Fermi National Accelerator Laboratory

FERMILAB-Pub-92/16
(NHM-91/002)

HEPnet
NHM TECHNICAL NOTES
Cisco "Proxy ARP" Problem

G. Roediger

Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510

January 1992

Operated by Universities Research Association Inc. under Contract No. DE-AC02-76CHO3000 with the United States Department of Energy

HEPnet

NHM TECHNICAL NOTES

Cisco "Proxy ARP" Problem

Gary Roediger

October, 1991

NHM-91/002

CISCO "PROXY ARP" PROBLEM

Gary Roediger

(October, 1991)

The use of PROXY ARP with some configurations of CISCO routers can cause arp table corruption in CISCO routers. CISCO is aware of this bug and says they are working on it. The symptom of a network experiencing this problem is IP addressed machines sitting on one side of a router will be able to send to machines on the router's other side but will never receive IP data from those same machines.

What is Proxy ARP:

Proxy arp is used in environments that have machines which cannot support IP subnetting in a subnetted network. Reasons why this may exist vary from hiding the physical routing topology from end user systems to not implementing subnetting in the routing software. Proxy arp operation is explained with the aid of the network in Figure 1.

In this example there is an IP class B network consisting of a segmented ethernet topology. Two IP routers are attached to the backbone ethernet (subnet 200).

If Host_1 were to communicate with Host_2 the following scenario would take place. Host_1 would issue an ARP broadcast request for the physical ethernet address of the machine with IP address of 150.200.100.2. Host_2 would respond with its ethernet address. Host_1 will put this ethernet/IP address tuple in an arp table for future reference. Each time Host_1 sends to Host_2 the arp table is consulted and the ethernet address of Host_2 is used for MAC layer addressing.

The use of proxy arp can be demonstrated by a communication between Host_4 and Host_2 who (for reasons unknown) are not aware of the site subnet topology. They see each other as having the same class B network address and thus view themselves in a flat IP class B address space. Host_4 wishing to communicate with Host_2 would do exactly the same as Host_1 did in the previous example. Assuming the same ethernet segment, Host_4 will issue and ARP broadcast request for Host_2. The problem is that there is no ethernet connection without going through [router1]. Host_2 will never see the ARP request. Had Host_4 been configured for subnetting, it would have never issued an ARP request but rather would route its traffic for Host_2 through [router1]. The answer is to configure [router1] to do PROXY ARP.

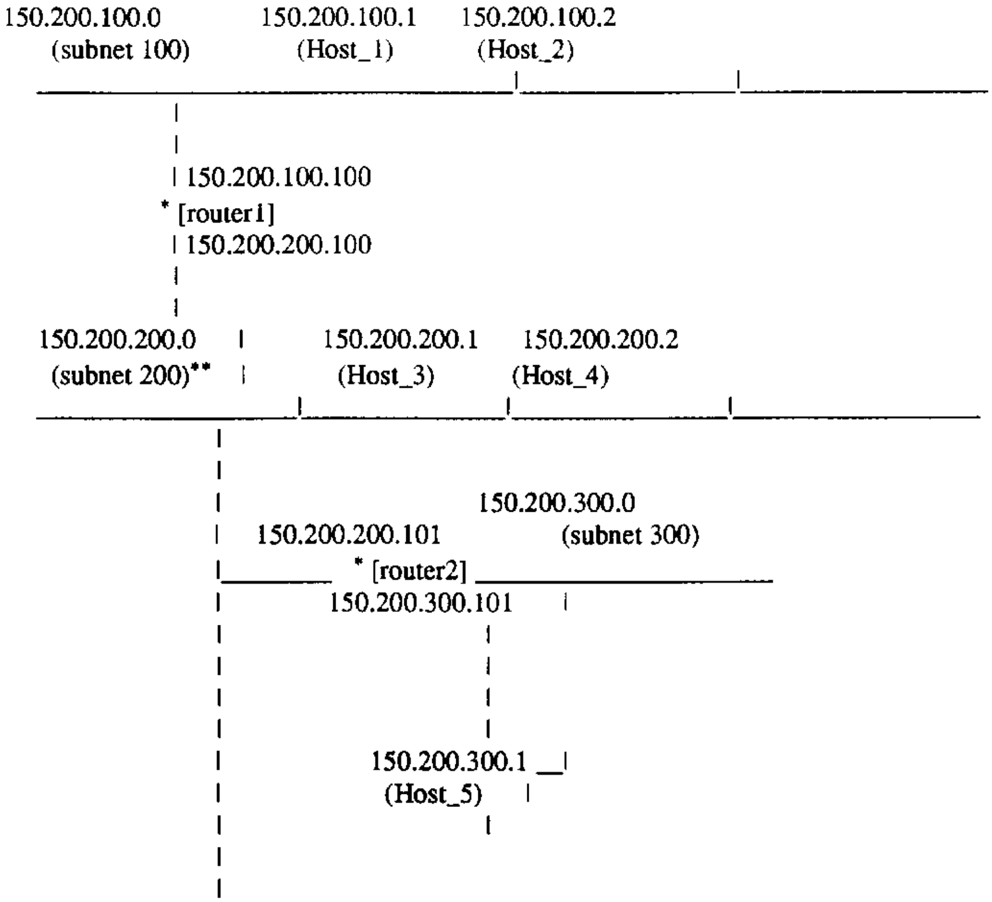

Figure 1
IP class B network 150.200.0.0:

With [router1] configured for Proxy Arp, when Host_4 issues the arp for 150.200.100.2,
[router1] intercepts it and does an arp reply to Host_4 with the ethernet address of [router1]'s 150.200.200.100 interface. Thus every time Host_4 sends to Host_2 the packet contains the ethernet address of [router1] who gets the packet and 'routes' it to Host_2. The same thing happens on the Host_2 to Host_4 communication path.

The Problem with CISCO's Proxy ARP:

Now the problem: [router1] is also set up as the know all router for the backbone ethernet. It has static routes set up for subnets hanging off the backbone (eg. subnet 300, subnet 100). The following scenario causes Host_5 to disappear.

---

* Cisco routers with 2 ethernet interfaces

** Subnet 200 is the backbone ethernet segment

Host_5 soon becomes known (via arp) to it's default [router2] as having its real ethernet address on the subnet 300 port. Thus anything coming from the backbone thru [router2] destined for Host_5 gets the ethernet address of Host_5 put on and sent out subnet 300 interface. Now what happens when Host_3 wants to send to Host_5? Host_3 does exactly what Host_4 did in the previous example when Host_4 sent to Host_2. The problem is that [router1] also does the same thing as in the previous example. It knows about it's static route to subnet 300 and it answers the Host_3 arp request (PROXY ARP) for Host_5's address. Host_3 is not the only one listening to the arp reply from [router1] for the Host_5 ethernet address. So is [router2] - who immediately updates its single arp table and changes its address of Host_5. The new arp table entry in [router2] for Host_5 is changed to be the ethernet address of [router1]'s 150.200.200.100 port and on interface of subnet 200.

Now we have a situation where Host_5 can send out but anything coming from subnet 200 (the backbone) to [router2] for Host_5 gets blocked. The arp table translation in [router2] causes the ethernet address of [router1] to be put on and sent out subnet 200 interface. Thus Host_5 disappears.

Magic at its best.