                  Requirements for Internet Gateways

Status of this Memo

   This document is a formal statement of the requirements to be met by
   gateways used in the Internet system. As such, it is an official
   specification for the Internet community. Distribution of this memo
   is unlimited.

   This RFC summarizes the requirements for gateways to be used between
   networks supporting the Internet protocols. While it was written
   specifically to support National Science Foundation research
   programs, the requirements are stated in a general context and are
   applicable throughout the Internet community.

   The purpose of this document is to present guidance for vendors
   offering gateway products that might be used or adapted for use in an
   Internet application. It enumerates the protocols required and gives
   references to RFCs and other documents describing the current
   specifications. In a number of cases the specifications are evolving
   and may contain ambiguous or incomplete information. In these cases
   further discussion giving specific guidance is included in this
   document. Specific policy issues relevant to the NSF scientific
   networking community are summarized in an Appendix. As other
   specifications are updated this document will be revised. Vendors
   are encouraged to maintain contact with the Internet research
   community.

1. Introduction

   The following material is intended as an introduction and background
   for those unfamiliar with the Internet architecture and the Internet
   gateway model. General background and discussion on the Internet
   architecture and supporting protocol suite can be found in the DDN
   Protocol Handbook [25] and ARPANET Information Brochure [26], see
   also [19, 28, 30, 31].

   The Internet protocol architecture was originally developed under
   DARPA sponsorship to meet both military and civilian communication
   requirements [32]. The Internet system presently supports a variety
   of government and government-sponsored operational and research
   activities. In particular, the National Science Foundation (NSF) is
   building a major extension to the Internet to provide user access to

national supercomputer centers and other national scientific resources, and to provide a computer networking capability to a large number of universities and colleges.

In this document there are many terms that may be obscure to one unfamiliar with the Internet protocols. There is not much to be done about that but to learn, so dive in. There are a few terms that are much abused in general discussion but are carefully and intentionally used in this document. These few terms are defined here.

Packet   A packet is the unit of transmission on a physical network.

Datagram   A datagram is the unit of transmission in the IP protocol. To cross a particular network a datagram is encapsulated inside a packet.

Router   A router is a switch that receives data transmission units from input interfaces and, depending on the addresses in those units, routes them to the appropriate output interfaces. There can be routers at different levels of protocol. For example, Interface Message Processors (IMPs) are packet-level routers.

Gateway   In the Internet documentation generally, and in this document specifically, a gateway is an IP-level router. In the Internet community the term has a long history of this usage [32].

1.1. The DARPA Internet Architecture

1.1.1. Internet Protocols

The Internet system consists of a number of interconnected packet networks supporting communication among host computers using the Internet protocols. These protocols include the Internet Protocol (IP), the Internet Control Message Protocol (ICMP), the Transmission Control Protocol (TCP), and application protocols depending upon them [22].

All Internet protocols use IP as the basic data transport mechanism. IP [1,31] is a datagram, or connectionless, internetwork service and includes provision for addressing, type-of-service specification, fragmentation and reassembly, and security information. ICMP [2] is considered an integral

part of IP, although it is architecturally layered upon IP.
ICMP provides error reporting, flow control and first-hop
gateway redirection.

Reliable data delivery is provided in the Internet protocol
suite by transport-level protocols such as the Transmission
Control Protocol (TCP), which provides end-end retransmission,
resequencing and connection control. Transport-level
connectionless service is provided by the User Datagram
Protocol (UDP).

1.1.2. Networks and Gateways

The constituent networks of the Internet system are required
only to provide packet (connectionless) transport. This
requires only delivery of individual packets. According to the
IP service specification, datagrams can be delivered out of
order, be lost or duplicated and/or contain errors. Reasonable
performance of the protocols that use IP (e.g., TCP) requires
an IP datagram loss rate of less than 5%. In those networks
providing connection-oriented service, the extra reliability
provided by virtual circuits enhances the end-end robustness of
the system, but is not necessary for Internet operation.

Constituent networks may generally be divided into two classes:

* Local-Area Networks (LANs)

   LANs may have a variety of designs, typically based upon
   buss, ring, or star topologies. In general, a LAN will
   cover a small geographical area (e.g., a single building or
   plant site) and provide high bandwidth with low delays.

* Wide-Area Networks (WANs)

   Geographically-dispersed hosts and LANs are interconnected
   by wide-area networks, also called long-haul networks.
   These networks may have a complex internal structure of
   lines and packet-routers (typified by ARPANET), or they may
   be as simple as point-to-point lines.

In the Internet model, constituent networks are connected
together by IP datagram forwarders which are called "gateways"
or "IP routers". In this document, every use of the term
"gateway" is equivalent to "IP router". In current practice,
gateways are normally realized with packet-switching software

executing on a general-purpose CPU, but special-purpose
hardware may also be used (and may be required for future
higher-throughput gateways).

A gateway is connected to two or more networks, appearing to
each of these networks as a connected host. Thus, it has a
physical interface and an IP address on each of the connected
networks. Forwarding an IP datagram generally requires the
gateway to choose the address of the next-hop gateway or (for
the final hop) the destination host. This choice, called
"routing", depends upon a routing data-base within the gateway.
This routing data-base should be maintained dynamically to
reflect the current topology of the Internet system; a gateway
normally accomplishes this by participating in distributed
routing and reachability algorithms with other gateways.
Gateways provide datagram transport only, and they seek to
minimize the state information necessary to sustain this
service in the interest of routing flexibility and robustness.

Routing devices may also operate at the network level; in this
memo we will call such devices MAC routers (informally called
"level-2 routers", and also called "bridges"). The name
derives from the fact that MAC routers base their routing
decision on the addresses in the MAC headers; e.g., in IEEE
802.3 networks, a MAC router bases its decision on the 48-bit
addresses in the MAC header. Network segments which are
connected by MAC routers share the same IP network number,
i.e., they logically form a single IP network.

Another variation on the simple model of networks connected
with gateways sometimes occurs: a set of gateways may be
interconnected with only serial lines, to effectively form a
network in which the routing is performed at the internetwork
(IP) level rather than the network level.

1.1.3. Autonomous Systems

For technical, managerial, and sometimes political reasons, the
gateways of the Internet system are grouped into collections
called "autonomous systems" [35]. The gateways included in a
single autonomous system (AS) are expected to:

    * Be under the control of a single operations and
       maintenance (O&M) organization;

    * Employ common routing protocols among themselves, to
       maintain their routing data-bases dynamically.

A number of different dynamic routing protocols have been
developed (see Section 4.1); the particular choice of routing
protocol within a single AS is generically called an interior
gateway protocol or IGP.

An IP datagram may have to traverse the gateways of two or more
ASs to reach its destination, and the ASs must provide each
other with topology information to allow such forwarding. The
Exterior Gateway Protocol (EGP) is used for this purpose,
between gateways of different autonomous systems.

1.1.4. Addresses and Subnets

An IP datagram carries 32-bit source and destination addresses,
each of which is partitioned into two parts -- a constituent
network number and a host number on that network.
Symbolically:

    IP-address ::= { <Network-number>, <Host-number> }

To finally deliver the datagram, the last gateway in its path
must map the host-number (or "rest") part of an IP address into
the physical address of a host connection to the constituent
network.

This simple notion has been extended by the concept of
"subnets", which were introduced in order to allow arbitrary
complexity of interconnected LAN structures within an
organization, while insulating the Internet system against
explosive growth in network numbers and routing complexity.
Subnets essentially provide a two-level hierarchical routing
structure for the Internet system. The subnet extension,
described in RFC-950 [21], is now a required part of the
Internet architecture. The basic idea is to partition the
<host number> field into two parts: a subnet number, and a true
host number on that subnet.

    IP-address ::=
            { <Network-number>, <Subnet-number>, <Host-number> }

The interconnected LANs of an organization will be given the
same network number but different subnet numbers. The
distinction between the subnets of such a subnetted network
must not be visible outside that network. Thus, wide-area
routing in the rest of the Internet will be based only upon the
<Network-number> part of the IP destination address; gateways
outside the network will lump <Subnet-number> and <Host-number>

together to form an uninterpreted "rest" part of the 32-bit IP
address. Within the subnetted network, the local gateways must
route on the basis of an extended network number:

    { <Network-number>, <Subnet-number> }.

The bit positions containing this extended network number are
indicated by a 32-bit mask called the "subnet mask" [21]; it is
recommended but not required that the <Subnet-number> bits be
contiguous and fall between the <Network-number> and the
<Host-number> fields. No subnet should be assigned the value
zero or -1 (all one bits).

Flexible use of the available address space will be
increasingly important in coping with the anticipated growth of
the Internet. Thus, we allow a particular subnetted network to
use more than one subnet mask. Several campuses with very
large LAN configurations are also creating nested hierarchies
of subnets, sub-subnets, etc.

There are special considerations for the gateway when a
connected network provides a broadcast or multicast capability;
these will be discussed later.

1.2. The Internet Gateway Model

There are two basic models for interconnecting local-area networks
and wide-area (or long-haul) networks in the Internet. In the
first, the local-area network is assigned a network number and all
gateways in the Internet must know how to route to that network.
In the second, the local-area network shares (a small part of) the
address space of the wide-area network. Gateways that support
this second model are called "address sharing gateways" or
"transparent gateways". The focus of this memo is on gateways
that support the first model, but this is not intended to exclude
the use of transparent gateways.

1.2.1. Internet Gateways

An Internet gateway is an IP-level router that performs the
following functions:

1. Conforms to specific Internet protocols specified in
    this document, including the Internet Protocol (IP),
    Internet Control Message Protocol (ICMP), and others as
    necessary. See Section 2 (Protocols Required).

2. Interfaces to two or more packet networks. For each

connected network the gateway must implement the
functions required by that network. These functions
typically include:

a. encapsulating and decapsulating the IP datagrams with
    the connected network framing (e.g., an Ethernet
    header and checksum);

b. sending and receiving IP datagrams up to the maximum
    size supported by that network, this size is the
    network's "Maximum Transmission Unit" or "MTU";

c. translating the IP destination address into an
    appropriate network-level address for the connected
    network (e.g., an Ethernet hardware address);

d. responding to the network flow control and error
    indication, if any.

See Section 3 (Constituent Network Interface), for
details on particular constituent network interfaces.

3. Receives and forwards Internet datagrams. Important
    issues are buffer management, congestion control, and
    fairness. See Section 4 (Gateway Algorithms).

    a. Recognizes various error conditions and generates
        ICMP error and information messages as required.

    b. Drops datagrams whose time-to-live fields have
        reached zero.

    c. Fragments datagrams when necessary to fit into the
        MTU of the next network.

4. Chooses a next-hop destination for each IP datagram,
    based on the information in its routing data-base. See
    Section 4 (Gateway Algorithms).

5. Supports an interior gateway protocol (IGP) to carry out
    distributed routing and reachability algorithms with the
    other gateways in the same autonomous system. In
    addition, some gateways will need to support the
    Exterior Gateway Protocol (EGP) to exchange topological
    information with other autonomous systems. See
    Section 4 (Gateway Algorithms).

       6.   Provides system support facilities, including loading,
            debugging, status reporting, exception reporting and
            control. See Section 5 (Operation and Maintenance).

   1.2.2. Embedded Gateways

   A gateway may be a stand-alone computer system, dedicated to
   its IP router functions. Alternatively, it is possible to
   embed gateway functionality within a host operating system
   which supports connections to two or more networks. The
   best-known example of an operating system with embedded gateway
   code is the Berkeley BSD system. The embedded gateway feature
   seems to make internetting easy, but it has a number of hidden
   pitfalls:

     1.   If a host has only a single constituent-network
          interface, it should not act as a gateway.

          For example, hosts with embedded gateway code that
          gratuitously forward broadcast packets or datagrams on
          the same net often cause packet avalanches.

     2.   If a (multihomed) host acts as a gateway, it must
          implement ALL the relevant gateway requirements
          contained in this document.

          For example, the routing protocol issues (see Sections
          2.6 and 4.1) and the control and monitoring problems are
          as hard and important for embedded gateways as for
          stand-alone gateways.

            Since Internet gateway requirements and
            specifications may change independently of operating
            system changes, an administration that operates an
            embedded gateway in the Internet is strongly advised
            to have an ability to maintain and update the gateway
            code (e.g., this might require gateway code source).

     3.   Once a host runs embedded gateway code, it becomes part
          of the Internet system. Thus, errors in software or
          configuration of such a host can hinder communication
          between other hosts. As a consequence, the host
          administrator must lose some autonomy.

          In many circumstances, a host administrator will need to
          disable gateway coded embedded in the operating system,
          and any embedded gateway code must be organized so it
          can be easily disabled.

4. If a host running embedded gateway code is concurrently
    used for other services, the O&M (operation and
    maintenance) requirements for the two modes of use may
    be in serious conflict.

    For example, gateway O&M will in many cases be performed
    remotely by an operations center; this may require
    privileged system access which the host administrator
    would not normally want to distribute.

1.2.3. Transparent Gateways

The basic idea of a transparent gateway is that the hosts on
the local-area network behind such a gateway share the address
space of the wide-area network in front of the gateway. In
certain situations this is a very useful approach and the
limitations do not present significant drawbacks.

The words "in front" and "behind" indicate one of the
limitations of this approach: this model of interconnection is
suitable only for a geographically (and topologically) limited
stub environment. It requires that there be some form of
logical addressing in the network level addressing of the
wide-area network (that is, all the IP addresses in the local
environment map to a few (usually one) physical address in the
wide-area network, in a way consistent with the { IP address
<-> network address } mapping used throughout the wide-area
network).

Multihoming is possible on one wide-area network, but may
present routing problems if the interfaces are geographically
or topologically separated. Multihoming on two (or more)
wide-area networks is a problem due to the confusion of
addresses.

The behavior that hosts see from other hosts in what is
apparently the same network may differ if the transparent
gateway cannot fully emulate the normal wide-area network
service. For example, if there were a transparent gateway
between the ARPANET and an Ethernet, a remote host would not
receive a Destination Dead message [3] if it sent a datagram to
an Ethernet host that was powered off.

1.3. Gateway Characteristics

Every Internet gateway must perform the functions listed above.
However, a vendor will have many choices on power, complexity, and
features for a particular gateway product. It may be helpful to
observe that the Internet system is neither homogeneous nor
fully-connected. For reasons of technology and geography, it is
growing into a global-interconnect system plus a "fringe" of LANs
around the "edge".

* The global-interconnect system is comprised of a number of
  wide-area networks to which are attached gateways of several
  ASs; there are relatively few hosts connected directly to
  it. The global-interconnect system includes the ARPANET,
  the NSFNET "backbone", the various NSF regional and
  consortium networks, other ARPA sponsored networks such as
  the SATNET and the WBNET, and the DCA sponsored MILNET. It
  is anticipated that additional networks sponsored by these
  and other agencies (such as NASA and DOE) will join the
  global-interconnect system.

* Most hosts are connected to LANs, and many organizations
  have clusters of LANs interconnected by local gateways.
  Each such cluster is connected by gateways at one or more
  points into the global-interconnect system. If it is
  connected at only one point, a LAN is known as a "stub"
  network.

Gateways in the global-interconnect system generally require:

* Advanced routing and forwarding algorithms

  These gateways need routing algorithms which are highly
  dynamic and also offer type-of-service routing. Congestion
  is still not a completely resolved issue [24]. Improvements
  to the current situation will be implemented soon, as the
  research community is actively working on these issues.

* High availability

  These gateways need to be highly reliable, providing 24 hour
  a day, 7 days a week service. In case of failure, they must
  recover quickly.

* Advanced O&M features

  These gateways will typically be operated remotely from a
  regional or national monitoring center. In their

interconnect role, they will need to provide sophisticated
means for monitoring and measuring traffic and other events
and for diagnosing faults.

* High performance

Although long-haul lines in the Internet today are most
frequently 56 Kbps, DS1 lines (1.5 Mbps) are of increasing
importance, and even higher speeds are likely in the future.
Full-duplex operation is provided at any of these speeds.

The average size of Internet datagrams is rather small, of
the order of 100 bytes. At DS1 line speeds, the
per-datagram processing capability of the gateways, rather
than the line speed, is likely to be the bottleneck. To
fill a DS1 line with average-sized Internet datagrams, a
gateway would need to pass -- receive, route, and send --
2,000 datagrams per second per interface. That is, a
gateway which supported 3 DS1 lines and and Ethernet
interface would need to be able to pass a dazzling 2,000
datagrams per second in each direction on each of the
interfaces, or a aggregate throughput of 8,000 datagrams per
second, in order to fully utilize DS1 lines. This is beyond
the capability of current gateways.

Note: some vendors count input and output operations
separately in datagrams per second figures; for these
vendors, the above example would imply 16,000 datagrams
per second !

Gateways used in the "LAN fringe" (e.g., campus networks) will
generally have to meet less stringent requirements for
performance, availability, and maintenance. These may be high or
medium-performance devices, probably competitively procured from
several different vendors and operated by an internal organization
(e.g., a campus computing center). The design of these gateways
should emphasize low average delay and good burst performance,
together with delay and type-of-service sensitive resource
management. In this environment, there will be less formal O&M,
more hand-crafted static configurations for special cases, and
more need for inter-operation with gateways of other vendors. The
routing mechanism will need to be very flexible, but need not be
so highly dynamic as in the global-interconnect system.

It is important to realize that Internet gateways normally operate
in an unattended mode, but that equipment and software faults can
have a wide-spread (sometimes global) effect. In any environment,

a gateway must be highly robust and able to operate, possibly in a
degraded state, under conditions of extreme congestion or failure
of network resources.

Even though the Internet system is not fully-interconnected, many
parts of the system do need to have redundant connectivity. A
rich connectivity allows reliable service despite failures of
communication lines and gateways, and it can also improve service
by shortening Internet paths and by providing additional capacity.
The engineering tradeoff between cost and reliability must be made
for each component of the Internet system.

2. Protocols Required in Gateways

   The Internet architecture uses datagram gateways to interconnect
   constituent networks. This section describes the various protocols
   which a gateway needs to implement.

   2.1. Internet Protocol (IP)

      IP is the basic datagram protocol used in the Internet system [19,
      31]. It is described in RFC-791 [1] and also in MIL-STD-1777 [5]
      as clarified by RFC-963 [36] ([1] and [5] are intended to describe
      the same standard, but in quite different words). The subnet
      extension is described in RFC-950 [21].

      With respect to current gateway requirements the following IP
      features can be ignored, although they may be required in the
      future: Type of Service field, Security option, and Stream ID
      option. However, if recognized, the interpretation of these
      quantities must conform to the standard specification.

      It is important for gateways to implement both the Loose and
      Strict Source Route options. The Record Route and Timestamp
      options are useful diagnostic tools and must be supported in all
      gateways.

      The Internet model requires that a gateway be able to fragment
      datagrams as necessary to match the MTU of the network to which
      they are being forwarded, but reassembly of fragmented datagrams
      is generally left to the destination hosts. Therefore, a gateway
      will not perform reassembly on datagrams it forwards.

      However, a gateway will generally receive some IP datagrams
      addressed to itself; for example, these may be ICMP Request/Reply
      messages, routing update messages (see Sections 2.3 and 2.6), or
      for monitoring and control (see Section 5). For these datagrams,
      the gateway will be functioning as a destination host, so it must
      implement IP reassembly in case the datagrams have been fragmented
      by some transit gateway. The destination gateway must have a
      reassembly buffer which is at least as large as the maximum of the
      MTU values for its network interfaces and 576. Note also that it
      is possible for a particular protocol implemented by a host or
      gateway to require a lower bound on reassembly buffer size which
      is larger than 576. Finally, a datagram which is addressed to a
      gateway may use any of that gateway's IP addresses as destination
      address, regardless of which interface the datagram enters.

      There are five classes of IP addresses: Class A through
      Class E [23]. Of these, Class D and Class E addresses are

reserved for experimental use. A gateway which is not
participating in these experiments must ignore all datagrams with
a Class D or Class E destination IP address. ICMP Destination
Unreachable or ICMP Redirect messages must not result from
receiving such datagrams.

There are certain special cases for IP addresses, defined in the
latest Assigned Numbers document [23]. These special cases can be
concisely summarized using the earlier notation for an IP address:

     IP-address ::= { <Network-number>, <Host-number> }

        or

     IP-address ::= { <Network-number>, <Subnet-number>,
                                            <Host-number> }

if we also use the notation "-1" to mean the field contains all 1
bits. Some common special cases are as follows:

     (a)   {0, 0}

          This host on this network. Can only be used as a source
          address (see note later).

     (b)   {0, <Host-number>}

          Specified host on this network. Can only be used as a
          source address.

     (c)   { -1, -1}

          Limited broadcast. Can only be used as a destination
          address, and a datagram with this address must never be
          forwarded outside the (sub-)net of the source.

     (d)   {<Network-number>, -1}

          Directed broadcast to specified network. Can only be used
          as a destination address.

     (e)   {<Network-number>, <Subnet-number>, -1}

          Directed broadcast to specified subnet. Can only be used as
          a destination address.

     (f)   {<Network-number>, -1, -1}

Directed broadcast to all subnets of specified subnetted
network. Can only be used as a destination address.

(g)   {127, <any>}

Internal host loopback address. Should never appear outside
a host.

The following two are conventional notation for network numbers,
and do not really represent IP addresses. They can never be used
in an IP datagram header as an IP source or destination address.

(h)   {<Network-number>, 0}

Specified network (no host).

(i)   {<Network-number>, <Subnet-number>, 0}

Specified subnet (no host).

Note also that the IP broadcast address, which has primary
application to Ethernets and similar technologies that support an
inherent broadcast function, has an all-ones value in the host
field of the IP address. Some early implementations chose the
all-zeros value for this purpose, which is not in conformance with
the specification [23, 49, 50].

2.2. Internet Control Message Protocol (ICMP)

ICMP is an auxiliary protocol used to convey advice and error
messages and is described in RFC-792 [2].

We will discuss issues arising from gateway handling of particular
ICMP messages. The ICMP messages are grouped into two classes:
error messages and information messages. ICMP error messages are
never sent about ICMP error messages, nor about broadcast or
multicast datagrams.

The ICMP error messages are: Destination Unreachable, Redirect,
Source Quench, Time Exceeded, and Parameter Problem.

The ICMP information messages are: Echo, Information,
Timestamp, and Address Mask.

2.2.1. Destination Unreachable

The distinction between subnets of a subnetted network, which
depends on the address mask described in RFC-950 [21], must not
be visible outside that network. This distinction is important
in the case of the ICMP Destination Unreachable message.

The ICMP Destination Unreachable message is sent by a gateway
in response to a datagram which it cannot forward because the
destination is unreachable or down. The gateway chooses one of
the following two types of Destination Unreachable messages to
send:

   * Net Unreachable

   * Host Unreachable

Net unreachable implies that an intermediate gateway was unable
to forward a datagram, as its routing data-base gave no next
hop for the datagram, or all paths were down. Host Unreachable
implies that the destination network was reachable, but that a
gateway on that network was unable to reach the destination
host. This might occur if the particular destination network
was able to determine that the desired host was unreachable or
down. It might also occur when the destination host was on a
subnetted network and no path was available through the subnets
of this network to the destination. Gateways should send Host
Unreachable messages whenever other hosts on the same
destination network might be reachable; otherwise, the source
host may erroneously conclude that ALL hosts on the network are
unreachable, and that may not be the case.

2.2.2. Redirect

The ICMP Redirect message is sent by a gateway to a host on the
same network, in order to change the gateway used by the host
for routing certain datagrams. A choice of four types of
Redirect messages is available to specify datagrams destined
for a particular host or network, and possibly with a
particular type-of-service.

If the directly-connected network is not subnetted, a gateway
can normally send a network Redirect which applies to all hosts
on a specified remote network. Using a network rather than a
host Redirect may economize slightly on network traffic and on
host routing table storage. However, the saving is not
significant, and subnets create an ambiguity about the subnet

mask to be used to interpret a network Redirect. In a general
subnet environment, it is difficult to specify precisely the
cases in which network Redirects can be used.

Therefore, it is recommended that a gateway send only host (or
host and type-of-service) Redirects.

2.2.3. Source Quench

All gateways must contain code for sending ICMP Source Quench
messages when they are forced to drop IP datagrams due to
congestion. Although the Source Quench mechanism is known to
be an imperfect means for Internet congestion control, and
research towards more effective means is in progress, Source
Quench is considered to be too valuable to omit from production
gateways.

There is some argument that the Source Quench should be sent
before the gateway is forced to drop datagrams [62]. For
example, a parameter X could be established and set to have
Source Quench sent when only X buffers remain. Or, a parameter
Y could be established and set to have Source Quench sent when
only Y per cent of the buffers remain.

Two problems for a gateway sending Source Quench are: (1) the
consumption of bandwidth on the reverse path, and (2) the use
of gateway CPU time. To ameliorate these problems, a gateway
must be prepared to limit the frequency with which it sends
Source Quench messages. This may be on the basis of a count
(e.g., only send a Source Quench for every N dropped datagrams
overall or per given source host), or on the basis of a time
(e.g., send a Source Quench to a given source host or overall
at most once per T millseconds). The parameters (e.g., N or T)
must be settable as part of the configuration of the gateway;
furthermore, there should be some configuration setting which
disables sending Source Quenches. These configuration
parameters, including disabling, should ideally be specifiable
separately for each network interface.

Note that a gateway itself may receive a Source Quench as the
result of sending a datagram targeted to another gateway. Such
datagrams might be an EGP update, for example.

2.2.4. Time Exceeded

The ICMP Time Exceeded message may be sent when a gateway
discards a datagram due to the TTL being reduced to zero. It

may also be sent by a gateway if the fragments of a datagram
addressed to the gateway itself cannot be reassembled before
the time limit.

2.2.5. Parameter Problem

The ICMP Parameter Problem message may be sent to the source
host for any problem not specifically covered by another ICMP
message.

2.2.6. Address Mask

Host and gateway implementations are expected to support the
ICMP Address Mask messages described in RFC-950 [21].

2.2.7. Timestamp

The ICMP Timestamp message has proven to be useful for
diagnosing Internet problems. The preferred form for a
timestamp value, the "standard value", is in milliseconds since
midnight GMT. However, it may be difficult to provide this
value with millisecond resolution. For example, many systems
use clocks which update only at line frequency, 50 or 60 times
per second. Therefore, some latitude is allowed in a
"standard" value:

  * The value must be updated at a frequency of at least 30
     times per second (i.e., at most five low-order bits of
     the value may be undefined).

  * The origin of the value must be within a few minutes of
     midnight, i.e., the accuracy with which operators
     customarily set CPU clocks.

To meet the second condition for a stand-alone gateway, it will
be necessary to query some time server host when the gateway is
booted or restarted. It is recommended that the UDP Time
Server Protocol [44] be used for this purpose. A more advanced
implementation would use NTP (Network Time Protocol) [45] to
achieve nearly millisecond clock synchronization; however, this
is not required.

Even if a gateway is unable to establish its time origin, it
ought to provide a "non-standard" timestamp value (i.e., with
the non-standard bit set), as a time in milliseconds from
system startup.

New gateways, especially those expecting to operate at T1 or
higher speeds, are expected to have at least millisecond
clocks.

2.2.8. Information Request/Reply

The Information Request/Reply pair was intended to support
self-configuring systems such as diskless workstations, to
allow them to discover their IP network numbers at boot time.
However, the Reverse ARP (RARP) protocol [15] provides a better
mechanism for a host to use to discover its own IP address, and
RARP is recommended for this purpose. Information
Request/Reply need not be implemented in a gateway.

2.2.9. Echo Request/Reply

A gateway must implement ICMP Echo, since it has proven to be
an extremely useful diagnostic tool. A gateway must be
prepared to receive, reassemble, and echo an ICMP Echo Request
datagram at least as large as the maximum of 576 and the MTU's
of all of the connected networks. See the discussion of IP
reassembly in gateways, Section 2.1.

The following rules resolve the question of the use of IP
source routes in Echo Request and Reply datagrams. Suppose a
gateway D receives an ICMP Echo Request addressed to itself
from host S.

1. If the Echo Request contained no source route, D should
    send an Echo Reply back to S using its normal routing
    rules. As a result, the Echo Reply may take a different
    path than the Request; however, in any case, the pair
    will sample the complete round-trip path which any other
    higher-level protocol (e.g., TCP) would use for its data
    and ACK segments between S and D.

2. If the Echo Request did contain a source route, D should
    send an Echo Reply back to S using as a source route the
    return route built up in the source-routing option of
    the Echo Request.

   2.3. Exterior Gateway Protocol (EGP)

      EGP is the protocol used to exchange reachability information
      between Autonomous Systems of gateways, and is defined in
      RFC-904 [11]. See also RFC-827 [51], RFC-888 [46], and
      RFC-975 [27] for background information. The most widely used EGP
      implementation is described in RFC-911 [13].

      When a dynamic routing algorithm is operated in the gateways of an
      Autonomous System (AS), the routing data-base must be coupled to
      the EGP implementation. This coupling should ensure that, when a
      net is determined to be unreachable by the routing algorithm, the
      net will not be declared reachable to other ASs via EGP. This
      requirement is designed to minimize spurious traffic to "black
      holes" and to ensure fair utilization of the resources on other
      systems.

      The present EGP specification defines a model with serious
      limitations, most importantly a restriction against propagating
      "third party" EGP information in order to prevent long-lived
      routing loops [27]. This effectively limits EGP to a two-level
      hierarchy; the top level is formed by the "core" AS, while the
      lower level is composed of those ASs which are direct neighbor
      gateways to the core AS. In practice, in the current Internet,
      nearly all of the "core gateways" are connected to the ARPANET,
      while the lower level is composed of those ASs which are directly
      gatewayed to the ARPANET or MILNET.

      RFC-975 [27] suggested one way to generalize EGP to lessen these
      topology restrictions; it has not been adopted as an official
      specification, although its ideas are finding their way into the
      new EGP developments. There are efforts underway in the research
      community to develop an EGP generalization which will remove these
      restrictions.

      In EGP, there is no standard interpretation (i.e., metric) for the
      distance fields in the update messages, so distances are
      comparable only among gateways of the same AS. In using EGP data,
      a gateway should compare the distances among gateways of the same
      AS and prefer a route to that gateway which has the smallest
      distance value.

      The values to be announced in the distance fields for particular
      networks within the local AS should be a gateway configuration
      parameter; by suitable choice of these values, it will be possible
      to arrange primary and backup paths from other AS's. There are
      other EGP parameters, such as polling intervals, which also need
      to be set in the gateway configuration.

When routing updates become large they must be transmitted in
parts. One strategy is to use IP fragmentation, another is to
explicitly send the routing information in sections. The Internet
Engineering Task Force is currently preparing a recommendation on
this and other EGP engineering issues.

2.4. Address Resolution Protocol (ARP)

ARP is an auxiliary protocol used to perform dynamic address
translation between LAN hardware addresses and Internet addresses,
and is described in RFC-826 [4].

ARP depends upon local network broadcast. In normal ARP usage,
the initiating host broadcasts an ARP Request carrying a target IP
address; the corresponding target host, recognizing its own IP
address, sends back an ARP Reply containing its own hardware
interface address.

A variation on this procedure, called "proxy ARP", has been used
by gateways attached to broadcast LANs [14]. The gateway sends an
ARP Reply specifying its interface address in response to an ARP
Request for a target IP address which is not on the
directly-connected network but for which the gateway offers an
appropriate route. By observing ARP and proxy ARP traffic, a
gateway may accumulate a routing data-base [14].

Proxy ARP (also known in some quarters as "promiscuous ARP" or
"the ARP hack") is useful for routing datagrams from hosts which
do not implement the standard Internet routing rules fully -- for
example, host implementations which predate the introduction of
subnetting. Proxy ARP for subnetting is discussed in detail in
RFC-925 [14].

Reverse ARP (RARP) allows a host to map an Ethernet interface
address into an IP address [15]. RARP is intended to allow a
self-configuring host to learn its own IP address from a server at
boot time.

2.5. Constituent Network Access Protocols

See Section 3.

   2.6. Interior Gateway Protocols

      Distributed routing algorithms continue to be the subject of
      research and engineering, and it is likely that advances will be
      made over the next several years. A good algorithm needs to
      respond rapidly to real changes in Internet connectivity, yet be
      stable and insensitive to transients. It needs to synchronize the
      distributed data-base across gateways of its Autonomous System
      rapidly (to avoid routing loops), while consuming only a small
      fraction of the available bandwidth.

      Distributed routing algorithms are commonly broken down into the
      following three components:

         A. An algorithm to assign a "length" to each Internet path.

             The "length" may be a simple count of hops (1, or infinity
             if the path is broken), or an administratively-assigned
             cost, or some dynamically-measured cost (usually an average
             delay).

             In order to determine a path length, each gateway must at
             least test whether each of its neighbors is reachable; for
             this purpose, there must be a "reachability" or "neighbor
             up/down" protocol.

         B. An algorithm to compute the shortest path(s) to a given
             destination.

         C. A gateway-gateway protocol used to exchange path length and
             routing information among gateways.

      The most commonly-used IGPs in Internet gateways are as follows.

      2.6.1. Gateway-to-Gateway Protocol (GGP)

         GGP was designed and implemented by BBN for the first
         experimental Internet gateways [41]. It is still in use in the
         BBN LSI/11 gateways, but is regarded as having serious
         drawbacks [58]. GGP is based upon an algorithm used in the
         early ARPANET IMPs and later replaced by SPF (see below).

         GGP is a "min-hop" algorithm, i.e., its length measure is
         simply the number of network hops between gateway pairs. It
         implements a distributed shortest-path algorithm, which
         requires global convergence of the routing tables after a
         change in topology or connectivity. Each gateway sends a GGP

routing update only to its neighbors, but each update includes
an entry for every known network, where each entry contains the
hop count from the gateway sending the update.

2.6.2. Shortest-Path-First (SPF) Protocols

SPF [40] is the name for a class of routing algorithms based on
a shortest-path algorithm of Dijkstra. The current ARPANET
routing algorithm is SPF, and the BBN Butterfly gateways also
use SPF. Its characteristics are considered superior to
GGP [58].

Under SPF, the routing data-base is replicated rather than
distributed. Each gateway will have its own copy of the same
data-base, containing the entire Internet topology and the
lengths of every path. Since each gateway has all the routing
data and runs a shortest-path algorithm locally, there is no
problem of global convergence of a distributed algorithm, as in
GGP. To build this replicated data-base, a gateway sends SPF
routing updates to ALL other gateways; these updates only list
the distances to each of the gateway's neighbors, making them
much smaller than GGP updates. The algorithm used to
distribute SPF routing updates involves reliable flooding.

2.6.3. Routing Information (RIP)

RIP is the name often used for a class of routing protocols
based upon the Xerox PUP and XNS routing protocols. These are
relatively simple, and are widely available because they are
incorporated in the embedded gateway code of Berkeley BSD
systems. Because of this simplicity, RIP protocols have come
the closest of any to being an "Open IGP", i.e., a protocol
which can be used between different vendors' gateways.
Unfortunately, there is no standard, and in fact not even a
good document, for RIP.

As in GGP, gateways using RIP periodically broadcast their
routing data-base to their neighbor gateways, and use a
hop-count as the metric.

A fixed value of the hop-count (normally 16) is defined to be
"infinity", i.e., network unreachable. A RIP implementation
must include measures to avoid both the slow-convergence
phenomen called "counting to infinity" and the formation of
routing loops. One such measure is a "hold-down" rule. This
rule establishes a period of time (typically 60 seconds) during
which a gateway will ignore new routing information about a
given network, once the gateway has learned that network is

unreachable (has hop-count "infinity"). The hold-down period
must be settable in the gateway configuration; if gateways with
different hold-down periods are using RIP in the same
Autonomous System, routing loops are a distinct possibility.
In general, the hold-down period is chosen large enough to
allow time for unreachable status to propagate to all gateways
in the AS.

2.6.4. Hello

The "Fuzzball" software for an LSI/11 developed by Dave Mills
incorporated an IGP called the "Hello" protocol [39]. This IGP
is mentioned here because the Fuzzballs have been widely used
in Internet experimentation, and because they have served as a
testbed for many new routing ideas.

2.7. Monitoring Protocols

See Section 5 of this document.

2.8. Internet Group Management Protocol (IGMP)

An extension to the IP protocol has been defined to provide
Internet-wide multicasting, i.e., delivery of copies of the same
IP datagram to a set of Internet hosts [47, 48]. This delivery is
to be performed by processes known as "multicasting agents", which
reside either in a host on each net or (preferably) in the
gateways.

The set of hosts to which a datagram is delivered is called a
"host group", and there is a host-agent protocol called IGMP,
which a host uses to join, leave, or create a group. Each host
group is distinguished by a Class D IP address.

This multicasting mechanism and its IGMP protocol are currently
experimental; implementation in vendor gateways would be premature
at this time. A datagram containing a Class D IP address must be
dropped, with no ICMP error message.

3. Constituent Network Interface

   This section discusses the rules used for transmission of IP
   datagrams on the most common types of constituent networks. A
   gateway must be able to send and receive IP datagrams of any size up
   to the MTU of any constituent network to which it is connected.

   3.1. Public data networks via X.25

      The formats specified for public data networks accessed via X.25
      are described in RFC-877 [8]. Datagrams are transmitted over
      standard level-3 virtual circuits as complete packet sequences.
      Virtual circuits are usually established dynamically as required
      and time-out after a period of no traffic. Link-level
      retransmission, resequencing and flow control are performed by the
      network for each virtual circuit and by the LAPB link-level
      protocol. Note that a single X.25 virtual circuit may be used to
      multiplex all IP traffic between a pair of hosts. However,
      multiple parallel virtual circuits may be used in order to improve
      the utilization of the subscriber access line, in spite of small
      X.25 window sizes; this can result in random resequencing.

      The correspondence between Internet and X.121 addresses is usually
      established by table-lookup. It is expected that this will be
      replaced by some sort of directory procedure in the future. The
      table of the hosts on the Public Data Network is in the Assigned
      Numbers [23].

      The normal MTU is 576; however, the two DTE's (hosts or gateways)
      can use X.25 packet size negotiation to increase this value [8].

   3.2. ARPANET via 1822 LH, DH, or HDH

      The formats specified for ARPANET networks using 1822 access are
      described in BBN Report 1822 [3], which includes the procedures
      for several subscriber access methods. The Distant Host (DH)
      method is used when the host and IMP (the Defense Communication
      Agency calls it a Packet Switch Node or PSN) are separated by not
      more than about 2000 feet of cable, while the HDLC Distant Host
      (HDH) is used for greater distances where a modem is required.
      Under HDH, retransmission, resequencing and flow control are
      performed by the network and by the HDLC link-level protocol.

      The IP encapsulation format is simply to include the IP datagram
      as the data portion of an 1822 message. In addition, the
      high-order 8 bits of the Message Id field (also known as the
      "link" field") should be set to 155 [23]. The MTU is 1007 octets.

While the ARPANET 1822 protocols are widely used at present, they
are expected to be eventually overtaken by the DDN Standard X.25
protocol (see Section 3.3). The original IP address mapping
(RFC-796 [38]) is in the process of being replaced by a new
interface specification called AHIP-E; see RFC-1005 [61] for the
proposal.

Gateways connected to ARPANET or MILNET IMPs using 1822 access
must incorporate features to avoid host-port blocking (i.e., RFNM
counting) and to detect and report as ICMP Unreachable messages
the failure of destination hosts or gateways (i.e., convert the
1822 error messages to the appropriate ICMP messages).

In the development of a network interface it will be useful to
review the IMP end-to-end protocol described in RFC-979 [29].

3.3.   ARPANET via DDN Standard X.25

The formats specified for ARPANET networks via X.25 are described
in the Defense Data Network X.25 Host Interface Specification [6],
which describes two sets of procedures: the DDN Basic X.25, and
the DDN Standard X.25. Only DDN Standard X.25 provides the
functionality required for interoperability assumptions of the
Internet protocol.

The DDN Standard X.25 procedures are similar to the public data
network X.25 procedures, except in the address mappings.
Retransmission, resequencing and flow control are performed by the
network and by the LAPB link-level protocol. Multiple parallel
virtual circuits may be used in order to improve the utilization
of the subscriber access line; this can result in random
resequencing.

Gateways connected to ARPANET or MILNET using Standard X.25 access
must detect and report as ICMP Unreachable messages the failure of
destination hosts or gateways (i.e., convert the X.25 diagnostic
codes to the appropriate ICMP messages).

To achieve compatibility with 1822 interfaces, the effective MTU
for a Standard X.25 interface is 1007 octets.

   3.4. Ethernet and IEEE 802

      The formats specified for Ethernet networks are described in
      RFC-894 [10]. Datagrams are encapsulated as Ethernet packets with
      48-bit source and destination address fields and a 16-bit type
      field (the type field values are listed in the Assigned
      Numbers [23]). Address translation between Ethernet addresses and
      Internet addresses is managed by the Address Resolution Protocol,
      which is required in all Ethernet implementations. There is no
      explicit link-level retransmission, resequencing or flow control,
      although most hardware interfaces will retransmit automatically in
      case of collisions on the cable.

      The IEEE 802 networks use a Link Service Access Point (LSAP) field
      in much the same way the ARPANET uses the "link" field. Further,
      there is an extension of the LSAP header called the Sub-Network
      Access Protocol (SNAP).

      The 802.2 encapsulation is used on 802.3, 802.4, and 802.5 network
      by using the SNAP with an organization code indicating that the
      following 16 bits specify the Ether-Type code [23].

      Headers:

```
         ...--------+--------+--------+
          MAC Header|    Length       |                     802.{3/4/5} MAC
         ...--------+--------+--------+


         +--------+--------+--------+
         | DSAP=K1| SSAP=K1| control|                       802.2 SAP
         +--------+--------+--------+


         +--------+--------+--------+--------+--------+
         |protocol id or org code=K2|    Ether-Type    |    802.2 SNAP
         +--------+--------+--------+--------+--------+
```

      The total length of the SAP Header and the SNAP header is
      8-octets, making the 802.2 protocol overhead come out on a 64-bit
      boundary.

      K1 is 170. The IEEE likes to talk about things in bit
      transmission order and specifies this value as 01010101. In
      big-endian order, as used in the Internet specifications, this
      becomes 10101010 binary, or AA hex, or 170 decimal. K2 is 0
      (zero).

      The use of the IP LSAP (K1 = 6) is reserved for future
      development.

The assigned values for the Ether-Type field are the same for
either this IEEE 802 encapsulation or the basic Ethernet
encapsulation [10].

In either Ethernets or IEEE 802 nets, the IP datagram is the data
portion of the packet immediately following the Ether-Type.

The MTU for an Ethernet or its IEEE-standard equivalent (802.3) is
1500 octets.

3.5. Serial-Line Protocols

In some configurations, gateways may be interconnected with each
other by means of serial asynchronous or synchronous lines, with
or without modems. When justified by the expected error rate and
other factors, a link-level protocol may be required on the serial
line. While there is no single Internet standard for this
protocol, it is suggested that one of the following protocols be
used.

   * X.25 LAPB (Synchronous Lines)

     This is the link-level protocol used for X.25 network
     access. It includes HDLC "bit-stuffing" as well as
     rotating-window flow control and reliable delivery.

        A gateway must be configurable to play the role of either
        the DCE or the DTE.

   * HDLC Framing (Synchronous Lines)

     This is just the bit-stuffing and framing rules of LAPB. It
     is the simplest choice, although it provides no flow control
     or reliable delivery; however, it does provide error
     detection.

   * Xerox Synchronous Point-to-Point (Synchronous Lines)

     This Xerox protocol is an elaboration upon HDLC framing that
     includes negotiation of maximum packet sizes, dial-up or
     dedicated circuits, and half- or full-duplex operation [12].

   * Serial Line Framing Protocol (Asynchronous Lines)

     This protocol is included in the MIT PC/IP package for an
     IBM PC and is defined in Appendix I to the manual for that
     system [20].

It will be important to make efficient use of the bandwidth available on a serial line between gateways. For example, it is desirable to provide some form of data compression. One possible standard compression algorithm, "Thinwire II", is described in RFC-914 [42]. This and similar algorithms are tuned to the particular types of redundancy which occur in IP and TCP headers; however, more work is necessary to define a standard serial-line compression protocol for Internet gateways. Until a standard has been adopted, each vendor is free to choose a compression algorithm; of course, the result will only be useful on a serial line between two gateways using the same compression algorithm.

Another way to ensure maximum use of the bandwidth is to avoid unnecessary retransmissions at the link level. For some kinds of IP traffic, low delay is more important than reliable delivery. The serial line driver could distinguish such datagrams by their IP TOS field, and place them on a special high-priority, no-retransmission queue.

A serial point-to-point line between two gateways may be considered to be a (particularly simple) network, a "null net". Considered in this way, a serial line requires no special considerations in the routing algorithms of the connected gateways, but does need an IP network number. To avoid the wholesale consumption of Internet routing data-base space by null nets, we strongly recommend that subnetting be used for null net numbering, whenever possible.

     For example, assume that network 128.203 is to be constructed
     of gateways joined by null nets; these nets are given (sub-)net
     numbers 128.203.1, 128.203.2, etc., and the two interfaces on
     each end of null net 128.203.s might have IP addresses
     128.203.s.1 and 128.203.s.2.

An alternative model of a serial line is that it is not a network, but rather an internal communication path joining two "half gateways". It is possible to design an IGP and routing algorithm that treats a serial line in this manner [39, 52].

4. Gateway Algorithms

   Gateways are general packet-switches that forward packets according
   to the IP address, i.e., they are IP routers.   While it is beyond
   the scope of this document to specify the details of the mechanisms
   used in any particular, perhaps proprietary, gateway architecture,
   there are a number of basic algorithms which must be provided by any
   acceptable design.

   4.1. Routing Algorithm

      The routing mechanism is fundamental to Internet operation. In
      all but trivial network topologies, robust Internet service
      requires some degree of routing dynamics, whether it be effected
      by manual or automatic means or by some combination of both. In
      particular, if routing changes are made manually, it must be
      possible to make these routing changes from a remote Network
      Operation Center (NOC) without taking down the gateway for
      reconfiguration. If static routes are used, there must be
      automatic fallback or rerouting features.

      Handling unpredictable changes in Internet connectivity must be
      considered the normal case, so that systems of gateways will
      normally be expected to have a routing algorithm with the
      capability of reacting to link and other gateway failures and
      changing the routing automatically.

      This document places no restriction on the type of routing
      algorithm, e.g., node-based, link-based or any other algorithm, or
      on the routing distance metric, e.g., delay or hop-count.
      However, the following features are considered necessary for a
      successful gateway routing algorithm:

         1. The algorithm must sense the failure or restoration of a
             link or other gateway and switch to appropriate paths. A
             design objective is to switch paths within an interval less
             than the typical TCP user time-out (one minute is a safe
             assumption).

         2. The algorithm must suppress routing loops between neighbor
             gateways and must contain provisions to avoid or suppress
             routing loops that may form between non-neighbor gateways.
             A design objective is for no loop to persist for longer
             than an interval greater than the typical TCP user
             time-out.

         3. The control traffic necessary to operate the routing
             algorithm must not significantly degrade or disrupt normal

network operation. Changes in state which might
momentarily disrupt normal operation in a local-area must
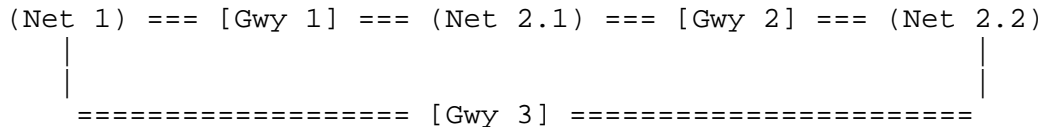not cause disruption in remote areas of the network.

4. As the size of the network increases, the demand on
    resources must be controlled in an efficient way. Table
    lookups should be hashed, for example, and data-base
    updates handled piecemeal, with only incremental changes
    broadcast over a wide-area.

5. The size of the routing data-base must not be allowed to
    exceed a constant, independent of network topology, times
    the number of nodes times the mean connectivity (average
    number of incident links). An advanced design might not
    require that the entire routing data-base be kept in any
    particular gateway, so that discovery and caching
    techniques would be necessary.

6. Reachability and delay metrics, if used, must not depend on
    direct connectivity to all other gateways or on the use of
    network-specific broadcast mechanisms. Polling procedures
    (e.g., for consistency checking) must be used only
    sparingly and in no case introduce an overhead exceeding a
    constant, independent of network topology, times the
    longest non-looping path.

7. Default routes (generally intended as a means to reduce the
    size of the routing data-base) must be used with care,
    because of the many problems with multiple paths, loops,
    and mis-configurations which routing defaults have caused.

    The most common application of defaults is for routing
    within an Internet region which is connected in a strictly
    hierarchical fashion and is a stub from the rest of the
    Internet system. In this case, the default is used for
    routing "up" the tree. Unfortunately, such restricted
    topology seldom lasts very long, and defaults cease to
    work.

    More generally, defaults could be used for initial routing
    guesses, with final routes to be discovered and cached from
    external or internal data-bases via the routing algorithm
    or EGP.

   4.2. Subnets and Routing

      We will call a gateway "subnetted" if at least one of its
      interfaces is connected to a subnet; the set of gateways directly
      connected to subnets of the same network will be referred to as a
      "subnet cluster". For example, in the following diagram, network
      2 is subnetted, with subnets 2.1 and 2.2, but network 1 is not;
      gateways 1, 2, and 3 are subnetted and are members of the same
      subnet cluster.

         (Net 1) === [Gwy 1] === (Net 2.1) === [Gwy 2] === (Net 2.2)
             | |
             | |
             ================== [Gwy 3] =======================

      Subnets have the following effects on gateway routing:

         A. Non-subnetted gateways are not affected at all.

         B. The routing data-base in a subnetted gateway must consider
             the address mask for subnet entries.

         C. Routing updates among the gateways in the same subnet
             cluster must include entries for the various subnets. The
             corresponding address mask(s) may be implicit, but for full
             generality the mask needs to be given explicitly for each
             entry. Note that if the routing data-base included a full
             32-bit mask for every IP network, the gateway could deal
             with networks and subnets in a natural way. This would
             also handle the case of multiple subnet masks for the same
             subnetted network.

         D. Routing updates from a subnetted gateway to a gateway
             outside the cluster can contain nets, never subnets.

         E. If a subnetted gateway (e.g., gateway 2 above) is unable to
             forward a datagram from one subnet to another subnet of the
             same network, then it must return a Host Unreachable, not a
             Net Unreachable, as discussed in Section 2.2.1.

      When considering the choice of routing protocol, a gateway builder
      must consider how that protocol generalizes for subnets. For some
      routing protocols it will be possible to use the same procedures
      in a regular gateway and a subnetted gateway, with only a change
      of parameters (e.g., address masks).

      A different subnet address mask must be configurable for each

interface of a given gateway. This will allow a subnetted gateway
to connect to two different subnetted networks, or to connect two
subnets of the same network with different masks.

4.3   Resource Allocation

In order to perform its basic datagram-forwarding functions, a
gateway must allocate resources; its packet buffers and CPU time
must be allocated to packets it receives from connected networks,
while the bandwidth to each of the networks must also be allocated
for sending packets. The choice of allocation strategies will be
critical when a particular resource is scarce. The most obvious
allocation strategy, first-come-first-served (FCFS), may not be
appropriate under overload conditions, for reasons which we will
now explore.

A first example is buffer allocation. It is important for a
gateway to allocate buffers fairly among all of its connected
networks, even if these networks have widely varying bandwidths.
A high-speed interface must not be allowed to starve slower
interfaces of buffers. For example, consider a gateway with a
10 Mbps Ethernet connection and two 56 Kbps serial lines. A buggy
host on the Ethernet may spray that gateway interface with packets
at high speed. Without careful algorithm design in the gateway,
this could tie up all the gateway buffers in such a way that
transit traffic between the serial lines would be completely
stopped.

Allocation of output bandwidth may also require non-FCFS
strategies. In an advanced gateway design, allocation of output
bandwidth may depend upon Type-of-Service bits in the IP headers.
A gateway may also want to give priority to datagrams for its own
up/down and routing protocols.

Finally, Nagle [24] has suggested that gateways implement "fair
queueing", i.e., sharing output bandwidth equitably among the
current traffic sources. In his scheme, for each network
interface there would be a dynamically-built set of output queues,
one per IP source address; these queues would be serviced in a
round-robin fashion to share the bandwidth. If subsequent
research shows fair queueing to be desirable, it will be added to
a future version of this document as a universal requirement.

   4.4. Special Addresses and Filters

      Section 2.1 contained a list of the 32-bit IP addresses which have
      special meanings. They do not in general represent unique IP
      addresses of Internet hosts, and there are restrictions on their
      use in IP headers.

      We can distinguish two classes of these special cases. The first
      class (specifically, cases (a), (b), (c), (g), (h), and (i) in
      section 2.1) contains addresses which should never appear in the
      destination address field of any IP datagram, so a gateway should
      never be asked to route to one of these addresses. However, in
      the real world of imperfect implementations and configuration
      errors, such bad destination addresses do occur. It is the
      responsibility of a gateway to avoid propagating such erroneous
      addresses; this is especially important for gateways included in
      the global interconnect system. In particular, a gateway which
      receives a datagram with one of these forbidden addresses should:

         1. Avoid inserting that address into its routing database, and
             avoid including it in routing updates to any other gateway.

         2. Avoid forwarding a datagram containing that address as a
             destination.

      To enforce these restrictions, it is suggested that a gateway
      include a configurable filter for datagrams and routing updates.
      A typical filter entry might consist of a 32-bit mask and value
      pair. If the logical AND of the given address with the mask
      equals the value, a match has been found. Since filtering will
      consume gateway resources, it is vital that the gateway
      configuration be able to control the degree of filtering in use.

      There is a second class of special case addresses (cases (d), (e),
      and (f) in section 2.1), the so-called "directed broadcasts". A
      directed broadcast is a datagram to be forwarded normally to the
      specified destination (sub-)net and then broadcast on the final
      hop. An Internet gateway is permitted, but not required, to
      filter out directed broadcasts destined for any of its
      locally-connected networks. Hence, it should be possible to
      configure the filter to block the delivery of directed broadcasts.

      Finally, it will also be useful for Internet O&M to have a
      configurable filter on the IP source address. This will allow a
      network manager to temporarily block traffic from a particular
      misbehaving host, for example.

4.5. Redirects

   The ICMP Redirect message is specified only for use by a gateway
   to update the routing table of a host on the same connected net.
   However, the Redirect message is sometimes used between gateways,
   due to the following considerations:

      The routing function in a host is very much like that in a
      "dumb gateway" (i.e., a gateway having only static routes). It
      is desirable to allow the routing tables of a dumb gateway to
      be changed under the control of a dynamic gateway (i.e., a
      gateway with full dynamic routing) on the same network. By
      analogy, it is natural to let the dynamic gateway send ICMP
      Redirect messages to dumb gateway.

   The use of ICMP Redirect between gateways in this fashion may be
   considered to be part of the IGP (in fact, the totality of the
   IGP, as far as the dumb gateway is concerned!) in the particular
   Autonomous System.   Specification of an IGP is outside the scope
   of this document, so we only note the possibility of using
   Redirect in this fashion. Gateways are not required to receive
   and act upon redirects, and in fact dynamic gateways must ignore
   them. We also note that considerable experience shows that dumb
   gateways often create problems resulting in "black holes"; a full
   routing gateway is always preferable.

   Routing table entries established by redirect messages must be
   removed automatically, either by a time-out or when a use count
   goes to zero.

4.6. Broadcast and Multicast

   A host which is connected to a network (generally a LAN) with an
   intrinsic broadcast capability may want to use this capability to
   effect multidestination delivery of IP datagrams. The basic
   Internet model assumes point-to-point messages, and we must take
   some care when we incorporate broadcasting. It is important to
   note that broadcast addresses may occur at two protocol levels:
   the local network header and the IP header.

   Incorrect handling of broadcasting has often been the cause of
   packet avalanches (sometimes dubbed "meltdown") in LANs. These
   avalanches are generally caused by gratuitous datagram-forwarding
   by hosts, or by hosts sending ICMP error messages when they
   discard broadcast datagrams.

   Gateways have a responsibility to prevent avalanches, or datagrams
   which can trigger avalanches, from escaping into another network.

In general, a gateway must not forward a datagram which arrives
via local network broadcast, and must not send an ICMP error
message when dropping the datagram. A discussion of the rules
will be found in Appendix A; see also [50].

As noted in Section 4.4, a gateway is permitted to filter out
directed broadcasts. Hence, directed broadcasts will only be
useful in limited Internet regions (e.g., the within the subnets
of a particular campus) in which delivery is supported by the
gateway administrators. Host group multicasting (see Sections 2.8
and 4.6) will soon provide a much more efficient mechanism than
directed broadcasting. Gateway algorithms for host group
multicasting will be specified in future RFC's.

4.7. Reachability Procedures

The architecture must provide a robust mechanism to establish the
operational status of each link and node in the network, including
the gateways, the links connecting them and, where appropriate,
the hosts as well. Ordinarily, this requires at least a
link-level reachability protocol involving a periodic exchange of
messages across each link. This function might be intrinsic to
the link-level protocols used (e.g., LAPB). However, it is in
general ill-advised to assume a host or gateway is operating
correctly even if its link-level reachability protocol is
operating correctly. Additional confirmation is required in the
form of an operating routing algorithm or peer-level reachability
protocol (such as used in EGP).

Failure and restoration of a link and/or gateway are considered
network events and must be reported to the control center. It is
desirable, although not required, that reporting paths not require
correct functioning of the routing algorithm itself.

4.8. Time-To-Live

The Time-to-Live (TTL) field of the IP header is defined to be a
timer limiting the lifetime of a datagram in the Internet. It is
an 8-bit field and the units are seconds. This would imply that
for a maximum TTL of 255 a datagram would time-out after about 4
and a quarter minutes. Another aspect of the definition requires
each gateway (or other module) that handles a datagram to
decrement the TTL by at least one, even if the elapsed time was
much less than a second. Since this is very often the case, the
TTL effectively becomes a hop count limit on how far a datagram
can propagate through the Internet.

As the Internet grows, the number of hops needed to get from one
edge to the opposite edge increases, i.e., the Internet diameter
grows.

If a gateway holds a datagram for more than one second, it must
decrement the TTL by one for each second.

If the TTL is reduced to zero, the datagram must be discarded, and
the gateway may send an ICMP Time Exceeded message to the source.
A datagram should never be received with a TTL of zero.

When it originates a datagram, a gateway is acting in the role of
a host and must supply a realistic initial value for the TTL.

5. Operation and Maintenance

   5.1. Introduction

      Facilities to support operation and maintenance (O&M) activities
      form an essential part of any gateway implementation. The
      following kinds of activity are included under gateway O&M:

         * Diagnosing hardware problems in the gateway processor, in
            its network interfaces, or in the connected networks,
            modems, or communication lines.

         * Installing a new version of the gateway software.

         * Restarting or rebooting a gateway after a crash.

         * Configuring (or reconfiguring) the gateway.

         * Detecting and diagnosing Internet problems such as
            congestion, routing loops, bad IP addresses, black holes,
            packet avalanches, and misbehaved hosts.

         * Changing network topology, either temporarily (e.g., to
            diagnose a communication line problem) or permanently.

         * Monitoring the status and performance of the gateways and
            the connected networks.

         * Collecting traffic statistics for use in (Inter-)network
            planning.

      Gateways, packet-switches, and their connected communication lines
      are often operated as a system by a centralized O&M organization.
      This organization will maintain a (Inter-)network operation
      center, or NOC, to carry out its O&M functions. It is essential
      that gateways support remote control and monitoring from such a
      NOC, through an Internet path (since gateways might not be
      connected to the same network as their NOC). Furthermore, an IP
      datagram traversing the Internet will often use gateways under the
      control of more than one NOC; therefore, Internet problem
      diagnosis will often involve cooperation of personnel of more than
      one NOC. In some cases, the same gateway may need to be monitored
      by more than one NOC.

      The tools available for monitoring at a NOC may cover a wide range
      of sophistication. Proposals have included multi-window, dynamic
      displays of the entire gateway system, and the use of AI
      techniques for automatic problem diagnosis.

Gateway O&M facilities discussed here are only a part of the large
and difficult problem of Internet management. These problems
encompass not only multiple management organizations, but also
multiple protocol layers. For example, at the current stage of
evolution of the Internet architecture, there is a strong coupling
between host TCP implementations and eventual IP-level congestion
in the gateway system [9]. Therefore, diagnosis of congestion
problems will sometimes require the monitoring of TCP statistics
in hosts. Gateway algorithms also interact with local network
performance, especially through handling of broadcast packets and
ARP, and again diagnosis will require access to hosts (e.g.,
examining ARP caches). However, consideration of host monitoring
is beyond the scope of this RFC.

There are currently a number of R&D efforts in progress in the
area of Internet management and more specifically gateway O&M. It
is hoped that these will lead quickly to Internet standards for
the gateway protocols and facilities required in this area. This
is also an area in which vendor creativity can make a significant
contribution.

5.2.   Gateway O&M Models

There is a range of possible models for performing O&M functions
on a gateway. At one extreme is the local-only model, under which
the O&M functions can only be executed locally, e.g., from a
terminal plugged into the gateway machine. At the other extreme,
the fully-remote model allows only an absolute minimum of
functions to be performed locally (e.g., forcing a boot), with
most O&M being done remotely from the NOC. There intermediate
models, e.g., one in which NOC personnel can log into the gateway
as a host, using the Telnet protocol, to perform functions which
can also be invoked locally. The local-only model may be adequate
in a few gateway installations, but in general remote operation
from a NOC will be required, and therefore remote O&M provisions
are required for most gateways.

Remote O&M functions may be exercised through a control agent
(program). In the direct approach, the gateway would support
remote O&M functions directly from the NOC using standard Internet
protocols (e.g., UDP or TCP); in the indirect approach, the
control agent would support these protocols and control the
gateway itself using proprietary protocols. The direct approach
is preferred, although either approach is acceptable. The use of
specialized host hardware and/or software requiring significant
additional investment is discouraged; nevertheless, some vendors
may elect to provide the control agent as an integrated part of
the network in which the gateways are a part. If this is the

case, it is required that a means be available to operate the
control agent from a remote site using Internet protocols and
paths and with equivalent functionality with respect to a local
agent terminal.

It is desirable that a control agent and any other NOC software
tools which a vendor provides operate as user programs in a
standard operating system. The use of the standard Internet
protocols UDP and TCP for communicating with the gateways should
facilitate this.

Remote gateway monitoring and (especially) remote gateway control
present important access control problems which must be addressed.
Care must also be taken to ensure control of the use of gateway
resources for these functions. It is not desirable to let gateway
monitoring take more than some limited fraction of the gateway CPU
time, for example. On the other hand, O&M functions must receive
priority so they can be exercised when the gateway is congested,
i.e., when O&M is most needed.

There are no current Internet standards for the control and
monitoring protocols, although work is in progress in this area.
The Host Monitoring Protocol (HMP) [7] could be used as a model
until a standard is developed; however, it is strongly recommended
that gateway O&M protocol be built on top of one of the standard
Internet end-to-end protocols UDP or TCP. An example of a very
simple but effective approach to gateway monitoring is contained
in RFC-996 [43].

5.3.   Gateway O&M Functions

The following O&M functions need to be performed in a gateway:

A.   Maintenance -- Hardware Diagnosis

Each gateway must operate as a stand-alone device for the
purposes of local hardware maintenance. Means must be
available to run diagnostic programs at the gateway site
using only on-site tools, which might be only a diskette or
tape and local terminal. It is desirable, although not
required, to be able to run diagnostics or dump the gateway
via the network in case of fault. Means should be provided
to allow remote control from the NOC of of modems attached
to the gateway. The most important modem control capability
is entering and leaving loopback mode, to diagnose line
problems.

B.   Control -- Dumping and Rebooting

It must be possible to dump and reboot a stand-alone gateway
upon command from the NOC. In addition, a stand-alone
gateway must include a watchdog timer that either initiates
a reboot automatically or signals a remote control site if
not reset periodically by the software. It is desirable
that the boot data involved reside at an Internet host
(e.g., the NOC host) and be transmitted via the net;
however, the use of local devices at the gateway site is
acceptable.

C.   Control -- Configuring the Gateway

Every gateway will have a number of configuration parameters
which must be set (see the next section for examples). It
must be possible to update the parameters without rebooting
the gateway; at worst, a restart may be required.

D.   Monitoring -- Status and Performance

A mechanism must be provided for retrieving status and
statistical information from a gateway. A gateway must
supply such information in response to a polling message
from the NOC. In addition, it may be desirable to configure
a gateway to transmit status spontaneously and periodically
to a NOC (or set of NOCs), for recording and display.

Examples of interesting status information include: link
status, queue lengths, buffer availability, CPU and memory
utilization, the routing data-base, error counts, and packet
counts. Counts should be kept for dropped datagrams,
separated by reason. Counts of ICMP datagrams should be
kept by type and categorized into those originating at the
gateway, and those destined for the gateway. It would be
useful to maintain many of these statistics by network
interface, by source/destination network pair, and/or by
source/destination host pair.

Note that a great deal of useful monitoring data is often to
be found in the routing data-base. It is therefore useful
to be able to tap into this data-base from the NOC.

E.   Monitoring -- Error Logging

A gateway should be capable of asynchronously sending
exception ("trap") reports to one or more specified Internet
addresses, one of which will presumably be the NOC host.

There must also be a mechanism to limit the frequency of
such trap reports, and the parameters controlling this
frequency must be settable in the gateway configuration.

Examples of conditions which should result in traps include:
datagrams discarded because of TTL expiration (an indicator
of possible routing loops); resource shortages; or an
interface changing its up/down status.

5.4.   Gateway Configuration Parameters

Every gateway will have a set of configuration parameters
controlling its operation. It must be possible to set these
parameters remotely from the NOC or locally at any time, without
taking the gateway down.

The following is a partial but representative list of possible
configuration parameters for a full-function gateway. The items
marked with "(i)" should be settable independently for each
network interface.

   * (i) IP (sub-) network address

   * (i) Subnet address mask

   * (i) MTU of local network

   * (i) Hardware interface address

   * (i) Broadcast compatibility option (0s or 1s)

   *   EGP parameters -- neighbors, Autonomous System number,
          and polling parameters

   *   Static and/or default routes, if any

   *   Enable/Disable Proxy ARP

   *   Source Quench parameters

   *   Address filter configuration

   *   Boot-host address

   *   IP address of time server host

   *   IP address(es) of logging host(s)

*   IP address(es) of hosts to receive traps

*   IP address(es) of hosts authorized to issue control
          commands

*   Error level for logging

*   Maximum trap frequency

*   Hold-down period (if any)

Appendix A.   Technical Details

   This Appendix collects a number of technical details and rules
   concerning datagram forwarding by gateways and datagram handling by
   hosts, especially in the presence of broadcasting and subnets.

   A.1.   Rules for Broadcasting

      The following rules define how to handle broadcasts of packets and
      datagrams [50]:

         a.   Hosts (which do not contain embedded gateways) must NEVER
              forward any datagrams received from a connected network,
              broadcast or not.

              When a host receives an IP datagram, if the destination
              address identifies the host or is an IP broadcast address,
              the host passes the datagram to its appropriate
              higher-level protocol module (possibly sending ICMP
              protocol unreachable, but not if the IP address was a
              broadcast address). Any other IP datagram must simply be
              discarded, without an ICMP error message. Hosts never send
              redirects.

         b.   All packets containing IP datagrams which are sent to the
              local-network packet broadcast address must contain an IP
              broadcast address as the destination address in their IP
              header. Expressed in another way, a gateway (or host) must
              not send in a local-network broadcast packet an IP datagram
              that has a specific IP host address as its destination
              field.

         c.   A gateway must never forward an IP datagram that arrives
              addressed to the IP limited broadcast address {-1,-1}.
              Furthermore, it must must not send an ICMP error message
              about discarding such a datagram.

         d.   A gateway must not forward an IP datagram addressed to
              network zero, i.e., {0, *}.

         e.   A gateway may forward a directed broadcast datagram, i.e.,
              a datagram with the IP destination address:

              { <Network-number>, -1}.

              However, it must not send such a directed broadcast out the
              same interface it came in, if this interface has
              <Network-number> as its network number. If the code in the

gateway making this decision does not know what interface
the directed-broadcast datagram arrived on, the gateway
cannot support directed broadcast to this connected network
at all.

   f. A gateway is permitted to protect its connected networks by
       discarding directed broadcast datagrams.

A gateway will broadcast an IP datagram on a connected network if
it is a directed broadcast destined for that network. Some
gateway-gateway routing protocols (e.g., RIP) also require
broadcasting routing updates on the connected networks. In either
case, the datagram must have an IP broadcast address as its
destination.

   Note: as observed earlier, some host implementations (those
   based on Berkeley 4.2BSD) use zero rather than -1 in the host
   field. To provide compatibility during the period until these
   systems are fixed or retired, it may be useful for a gateway to
   be configurable to send either choice of IP broadcast address
   and accept both if received.

A.2. ICMP Redirects

A gateway will generate an ICMP Redirect if and only if the
destination IP address is reachable from the gateway (as
determined by the routing algorithm) and the next-hop gateway is
on the same (sub-)network as the source host. Redirects must not
be sent in response to an IP network or subnet broadcast address
or in response to a Class D or Class E IP address.

A host must discard an ICMP Redirect if the destination IP address
is not its own IP address, or the new target address is not on the
same (sub-)network. An accepted Redirect updates the routing
data-base for the old target address. If there is no route
associated with the old target address, the Redirect is ignored.
If the old route is associated with a default gateway, a new route
associated with the new target address is inserted in the
data-base.

Appendix B. NSFNET Specific Requirements

   The following sections discuss certain issues of special concern to
   the NSF scientific networking community. These issues have primary
   relevance in the policy area, but also have ramifications in the
   technical area.

   B.1. Proprietary and Extensibility Issues

      Although hosts, gateways and networks supporting Internet
      technology have been in continuous operation for several years,
      vendors users and operators must understand that not all
      networking issues are fully resolved. As a result, when new needs
      or better solutions are developed for use in the NSF networking
      community, it may be necessary to field new protocols or augment
      existing ones. Normally, these new protocols will be designed to
      interoperate in all practical respects with existing protocols;
      however, occasionally it may happen that existing systems must be
      upgraded to support these new or augmented protocols.

      NSF systems procurements may favor those vendors who undertake a
      commitment to remain aware of current Internet technology and be
      prepared to upgrade their products from time to time as
      appropriate. As a result, vendors are strongly urged to consider
      extensibility and periodic upgrades as fundamental characteristics
      of their products. One of the most productive and rewarding ways
      to do this on a long-term basis is to participate in ongoing
      Internet research and development programs in partnership with the
      academic community.

   B.2. Interconnection Technology

      In order to ensure network-level interoperability of different
      vendor's gateways within the NSFNET context, we specify that a
      gateway must at a minimum support Ethernet connections and serial
      line protocol connections.

      Currently the most important common interconnection technology
      between Internet systems of different vendors is Ethernet. Among
      the reasons for this are the following:

         1. Ethernet specifications are well-understood and mature.

         2. Ethernet technology is in almost all aspects vendor
             independent.

         3. Ethernet-compatible systems are common and becoming more
             so.

These advantages combined favor the use of Ethernet technology as
the common point of demarcation between NSF network systems
supplied by different vendors, regardless of technology. It is a
requirement of NSF gateways that, regardless of the possibly
proprietary switching technology used to implement a given
vendor-supplied network, its gateways must support an Ethernet
attachment to gateways of other vendors.

It is expected that future NSF gateway requirements will specify
other interconnection technologies. The most likely candidates
are those based on X.25 or IEEE 802, but other technologies
including broadband cable, optical fiber, or other media may also
be considered.

B.3. Routing Interoperability

The Internet does not currently have an "open IGP" standard, i.e.,
a common IGP which would allow gateways from different vendors to
form a single Autonomous System. Several approaches to routing
interoperability are currently in use among vendors and the NSF
networking community.

* Proprietary IGP

   At least one gateway vendor has implemented a proprietary IGP
   and uses EGP to interface to the rest of the Internet.

* RIP

   Although RIP is undocumented and various implementations of it
   differ in subtle ways, it has been used successfully for
   interoperation among multiple vendors as an IGP.

* Gateway Daemon

   The NSF networking community has built a "gateway daemon"
   program which can mediate among multiple routing protocols to
   create a mixed-IGP Autonomous System. In particular, the
   prototype gateway daemon executes on a 4.3BSD machine acting as
   a gateway and exchanges routing information with other
   gateways, speaking both RIP and Hello protocols; in addition,
   it supports EGP to other Autonomous Systems.

   B.4. Multi-Protocol Gateways

      The present NSF gateway requirements specify only the Internet
      protocol IP. However, in a few years the Internet will begin a
      gradual transition to the functionally-equivalent subset of the
      ISO protocols [17]. In particular, an increasing percentage of
      the traffic will use the ISO Connectionless Mode Network Service
      (CLNS, but commonly called "ISO IP") [33] in place of IP. It is
      expected that the ISO suite will eventually become the dominant
      one; however, it is also expected that requirements to support
      Internet IP will continue, perhaps indefinitely.

      To support the transition to ISO protocols and the coexistence
      stage, it is highly desirable that a gateway design provide for
      future extensions to support more than one protocol simultaneous,
      and in particular both IP and CLNS [18].

      Present NSF gateway requirements do not include protocols above
      the network layer, such as TCP, unless necessary for network
      monitoring or control. Vendors should recognize that future
      requirements to interwork between Internet and ISO applications,
      for example, may result in an opportunity to market gateways
      supporting multiple protocols at all levels up through the
      application level [16]. It is expected that the network-level NSF
      gateway requirements summarized in this document will be
      incorporated in the requirements document for these
      application-level gateways.

      Internet gateways function as intermediate systems (IS) with
      respect to the ISO connectionless network model and incorporate
      defined packet formats, routing algorithms and related procedures
      [33, 34]. The ISO ES-IS [37] provides the functions of ARP and
      ICMP Redirect.

   B.5. Access Control and Accounting

      There are no requirements for NSF gateways at this time to
      incorporate specific access-control and accounting mechanisms in
      the design; however, these important issues are currently under
      study and will be incorporated into a subsequent edition of this
      document. Vendors are encouraged to plan for the introduction of
      these mechanisms into their products. While at this time no
      definitive common model for access control and accounting has
      emerged, it is possible to outline some general features such a
      model is likely to have, among them the following:

1. The primary access control and accounting mechanisms will
    be in the service hosts themselves, not the gateways,
    packet-switches or workstations.

2. Agents acting on behalf of access control and accounting
    mechanisms may be necessary in the gateways, to collect
    data, enforce password protection, or mitigate resource
    priority and fairness. However, the architecture and
    protocols used by these agents may be a local matter and
    cannot be specified in advance.

3. NSF gateways may be required to incorporate access control
    and accounting mechanisms based on datagram
    source/destination address, as well as other fields in the
    IP header.

4. NSF gateways may be required to enforce policies on access
    to gateway and communication resources. These policies may
    be based upon equity ("fairness") or upon inequity
    ("priority").

Acknowledgments

References and Bibliography

   Many of these references are available from the DDN Network
   Information Center, SRI International, 333 Ravenswood Avenue, Menlo
   Park, California 94025 (telephone: 800-235-3155).

   [1]   Postel, J., "Internet Protocol", RFC-791, USC Information
           Sciences Institute, September 1981.

   [2]   Postel, J., "Internet Control Message Protocol", RFC-792, USC
           Information Sciences Institute, September 1981.

   [3]   BBN, "Interface Message Processor - Specifications for the
           Interconnection of a Host and an IMP", Report 1822, Bolt
           Beranek and Newman, December 1981.

   [4]   Plummer, D., "An Ethernet Address Resolution Protocol",
           RFC-826, Symbolics, September 1982.

   [5]   DOD, "Military Standard Internet Protocol", Military Standard
           MIL-STD-1777, United States Department of Defense, August 1983.

   [6]   BBN, "Defense Data Network X.25 Host Interface Specification",
           Report 5476, Bolt Beranek and Newman, December 1983.

   [7]   Hinden, R., "A Host Monitoring Protocol", RFC-869, BBN
           Communications, December 1983.

   [8]   Korb, J.T., "A Standard for the Transmission of IP Datagrams
           over Public Data Networks", RFC-877, Purdue University,
           September 1983.

   [9]   Nagle, J., "Congestion Control in IP/TCP Internetworks",
           RFC-896, Ford Aerospace, January 1984.

   [10]   Hornig, C., "A Standard for the Transmission of IP Datagrams
           over Ethernet Networks", RFC-894, Symbolics, April 1984.

   [11]   Mills, D.L., "Exterior Gateway Formal Specification", RFC-904,
           M/A-COM Linkabit, April 1984.

   [12]   Xerox, "Xerox Synchronous Point-to-Point Protocol", Xerox
           System Integration Standard 158412, December 1984.

   [13]   Kirton, P., "EGP Gateway under Berkeley UNIX 4.2", RFC-911, USC
           Information Sciences Institute, August 1984.

[14] Postel, J., "Multi-LAN Address Resolution", RFC-925, USC
      Information Sciences Institute, October 1984.

[15] Finlayson, R., T. Mann, J. Mogul, and M. Theimer, "A Reverse
      Address Resolution Protocol", RFC-904, Stanford University,
      June 1984.

[16] NRC, "Transport Protocols for Department of Defense Data
      Networks", RFC-942, National Research Council, March 1985.

[17] Postel, J., "DOD Statement on NRC Report", RFC-945, USC
      Information Sciences Institute, April 1985.

[18] ISO, "Addendum to the Network Service Definition Covering
      Network Layer Addressing", RFC-941, International Standards
      Organization, April 1985.

[19] Leiner, B., J. Postel, R. Cole and D. Mills, "The DARPA
      Internet Protocol Suite", Proceedings INFOCOM 85, IEEE,
      Washington DC, March 1985. Also in: IEEE Communications
      Magazine, March 1985. Also available as ISI-RS-85-153.

[20] Romkey, J., "PC/IP Programmer's Manual", MIT Laboratory for
      Computer Science, pp. 57-59, April 1986.

[21] Mogul, J., and J. Postel, "Internet Standard Subnetting
      Procedure", RFC-950, Stanford University, August 1985.

[22] Reynolds, J., and J. Postel, "Official Internet Protocols",
      RFC-1011, USC Information Sciences Institute, May 1987.

[23] Reynolds, J., and J. Postel, "Assigned Numbers", RFC-1010, USC
      Information Sciences Institute, May 1987.

[24] Nagle, J., "On Packet Switches with Infinite Storage", RFC-970,
      Ford Aerospace, December 1985.

[25] SRI, "DDN Protocol Handbook", NIC-50004, NIC-50005, NIC-50006,
      (three volumes), SRI International, December 1985.

[26] SRI, "ARPANET Information Brochure", NIC-50003, SRI
      International, December 1985.

[27] Mills, D.L., "Autonomous Confederations", RFC-975, M/A-COM
      Linkabit, February 1986.

[28] Jacobsen, O., and J. Postel, "Protocol Document Order
      Information", RFC-980, SRI International, March 1986.

   [29]   Malis, A.G., "PSN End-to-End Functional Specification",
          RFC-979, BBN Communications, March 1986.

   [30]   Postel, J, "Internetwork Applications using the DARPA Protocol
          Suite", Proceedings INFOCOM 85, IEEE, Washington DC,
          March 1985. Also available as ISI-RS-85-151.

   [31]   Postel, J, C. Sunshine, and D. Cohen, "The ARPA Internet
          Protocol", Computer Networks, Vol. 5, No. 4, July 1981.

   [32]   Cerf, V., and R. Kahn, "A Protocol for Packet Network
          Intercommunication", IEEE Transactions on Communication,
          May 1974.

   [33]   ISO, "Protocol for Providing the Connectionless-mode Network
          Service", RFC-994, DIS-8473, International Standards
          Organization, March 1986.

   [34]   ANSI, "Draft Network Layer Routing Architecture", ANSI X3S3.3,
          86-215R, April 1987.

   [35]   Rosen, E., "Exterior Gateway Protocol (EGP)", RFC-827, Bolt
          Beranek and Newman, October 1982.

   [36]   Sidhu, D., "Some Problems with the Specification of the
          Military Standard Internet Protocol", RFC-963, Iowa State
          University, November 1985.

   [37]   ISO, "End System to Intermediate System Routing Exchange
          Protocol for use in conjunction with ISO 8473", RFC-995,
          April 1986.

   [38]   Postel, J., "Address Mappings", RFC-796, USC/Information
          Sciences Institute, September 1981.

   [39]   Mills, D., "DCN Local Network Protocols", RFC-891, M/A-COM
          Linkabit, December 1983.

   [40]   McQuillan, J. M., I. Richer, and E. C. Rosen, "The New Routing
          Algorithm for the ARPANET", IEEE Transactions on
          Communications, May 1980.

   [41]   Hinden, R., and A. Sheltzer, "The DARPA Internet Gateway",
          RFC-823, Bolt Beranek and Newman, September 1982.

   [42]   Farber, D., G. Delp, and T. Conte, "A Thinwire Protocol for
          Connecting Personal Computers to the Internet", RFC-914,
          University of Delaware, September 1984.

[43]   Mills, D., "Statistics Server", RFC-996, University Of
        Delaware, February 1987.

[44]   Postel, J. and K. Harrenstien, "Time Protocol", RFC-868,
        May 1983.

[45]   Mills, D., "Network Time Protocol (NTP)", RFC-958, M/A-Com
        Linkabit, September 1985.

[46]   Seamonson, L., and E. Rosen, "Stub Exterior Gateway Protocol",
        RFC-888, Bolt Beranek And Newman, January 1984.

[47]   Deering, S., and D. Cheriton, "Host Groups: A Multicast
        Extension to the Internet Protocol", RFC-966, Stanford
        University, December 1985.

[48]   Deering, S., "Host Extensions for IP Multicasting", RFC-988,
        Stanford University, July 1986.

[49]   Mogul, J., "Broadcasting Internet Datagrams", RFC-919, Stanford
        University, October 1984.

[50]   Mogul, J., "Broadcasting Internet Datagrams in the Presence of
        Subnets", RFC-922, Stanford University, October 1984.

[51]   Rosen, E., "Exterior Gateway Protocol", RFC-827, Bolt Beranek
        and Newman, October 1982.

[52]   Rose, M., "Low Tech Connection into the ARPA Internet: The Raw
        Packet Split Gateway", Technical Report 216, Department of
        Information and Computer Science, University of California,
        Irvine, February 1984.

[53]   Rosen, E., "Issues in Buffer Management", IEN-182, Bolt Beranek
        and Newman, May 1981.

[54]   Rosen, E., "Logical Addressing", IEN-183, Bolt Beranek and
        Newman, May 1981.

[55]   Rosen, E., "Issues in Internetting - Part 1: Modelling the
        Internet", IEN-184, Bolt Beranek and Newman, May 1981.

[56]   Rosen, E., "Issues in Internetting - Part 2: Accessing the
        Internet", IEN-187, Bolt Beranek and Newman, June 1981.

[57]   Rosen, E., "Issues in Internetting - Part 3: Addressing",
        IEN-188, Bolt Beranek and Newman, June 1981.

[58] Rosen, E., "Issues in Internetting - Part 4: Routing", IEN-189,
      Bolt Beranek and Newman, June 1981.

[59] Sunshine, C., "Comments on Rosen's Memos", IEN-191, USC
      Information Sciences Institute, July 1981.

[60] NTAG, "Requirements for Internet Gateways -- Draft", RFC-985,
      Network Technical Advisory Group, National Science Foundation,
      May 1986.

[61] Khanna, A., and Malis, A., "The ARPANET AHIP-E Host Access
      Protocol (Enhanced AHIP)", RFC-1005, BBN Communications,
      May 1987

[62] Nagle, J., "Congestion Control in IP/TCP Internetworks", ACM
      Computer Communications Review, Vol.14, no.4, October 1984.