.euro.net.nl>, Johan Hendriks wrote:

- >
- > *To send multicast messages, you don't have to do anything special.*
- > *Just apply the multicast address and the kernel (network driver) will*
- > *recognize this and generate*
- > *the expected packets.*
- >
- > *Only on the receiving side you must join a multicast group (setsockopt)*
- > *after binding the socket to a port.*
- >
- > *Above applies to a socket implementation. Your JAVA classes should not do*
- > *anything more*
- >
- > *Johan*
- >
- >