



Tech Info Library

IBM 8209/AppleTalk: Support Alert Information

This article last reviewed: 1 February 1991

TOPIC -----

IBM has a relatively new product, model 8209, that is a media access control (MAC) sub-layer bridge. Most likely encountered in a Token-Ring to Token-Ring bridging configuration, it also can be configured for Token Ring to Ethernet. Do you have any details?

DISCUSSION -----

We recently had the opportunity of observing/troubleshooting such an environment involving an 8209, an IBM mainframe on Token Ring supporting TCP/IP, Macintoshes running both Phase 1 and Phase 2 AppleTalk on Ethernet, Macintoshes on Token Ring, and a variety of TCP/IP hosts on Ethernet. This leads to several interesting discoveries.

IBM 8209 and AppleTalk

Ideally, with the 8209 providing Token-Ring-to-Ethernet bridging services, Macintoshes on both transports would see the bridged network as a single logical network. Unfortunately, for a number of reasons, this is not true.

The first problem stems from the 8209's generosity in forwarding all datalink (MAC) broadcasts and multicasts. If you are running ANY Phase 1 EtherTalk, adding an 8209 into the net automatically generates Phase 1 TokenTalk. This is useless traffic, because the TokenTalk drivers expect Phase 2 protocols. As far as we can tell, it is not dangerous traffic--it is safely ignored by all nodes. But it does take up bandwidth, and if the Phase 1 Ethernet is large and has many routers, there can be significant broadcast traffic forwarded onto the Token Ring.

The answer here would be to filter Phase 1 EtherTalk at the 8209.

Token Ring and Ethernet do not agree on the order of bits. So, the 8209 flips the bits of the (MAC) datalink-level addresses when it forwards the packets. Normally, this affects only the destination and source addresses in the packet header. However, address resolution protocols like TCP/IP ARP and AppleTalk's AARP also carry MAC-level addresses as a part of their data portion of the packet.

The 8209 is specially programmed to look for TCP's ARP and to go down inside the data, flipping the bits of the embedded MAC address. Later, when a Token Ring host wants to talk with an Ethernet node, it uses the flipped version of the MAC address, which, when encountered by the 8209, is flipped back and thus delivered correctly.

The 8209 is NOT specially programmed to assist AARP. Thus, the AARP address is stored in the Token Ring AppleTalk node in its "unflipped" form. Then, should it be used later to address a packet, the 8209 will flip the bits when forwarding back to Ethernet. Now, the packet is incorrectly addressed, and cannot be delivered.

The long and short of this is that when using an IBM 8209 to bridge Token Ring and Ethernet, AppleTalk nodes on the ring and Ethernet cannot communicate, because they can never AARP correctly.

Curiously, if the Token Ring is used only as a backbone, and a Macintosh on Ethernet wants to talk to a Macintosh on another Ethernet, with both Ethernets attached by 8209s, this connection will work. In fact, as long as you go through an even number (2,4,6,8...) of 8209 bridges, connection is possible. When there is an odd number, connection will fail. Note that this makes some extremely bizarre network configurations possible, like a system of alternating Token Rings and Ethernets connected by 8209s, where:

- All the Ethernets would have AppleTalk connectivity.
- All the Token Rings would have AppleTalk connectivity.
- The rings and Enets would have AppleTalk connectivity between them.

You may remember that under AppleTalk Phase 2, Apple uses multicast addressing for all protocol "overhead" packets, like RTMPs, NBPs, and ZIPs. You may also recall that Apple has a range of 254 addresses used in Ethernet and 19 (different) addresses in Token Ring available for multicasts.

The fact that we use different ranges of addresses (in addition to the bit flipping described above) ensures that AppleTalk nodes cannot interoperate across an 8209 bridging Ethernet and Token Ring. Just look at the boot process.

A node begins by AARP'ing for a protocol address (multicast). This AARP is forwarded by the 8209, but because the bits are flipped, a response to the AARP probe would be lost on the return trip. So, the node assumes that its protocol address is unique, and then proceeds to check for a router by multicasting (protocol broadcasting) a ZIP GetNetInfo request. If there is an AppleTalk router present on my cable (whether we are on Ethernet or Token Ring), we are okay. The router hears my request, provides the needed information about the cable, and I re-AARP if necessary.

However, if the router is on the other side of the 8209, the multicast is forwarded. Even if the bit-flipping wasn't a problem (say, if IBM were to change the 8209 software to support AARP), it will still not make

connection with the router, because a TokenTalk router expects multicast addresses in a different range than a EtherTalk node can send, and vice versa.

The only answer is to use an AppleTalk router, which manages protocol and zone multicast addresses for each data link appropriately.

In short, AppleTalk and 8209s don't mix. Don't use bridging strategies between AppleTalk Token Ring and Ethernet networks.

There are additional problems with 8209 and AppleTalk described in a separate article "IBM 8209/MacTCP: Support Alert Information."
Copyright 1991 Apple Computer, Inc.

Keywords: <None>

=====

This information is from the Apple Technical Information Library.

19960215 11:05:19.00

Tech Info Library Article Number: 6737