

Ethernet Address Caching

Eliminate long-standing bug

Thu, May 9, 2002

The IRM system code that supports IP protocols has always done something that is not quite in keeping with IP standards. When a datagram is received, the 6-byte source hardware address is cached in the IPARP table. This was originally done in order to avoid having to do an ARP request to obtain the hardware address, in the case that a reply is due in response to the message just received. It was especially useful at that time, because if a datagram was to be sent to a node for which no entry existed in the IPARP cache, the datagram was “thrown away,” and an ARP request was issued instead. This meant that network communications with other nodes, for which hardware addresses were already cached, could proceed apace, without waiting for a response from the ARP request. Caching received hardware addresses allowed the system to successfully respond to surprising one-shot requests.

One might say, so what? If one receives a datagram from a node, why would the 6-byte source address not be the same address received in response to an ARP request? In the early days, when networking architecture was simpler, it was ok. But now we have many routers involved tying the various networks together into one internet, and things are much more complicated. Imagine a case where two routers are connected to one subnet of nodes, and one of the routers is configured to pass datagrams in to nodes inside the subnet, while the other is used for passing datagrams to nodes outside the subnet. If the first one cannot pass datagrams outside the network, and the above-mentioned scheme is in place, connectivity will fail. One might think that both routers would be configured to pass outside the subnet properly, but there may be reasons why they are not. The official standard means of supporting IP requires that only an ARP reply be used to fill an ARP cache entry.

In time, a method was devised to get around the problem of datagrams being thrown away in lieu of sending an ARP request. When a surprising message is to be sent to a node for which there is currently no hardware address known in the IPARP table, the message is queued to be sent later, and an ARP request is sent immediately. When an ARP reply is received, this queue is checked, and any messages found therein are queued to the network again, when they will presumably be successful. This design did not hold up communications with other nodes during the time that the ARP request was pending, which was a goal.

Now that this method has been working for a long time now, there have been occasions, not often, in which communications with some node appears to be broken. Knowing about this possible weakness in the IP support, a check of the IPARP entry showed that two different router addresses were in place. Changing the problematic node’s entry to match the other router address resulted in connectivity with that node being miraculously restored.

The code is now changed in the latest version (May 9, 2002) of the system so that only hardware addresses found in ARP replies are used to populate cache entries. Unfortunately, it may be some time before we can be sure that this fixes the problem, because it doesn’t happen very often.

The detailed changes made to the system code were four instances in the SNAP module, in which a NULL pointer was passed in a call to PSNIPARP, rather than a pointer to a hardware address.