              Internet Architecture Extensions for Shared Media

Status of This Memo

   This memo provides information for the Internet community. This memo
   does not specify an Internet standard of any kind. Distribution of
   this memo is unlimited.

Abstract

   The original Internet architecture assumed that each network is
   labeled with a single IP network number. This assumption may be
   violated for shared media, including "large public data networks"
   (LPDNs). The architecture still works if this assumption is
   violated, but it does not have a means to prevent multiple host-
   router and router-router hops through the shared medium. This memo
   discusses alternative approaches to extending the Internet
   architecture to eliminate some or all unnecessary hops.

Table of Contents

1. INTRODUCTION

   This memo concerns the implications of shared medium networks for the
   architecture of the TCP/IP protocol suite. General familiarity with
   the TCP/IP architecture and the IP protocol is assumed.

   The Internet architecture is founded upon what was originally called
   the "Catenet model" [PSC81]. Under this model, the Internet
   (originally dubbed "the Catenet") is formed using routers (originally
   called "gateways") to interconnect distinct and perhaps diverse
   networks. An IP host address (more correctly characterized as a
   network interface address) is formed of the pair (net#,host#). Here
   "net#" is a unique IP number assigned to the network (or subnet) to
   which the host is attached, and "host#" identifies the host on that
   network (or subnet).

   The original Internet model made the implicit assumptions that each
   network has a single IP network number and that networks with
   different numbers may interchange packets only through routers.
   These assumptions may be violated for networks implemented using a
   common "shared medium" (SM) at the link layer (LL). For example,
   network managers sometimes configure multiple IP network numbers
   (usually subnet numbers) on a single broadcast-type LAN such as an
   Ethernet. The large (switched) public data networks (LPDNs), such as
   SMDS and B-ISDN, form a potentially more important example of shared
   medium networks. Any two systems connected to the same shared medium
   network are capable of communicating directly at the LL, without IP
   layer switching by routers. This presents an opportunity to optimize
   performance and perhaps lower cost by eliminating unnecessary LL hops
   through the medium.

   This memo discusses how unnecessary hops can be eliminated in a
   shared medium, while retaining the coherence of the existing Internet
   architecture. This issue has arisen in a number of IETF Working
   Groups concerned with LPDNs, including IPLPDN, IP over ATM, IDRP for
   IP, and BGP. It is time to take a careful look at the architectural
   issues to be solved. This memo first summarizes the relevant aspects
   of the original Internet architecture (Section 2), and then it
   explains the extra-hop problems created by shared media networks
   (Section 3). Finally, it discusses some possible solutions (Section
   4).

2. THE ORIGINAL INTERNET ARCHITECTURE

   We very briefly review the original architecture, to introduce the
   terminology and concepts. Figure 1 illustrates a typical set of four
   networks A, ... D, represented traditionally as clouds,
   interconnected by routers R2, R3, and R4. Routers R1 and R5 connect

to other parts of the Internet. Ha, ... Hd represent hosts connected
to these networks.

It is not necessary to distinguish between network and subnet in this
memo. We may assume that there is some address mask associated with
each "network" in Figure 1, allowing a host or router to divide the
32 bits of an IP address into an address for the cloud and a host
number that is defined uniquely only within that cloud.

```
           Ha              Hb              Hc              Hd

            |               |               |               |
            |               |               |               |
           _|_             _|_             _|_             _|_
          (   )           (   )           (   )           (   )
          (Net)           (Net)           (Net)           (Net)
          ( A )           ( B )           ( C )           ( D )
   - R1 --(   )-- R2 --(   )-- R3 --(   )-- R4 --(   )-- R5 --
          (   )           (   )           (   )           (   )
          (___)           (___)           (___)           (___)
```

               Figure 1. Example Internet Fragment

An Internet router is connected to local network(s) as a special kind
of host. Indeed, for network management purposes, a router plays the
role of a host by originating and terminating datagrams. However,
there is an important difference between a host and a router: the
routing function is mostly centralized in the routers, allowing hosts
to be "dumb" about routing. Internet hosts are required [RFC-1122]
to make only one simple routing decision: is the destination address
local to the connected network? If the address is not local, we say
it is "foreign" (relative to the connected network or to the host).

A host sends a datagram directly to a local destination address or
(for a foreign destination) to a first-hop router. The host
initially uses some "default" router for any new destination address.
If the default is the wrong choice, that router returns a Redirect
message and forwards the datagram. The Redirect message specifies
the preferred first-hop router for the given destination address.
The host uses this information, which it maintains in a "routing
cache" [RFC-1122], to determine the first-hop address for subsequent
datagrams to the same destination.

To actually forward an IP datagram across a network hop, the sender
must have the link layer (LL) address of the target. Therefore, each
host and router must have some "address resolution" procedure to map
IP address to an LL address. ARP, used for networks with broadcast
capability, is the most common address resolution procedure

[Plummer82]. If there is no LL broadcast capability (or if it is too
expensive), then there are two other approaches to address
resolution: local configuration tables, and "address-resolution
servers" (AR Servers).

If AR Servers are used for address resolution, hosts must be
configured with the LL address(es) of one or more nearby servers.
The mapping information provided by AR Servers might itself be
collected using a protocol that allows systems to register their LL
addresses, or from static configuration tables. The ARP packet
format and the overall ARP protocol structure (ARP Request/ARP Reply)
may be suitable for the communications between a host and an AR
server, even in the absence of the LL broadcast capabilities; this
would ease conversion of hosts to using AR Servers.

The examples in this memo use ARP for address resolution. At least
some of the LPDN's that are planned will provide sufficient broadcast
capability to support ARP. It is important to note that ARP operates
at the link layer, while the Redirect and routing cache mechanisms
operate at the IP layer of the protocol stack.

3. THE PROBLEMS INTRODUCED BY SHARED MEDIA

Figure 2 shows the same configuration as Figure 1, but now networks
A, B, C, and D are all within the same shared medium (SM), shown by
the dashed box enclosing the clouds. Networks A, ... D are now
logical IP networks (called LIS's in [Laubach93]) rather than
physical networks. Each of these logical networks may (or may not)
be administratively distinct. The SM allows direct connectivity
between any two hosts or routers connected to it. For example, host
Ha can interchange datagrams directly with host Hd or with router R4.
A router that has some but not all of its interfaces connected to the
shared medium is called a "border router"; R1 and R5 are examples.

Figure 2 illustrates the "classical" model [Laubach93] for use of the
Internet architecture within a shared medium, i.e., simply applying
the original Internet architecture described earlier. This will
provide correct but not optimal operation. For example, in the case
of two hosts on the same logical network (not shown in Figure 2), the
original rules will clearly work; the source host will forward a
datagram directly in a single hop to a host on the same logical
network. The original architectural rules will also work for
communication between any pair of hosts shown in Figure 2; for
example, host Ha would send a datagram to host Hd via the four-hop
path Ha -> R2 -> R3 -> R4 -> Hd. However, the classical model does
not take advantage of the direct connectivity Ha -> Hd allowed by the
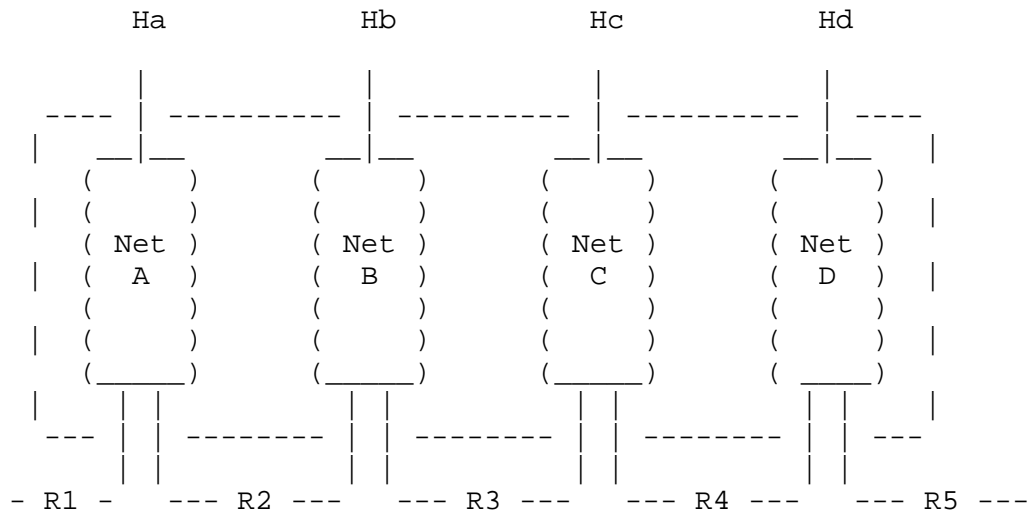shared medium.

```
            Ha              Hb              Hc              Hd

             |               |               |               |
      ---- | ---------- | ---------- | ---------- | ----
     |    __|__         __|__         __|__         __|__    |
          (     )       (     )       (     )       (     )
     |    (     )       (     )       (     )       (     )  |
          ( Net )       ( Net )       ( Net )       ( Net )
     |    (  A  )       (  B  )       (  C  )       (  D  )  |
          (     )       (     )       (     )       (     )
     |    (     )       (     )       (     )       (     )  |
          (_____)       (_____)       (_____)       ( ____)
     |      | |           | |           | |           | |    |
      ---  | | --------  | | --------  | | --------  | |  ---
           | |           | |           | |           | |
       - R1 -    --- R2 ---    --- R3 ---    --- R4 ---    --- R5 ---
```

              Figure 2. Logical IP Networks in Shared Medium

   This memo concerns mechanisms to achieve minimal-hop connectivity
   when it is desired. We should note that is may not always be
   desirable to achieve minimal-hop connectivity in a shared medium.
   For example, the "extra" hops may be needed to allow the routers to
   act as administrative firewalls. On the other hand, when such
   firewall protection is not required, it should be possible to take
   advantage of the shared medium to allow this datagram to use shorter
   paths. In general, it should be possible to choose between firewall
   security and efficient connectivity. This is discussed further in
   Section 4.6 below.

   We also note that the mechanisms described here can only optimize the
   path within the local SM. When the SM is only one segment of the
   path between source and receiver, removing hops locally may limit the
   ability to switch to globally more optimal paths that may become
   available as the result of routing changes. Thus, consider Ha-
   >...Hx, where host Hx is outside the SM to which host Ha is attached.
   Suppose that the shortest global path to Hx is via some border router
   Rb1. Local optimization using the techniques described below will
   remove extra hops in the SM and allow Ha->Rb1->...Hx. Now suppose
   that a later route change outside the SM makes the path Ha->Rb2-
   >...Hx more globally optimum, where Rb2 is another border router.
   Since Ha does not participate in the routing protocol, it does not
   know that it should switch to Rb2. It is possible that Rb2 may not
   realize it either; this is the situation:

     GC(Ha->Rb2->...Hx) < GC(Ha->Rb1->Rb2->...Hx) < GC(Ha->Rb1->...Hx)

where GC() represents some global cost function of the specified
path.

Note that ARP requires LL broadcast. Even if the SM supports
broadcast, it is likely that administrators will erect firewalls to
keep broadcasts local to their LIS.

There are three cases to be optimized. Suppose H and H' are hosts
and Rb and Rb' are border routers connected to the same same SM.
Then the following one-hop paths should be possible:

     H -> H': Host to host within the SM

     H -> Rb: Host to exit router

     Rb -> Rb': Entry border router to exit border router,
                 for transit traffic.

We may or not be able to remove the extra hop implicit in Rb -> R ->
H, where Rb, R, and H are within the same SM, but the ultimate source
is outside the SM. To remove this hop would require distribution of
host routes, not just network routes, between the two routers R and
Rb; this would adversely impact routing scalability.

There are a number of important requirements for any architectural
solution to these problems.

*   Interoperability

     Modified hosts and routers must interoperate with unmodified
     nodes.

*   Practicality

     Minimal software changes should be required.

*   Robustness

     The new scheme must be at least as robust against errors in
     software, configuration, or transmission as the existing
     architecture.

*   Security

     The new scheme must be at least as securable against subversion
     as the existing architecture.

The distinction between host and router is very significant from an
engineering viewpoint. It is considered to be much harder to make a
global change in host software than to change router software,
because there are many more hosts and host vendors than routers and
router vendors, and because hosts are less centrally administered
than routers. If it is necessary to change the specification of what
a host does (and it is), then we must minimize the extent of this
change.

4. SOME SOLUTIONS TO THE SM PROBLEMS

   Four different approaches have been suggested for solving these SM
   problems.

   (1) Hop-by-Hop Redirection

        In this approach, the host Redirect mechanism is extended to
        collapse multiple-hop paths within the same shared medium, hop-
        by-hop. A router is to be allowed to send, and a host allowed
        to accept, a Redirect message that specifies a foreign IP
        address within the same SM. We refer to this as a "foreign
        Redirect". Section 4.1 analyzes this approach in some detail.

   (2) Extended Routing

        Routing protocols can be modified to know about the SM and to
        provide LL addresses.

   (3) Extended Proxy ARP

        This is a form of the proxy ARP approach, in which the routing
        problem is solved implicitly by an extended address resolution
        mechanism at the LL. This approach has been described by
        Heinanen [Heinanen93] and by Garrett et al [Garratt93].

   (4) Route Query Messages

        This approach has been suggested by Halpern [Halpern93]. Rather
        than adding additional information to routing, this approach
        would add a new IP-layer mechanism using end-to-end query and
        reply datagrams.

   These four are discussed in the following four subsections.

   4.1 Hop-by-Hop Redirection

      The first scheme we consider would operate at the IP layer. It
      would cut out extra hops one by one, with each router in the path

operating on local information only. This approach requires both
host and router changes but no routing protocol changes.

The basic idea is that the first-hop router, upon observing that
the next hop is within the same SM, sends a foreign Redirect to
the source, redirecting it to the next hop. Successive
application of this algorithm at each intermediate router will
eventually result in a direct path from source host to destination
host, if both are within the same SM.

Suppose that Ha wants to send a datagram to Hd. We use the
notation IP.a for the IP address of entity a, and LL.a for the
corresponding LL address. Each line in the following shows an IP
datagram and the path that datagram will follow, separated by a
colon. The notation "Redirect(h, IP.a)" means a Redirect
specifying IP.a as the best next hop to reach host h.

    (1)   Datagram 1: Ha -> R2 -> R3 -> R4 -> Hd

    (2)   Redirect(Hd, IP.R3): R2 -> Ha

    (3)   Datagram 2: Ha -> R3 -> R4 -> Hd

    (4)   Redirect(Hd, IP.R4): R3 -> Ha

    (5)   Datagram 3: Ha -> R4 -> Hd

    (6)   Redirect(Hd, IP.Hd): R4 -> Ha

    (7)   Datagram 4: Ha -> Hd

There are three problems to be solved to make hop-by-hop
redirection work; we label them HH1, HH2, and HH3.

HH1: Each router must be able to resolve the LL address of the
     source Ha, to send a (foreign) Redirect.

     Let us assume that the link layer provides the source LL
     address when an IP datagram arrives. If the router
     determines that a Redirect should be sent, then it will be
     sent to the source LL address of the received datagram.

HH2: A source host must be able to perform address resolution to
     obtain the LL address of each router to which it is
     redirected.

     It would be possible for each router R, upon sending a
     Redirect to Ha, to also send an unsolicited ARP Reply point-

to-point to LL.Ha, updating Ha's ARP cache with LL.R.
However, there is not guarantee that this unsolicited ARP
Reply would be delivered. If it was lost, there would be a
forwarding black hole. The host could recover by starting
over from the original default router; however, this may be
too inefficient a solution.

A much more direct and efficient solution would introduce an
extended ICMP Redirect message (call it XRedirect) that
carries the LL address as well as the IP address of the
target. This would remove the issue of reliable delivery of
the unsolicited ARP described earlier, because the fate of
the LL address would be shared with the IP target address;
both would be delivered or neither would. (An XRedirect is
essentially the same as a Redirect in the OSI ES-IS
protocol).

Using XRedirect, the previous example becomes:

    (1)   Datagram 1: Ha -> R2 -> R3 -> R4 -> Hd

    (2)   XRedirect(Hd, IP.R3, LL.R3): R2 -> Ha

    (3)   Datagram 2: Ha -> R3 -> R4 -> Hd

    (4)   XRedirect(Hd, IP.R4, LL.R4): R3 -> Ha

    (5)   Datagram 3: Ha -> R4 -> Hd

    (6)   XRedirect(Hd, IP.Hd, LL.Hd): R4 -> Ha

    (7)   Datagram 4: Ha -> Hd

HH3: Each router should be able to recognize when it is the first
hop in the path, since a Redirect should be sent only by the
first hop router. Unfortunately this will be possible only
if the LL address corresponding to the IP source address has
been cached from an earlier event; a router in this chain
determines the LL address of the source from the arriving
datagram (see HH1 above). If it cannot determine whether it
is the first hop, a router must always send an [X]Redirect,
which will be spurious if the router is not the first hop.

Such spurious [X]Redirects will be sent to the IP address of
the source host, but using the LL address of the previous-hop
router. The propagation scope of [X]Redirects can be limited
to a single IP hop (see below), so they will go no further
than the previous-hop router, where they will be discarded.

However, there will be some router overhead to process these
useless [X]Redirects

Next, we discuss the changes in hosts and in routers required for
hop-by-hop redirection.

o   Host Changes

      The Host Requirements RFC [RFC-1122] specifies the host
      mechanism for routing an outbound datagram in terms of
      sending the datagram directly to a local destination or else
      to the first hop router (to reach a foreign destination)
      [RFC-1122 3.3.1]. Although this mechanism assumes a local
      address, a foreign address for a first-hop router should work
      equally well.

      The target address contained in the routing cache is updated
      by Redirect messages. There is currently a restriction on
      what target addresses may be accepted in Redirect messages
      [RFC-1122 3.2.2.2], which would prevent foreign Redirects
      from working:

           A Redirect message SHOULD be silently discarded if the
           new router address it specifies is not on the same
           connected (sub-) net through which the Redirect arrived,
           or if the source of the Redirect is not the current
           first-hop router for the specified destination.

      To support foreign Redirects requires simply removing the
      first validity check. The second check, which requires an
      acceptable Redirect to come from the node to which the
      datagram that triggered the Redirect was sent, is retained.
      The same validity check would be used for XRedirects.

      In order to send a datagram to the target address found in
      the routing cache, a host must resolve the IP address into a
      LL address. No change should be necessary in the host's IP-
      to-LL resolution mechanism to handle a foreign rather than a
      local address.

      The Hop-by-Hop redirection requires changes to the semantics
      of the IP address that an ICMP Redirect is allowed to carry.
      Under the present definition [Postel81b], an ICMP Redirect
      message is only allowed to carry an IP address of a router.
      In order for the hop-by-hop redirection mechanism to
      eliminate all router hops, allowing two hosts connected to
      the same SM to communicate directly, a [X]Redirect message
      must be able to carry the IP address of the destination host.

     o   Router Changes

          The router changes required for hop-by-hop redirection are
          much more extensive than the host changes. The examples
          given earlier showed the additional router functions that
          would be needed.

          Consider a router that is connected to an SM. When it
          receives a datagram from the SM, it tests whether the next
          hop is on the same SM, and if so, it sends a foreign
          XRedirect to the source host, using the link layer address
          with which the datagram arrived.

          A router should avoid sending more than a limited number of
          successive foreign Redirects to the same host. This is
          necessary because an unmodified host may legitimately ignore
          a Redirect to a foreign network and continue to forward
          datagrams to the same router. A router can accomplish this
          limitation by keeping a cache of foreign Redirects sent.

          Note that foreign Redirects generated by routers according to
          these rules, like the current local Redirects, may travel
          exactly one link-layer hop. It is therefore reasonable and
          desirable to set their TTL to 1, to ensure they cannot stray
          outside the SM.

          The extra check needed to determine whether to generate a
          Redirect may incur additional processing and thus result in a
          performance degradation; to avoid this, a router may not
          perform the check at all but just forward the packet. The
          scheme with [X]Redirects is not applicable to such a router.

          Finally, note that the hop-by-hop redirection scheme is only
          applicable when the source host is connected to an SM, since
          routers do not listen to Redirects. To optimize the
          forwarding of transit traffic between entry and exit border
          routers, an extension to routing is required, as discussed in
          the following section. Conversely, an extension to the
          routing protocol cannot be used to optimize forwarding
          traffic from a host connected to the SM, since a host should
          not listen to routing protocols.

   4.2 Extended Routing

      The routing protocols may be modified to carry additional
      information that is specific to the SM. The router could use the
      attribute "SameSM" for a route to deduce the shortest path to be
      reported to its neighbors. It could also carry the LL addresses

with each router IP address.

For example, the extended routing protocol would allow R2 to know
that R4 is the best next-hop to reach the destination network in
the same SM, and to know both IP.R4 and LL.R4, leading to the path
Ha->R2->R4->Hb. Further optimization cannot be done with extended
routing alone, since the host does not participate in routing, and
because we want the routing protocol to handle only per-network
information, not per-host information. Hop-by-hop redirection
could then be used to eliminate all router hops, as in the
following sequence:

    (1) Datagram 1: Ha -> R2 -> R4 -> Hd

    (2) XRedirect(Hd, IP.R4, LL.R4): R2 -> Ha

    (3) Datagram 2: Ha -> R4 -> Hd

    (4) XRedirect(Hd, IP.Hd, LL.Hd): R4 -> Ha

    (5) Datagram 3: Ha -> Hd

There are three aspects to the routing protocol extension:

(1)   the ability to pass "third-party" information -- a router
      should be able to specify the address (IP address and perhaps
      LL address) of some other router as the next-hop;

(2)   knowledge of the "SameSM" attribute for routes; and

(3)   knowledge of LL addresses corresponding to IP addresses of
      routers within the same SM.

A router must be able to determine that a particular IP address
(e.g., the source address) is in the same SM. There are several
possible ways to make this information available to a router in
the SM.

(1)   A router may use a single physical interface to an SM; this
      implies that all its logical interfaces lie within the same
      SM.

(3)   There might be some administrative structure in the IP
      addresses, e.g., all IP addresses within a particular
      national SM might have a common prefix string.

(3)   There might be configuration information, either local to the
      router or available from some centralized server (e.g, the

DNS). Note that a router could consult this server in the
background while continuing to forward datagrams without
delay. The only consequence of a delay in obtaining the
"SameSM" information would be some unnecessary (but
temporary) hops.

4.3 Extended Proxy ARP

The approach of Heinanen [Heinanen93] was intended to solve the
problem of address resolution in a shared medium with no broadcast
mechanism ("Non-Broadcast, MultiAccess" or NBMA). Imagine that
the shared medium has a single IP network number, i.e., it is one
network "cloud". Heinanen envisions a set of AR Servers within
this medium. These AR Servers run some routing protocol among
themselves. A source host issues an ARP Request (via a point-to-
point LL transmission) to an AR Server with which it is
associated. This ARP Request is forwarded hop-by-hop at the link
layer through the AR Servers, towards the AR Server that is
associated with the destination host. That AR Server resolves the
address (using information learned from either host advertisement
or a configuration file), and returns an ARP Reply back through
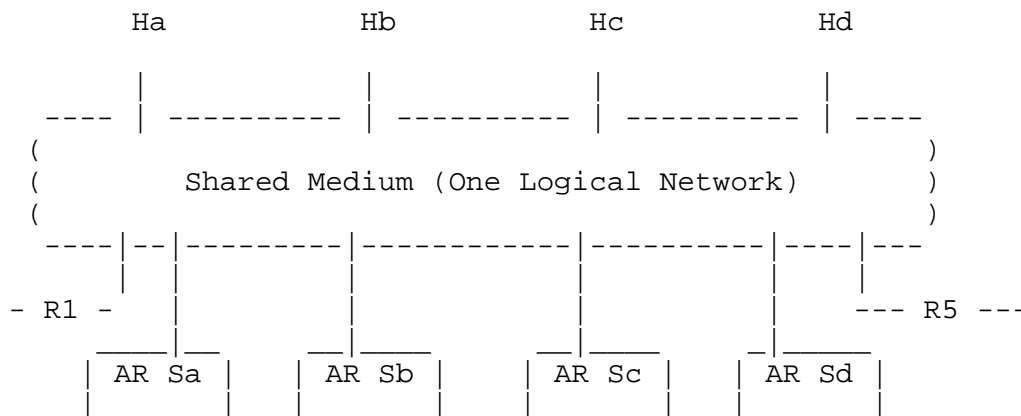the AR Servers to the source host.

```
           Ha              Hb              Hc              Hd

            |               |               |               |
     ----   | ----------    | ----------    | ----------    | ----
    (                                                            )
    (          Shared Medium (One Logical Network)        )
    (                                                            )
     ----|--|---------|-----------|----------|----|---
         |  |         |           |          |    |
   - R1 - |  |         |           |          | --- R5 ---
      ____|__        __|___        __|___       _|___
     | AR Sa |      | AR Sb |     | AR Sc |    | AR Sd |
     |_____|      |_____|     |_____|    |_____|
```

        Figure 3. Single-Cloud Shared Medium

Figure 3 suggests that each of the hosts Ha, ... Hd is associated
with a corresponding AR Server "AR Sa", ..."AR Sd".

This same scheme could be applied to the LIS model of Figure 2.
The AR Servers would be implemented in the routers, and if the
medium supports broadcast then the hosts would be configured for
proxy ARP. That is, the host would be told that all destinations

are local, so it will always issue an ARP request for the final
destination. The set of AR Servers would resolve this request.

Since routing loops are a constant possibility, Heinanen's
proposal includes the addition of a hop count to ARP requests and
replies.

Like all proxy ARP schemes, this one has a seductive simplicity.
However, solving the SM problem at the LL has several costs. It
requires a complete round-trip time before the first datagram can
flow. It requires a hop count in the ARP packet. This seems like
a tip-off that the link layer may not be the most appropriate
place to solve the SM problem.

4.4 Routing Query Messages

This scheme [Halpern93] introduces a new IP level mechanism: SM
routing query and reply messages. These messages are forwarded as
IP datagrams hop-by-hop in the direction of the destination
address. The exit router can return a reply, again hop-by-hop,
that finally reaches the source host as an XRedirect. It would
also be possible (but not necessary) to modify hosts to initiate
these queries.

The query/reply pair is supplying the same information that we
would add to routing protocols under Extended Routing. However,
the Query/Reply messages would allow us to keep the current
routing protocols unchanged, and would also provide the extra
information only for the routes that are actually needed, thus
reducing the routing overhead. Note that the Query/Reply sequence
can happen in parallel with forwarding the initial datagram hop-
by-hop, so it does not add an extra round-trip delay.

4.5 Stale Routing Information

We must consider what happens when the network topology changes.
The technique of extended routing (Section 4.2) is capable of
providing sufficient assurances that stale information will be
purged from the system within the convergence time associated with
a particular routing protocol being used.

However, the three other techniques (hop-by-hop redirection,
extended Proxy ARP, and routing query messages) may be expected to
provide minimal-hop forwarding only as long as the network
topology remains unchanged since the time such information was
acquired. Changes in the topology may result in a change in the
minimal-hop path, so that the first-hop router may no longer be
the correct choice. If the host that is using this first-hop

router is not aware of the changes, then instead of a minimal-hop
path the host could be using a path that is now suboptimal,
perhaps highly sub-optimal, with respect to the number of hops.

Futhermore, use of the information acquired via either extended
Proxy ARP or routing query messages to optimize routing between
routers attached to the same SM is highly problematic, because
presence of stale information on routers could result in
forwarding loops that might persist as long as the information
isn't purged; neither approach provides suitable handling of stale
information.

4.6 Implications of Filtering (Firewalls)

For a variety of reasons an administrator of a LIS may erect IP
Layer firewalls (perform IP-layer filtering) to constrain LL
connectivity between the hosts/routers within the LIS and
hosts/routers in other LISs within the same SM. To avoid
disruption in forwarding, the mechanisms described in this
document need to take into account such firewalls.

Using [X]Redirects requires a router that generates an [X]Redirect
to be cognizant of possible Link Layer connectivity constraints
between the router that is specified as the Next Hop in the
Redirect and the host that is the target of the Redirect.

Using extended routing requires a router that originates and/or
propagates "third-party" information be cognizant of the possible
Link Layer connectivity constraints. Specifically, a router should
not propagate "third-party" information when there is a lack of
Link Layer connectivity between the router depicted by the
information and the router which is the immediate recipient of
that information.

Using extended proxy ARP requires an ARP Server not to propagate
an ARP Request to another ARP server if there are Link Layer
connectivity constraints between the originator of the ARP Request
and the other ARP server.

Using SM routing query and reply messages requires the routers
that pass the messages to be aware of the possible Link Layer
connectivity constraints. The flow of messages need to reflect
these constraints.

5. SECURITY CONSIDERATIONS

   We should discuss the security issues raised by our suggested
   changes. We should note that we are not talking about "real"
   security here; real Internet security will require cryptographic
   techniques on an end-to-end basis. However, it should not be easy to
   subvert the basic delivery mechanism of IP to cause datagrams to flow
   to unexpected places.

   With this understanding, the security problems arise in two places:
   the ICMP Redirect messages and the ARP replies.

   *   ICMP Redirect Security

        We may reasonably require that the routers be secure. They are
        generally under centralized administrative control, and we may
        assume that the routing protocols will contain sufficient
        authentication mechanisms (even if it is not currently true).
        Therefore, a host will reasonably be able to trust a Redirect
        that comes from a router.

        However, it will NOT be reasonable for a host to trust another
        host. Suppose that the target host in the examples of Section
        4.1 is untrustworthy; there is no way to prevent its issuing a
        new Redirect to some other destination, anywhere in the
        Internet. On the other hand, this exposure is no worse than it
        was; the target host, once subverted, could always act as a
        hidden router to forward traffic elsewhere.

   *   ARP Security

        Currently, an ARP Reply can come only from the local network,
        and a physically isolated network can be administrative secured
        from subversion of ARP. However, an ARP Reply can come from
        anywhere within the SM, and an evil-doer can use this fact to
        divert the traffic flow from any host within the SM
        [Bellovin89].

        The XRedirect closes this security hole. Validating the
        XRedirect (as coming from the node to which the last datagram
        was sent) will also validate the LL address.

        Another approach is to validate the source address from which
        the ARP Reply was received (assuming the link layer protocol
        carries the source address and the driver supplies it). An
        acceptable ARP reply for destination IP address D can only come
        from LL address x, where the routing cache maps D -> E and the
        ARP cache gives x as the translation of E. This validation,

involving both routing and ARP caches, might be ugly to
implement in a strictly-layered implementation. It would be
natural if layering were already violated by combining the ARP
cache and routing cache.

It is possible for the link layer to have security mechanisms that
could interfere with IP-layer connectivity. In particular, there
could possible be non-transitivity of logical interconnection within
a shared medium. In particular, some large public data networks may
include configuration options that could allow Net A to talk to Net B
and Net B to talk to Net C, but prevent A from talking directly to C.
In this case, the routing protocols have to be sophisticated enough
to handle such anomalies.

6. CONCLUSIONS

We have discussed four possible extensions to the Internet
architecture to allow hop-efficient forwarding of IP datagrams within
shared media, when this optimization is allowed by IP-layer
firewalls. We do not draw any conclusions in this paper about the
best mechanisms.

Our suggested extensions are evolutionary, leaving intact the basic
ideas of the current Internet architecture. It would be possible to
make (and some have suggested) much more radical changes to
accommodate shared media. In the extreme, one could entirely abolish
the inner clouds in Figure 2, so that there would be no logical
network structure within the SM. The IP addresses would then be
logical, and some mechanism of distributed servers would be needed to
find routes within this random haze. We think this approach ignores
all the requirements for management and security in today's Internet.
It might make a good research paper, but it would not be good
Internet design strategy.

7. ACKNOWLEDGMENTS

8. REFERENCES

  [Bellovin89] Bellovin, S., "Security Problems in the TCP/IP Protocol
       Suite", ACM CCR, v. 19. no. 2, April 1989.

  [Garrett93] Garrett, J., Hagan, J. and J. Wong, "Directed ARP", RFC
       1433, AT&T Bell Laboratories, University of Pennsylvania, March
       1993.

  [Plummer82] Plummer, D., "An Ethernet Address Resolution Protocol",
       STD 37, RFC 826, MIT, November 1982.

  [Halpern93] Halpern, J., Private Communication, July 1993.

  [Heinanen93] Heinanen, J., "NBMA Address Resolution Protocol (NBMA
       ARP)", Work in Progress, June 1993.

  [Laubach93] Laubach, M., "Classical IP and ARP over ATM", RFC 1577,
       Hewlett-Packard Laboratories, January 1994.

  [Postel81a] Postel, J., "Internet Protocol - DARPA Internet Program
       Protocol Specification", STD 5, RFC 791, DARPA, September 1981.

  [Postel81b] Postel, J., "Internet Control Message Protocol- DARPA
       Internet Program Protocol Specification", STD 5, RFC 792, ISI,
       September 1981.

  [PSC81] Postel, J., Sunshine, C., and D. Cohen, "The ARPA Internet
       Protocol", Computer Networks 5, pp. 261-271, 1983.

  [RFC-1122] Braden, R., Editor, "Requirements for Internet Hosts --
       Communication Layers", STD 3, RFC 1122, USC/Information Sciences
       Institutue, October 1989.

Authors' Addresses

    Bob Braden
    Information Sciences Institute
    University of Southern California
    4676 Admiralty Way
    Marina del Rey, CA 90292

    Phone: (310) 822-1511
    EMail: Braden@ISI.EDU

    Jon Postel
    Information Sciences Institute
    University of Southern California
    4676 Admiralty Way
    Marina del Rey, CA 90292

    Phone: (310) 822-1511
    EMail: Postel@ISI.EDU

    Yakov Rekhter
    Office 32-017
    T.J. Watson Research Center, IBM Corp.
    P.O. Box 218,
    Yorktown Heights, NY 10598

    Phone: (914) 945-3896
    EMail: Yakov@WATSON.IBM.COM