Network Working Group
Request for Comments: 1454

T. Dixon
RARE

May 1993

Comparison of Proposals for Next Version of IP

Status of this Memo

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

Abstract

This is a slightly edited reprint of RARE Technical Report (RTC(93)004).

The following is a brief summary of the characteristics of the three main proposals for replacing the current Internet Protocol. It is not intended to be exhaustive or definitive (a brief bibliography at the end points to sources of more information), but to serve as input to the European discussions on these proposals, to be co-ordinated by RARE and RIPE. It should be recognised that the proposals are themselves "moving targets", and in so far as this paper is accurate at all, it reflects the position at the 25th IETF meeting in Washington, DC. Comments from Ross Callon and Paul Tsuchiya on the original draft have been incorporated. Note that for a time the term "IPv7" was use to mean the eventual next version of IP, but that the same term was closely associated with a particilar proposal, so the term "IPng" is now used to identify the eventual next generation of IP.

The paper begins with a "generic" discussion of the mechanisms for solving problems and achieving particular goals, before discussing the proposals invidually.

1. WHY IS THE CURRENT IP INADEQUATE?

The problem has been investigated and formulated by the ROAD group, but briefly reduces to the following:

- Exhaustion of IP Class B Address Space.
- Exhaustion of IP Address Space in General.
- Non-hierarchical nature of address allocation leading to flat routing space.

Although the IESG requirements for a new Internet Protocol go further than simply routing and addressing issues, it is these issues that make extension of the current protocol an impractical option. Consequently, most of the discussion and development of the various proposed protocols has concentrated on these specific problems.

Near term remedies for these problems include the CIDR proposals (which permit the aggregation of Class C networks for routing purposes) and assignment policies which will allocate Class C network numbers in a fashion which CIDR can take advantage of. Routing protocols supporting CIDR are OSPF and BGP4. None of these are pre-requisites for the new IP (IPng), but are necessary to prolong the life of the current Internet long enough to work on longer-term solutions. Ross Callon points out that there are other options for prolonging the life of IP and that some ideas have been distributed on the TUBA list.

Longer term proposals are being sought which ultimately allow for further growth of the Internet. The timescale for considering these proposals is as follows:

- Dec 15 Issue selection criteria as RFC.

- Feb 12 Two interoperable implementations available.

- Feb 26 Second draft of proposal documents available.

The (ambitious) target is for a decision to be made at the 26th IETF (Columbus, Ohio in March 1993) on which proposals to pursue.

The current likely candidates for selection are:

- PIP (’P’ Internet Protocol - an entirely new protocol).

- TUBA (TCP/UDP with Big Addresses - uses ISO CLNP).

- SIP (Simple IP - IP with larger addresses and fewer options).

There is a further proposal from Robert Ullman of which I don’t claim to have much knowledge. Associated with each of the candidates are transition plans, but these are largely independent of the protocol itself and contain elements which could be adopted separately, even with IP v4, to further extend the life of current implementations and systems.

2. WHAT THE PROPOSALS HAVE IN COMMON

2.1 Larger Addresses

All the proposals (of course) make provision for larger address fields which not only increase the number of addressable systems, but also permit the hierarchical allocation of addresses to facilitate route aggregation.

2.2 Philosophy

The proposals also originate from a "routing implementation" view of the world - that is to say they focus on the internals of routing within the network and do not primarily look at the network service seen by the end-user, or by applications. This is perhaps inevitable, especially given the tight time constraints for producing interoperable implementations. However, the (few) representatives of real users at the 25th IETF, the people whose support is ultimately necessary to deploy new host implementations, were distinctly unhappy.

There is an inbuilt assumption in the proposals that IPng is intended to be a universal protocol: that is, that the same network-layer protocol will be used between hosts on the same LAN, between hosts and routers, between routers in the same domain, and between routers in different domains. There are some advantages in defining separate "access" and "long-haul" protocols, and this is not precluded by the requirements. However, despite the few opportunities for major change of this sort within the Internet, the need for speed of development and low risk have led to the proposals being incremental, rather than radical, changes to well-proven existing technology.

There is a further unstated assumption that the architecture is targeted at the singly-connected host. It is currently difficult to design IPv4 networks which permit hosts with more than one interface to benefit from increased bandwidth and reliability compared with singly-connected hosts (a consequence of the address belonging to the interface and not the host). It would be preferable if topological constraints such as these were documented. It has been asserted that this is not necessarily a constraint of either the PIP or TUBA proposals, but I believe it is an issue that has not emerged so far amongst the comparative criteria.

2.3 Source Routing

The existing IPv4 has provision for source-specified routes, though this is little used [would someone like to contradict me here?], partly because it requires knowledge of the internal structure of the network down to the router level. Source routes are usually required by users when there are policy requirements which make it preferable or imperative that traffic between a source and destination should pass through particular administrative domains. Source routes can also be used by routers within administrative domains to route via particular logical topologies. Source-specified routing requires a number of distinct components:

a. The specification by the source of the policy by which the route should be selected.

b. The selection of a route appropriate to the policy.

c. Marking traffic with the identified route.

d. Routing marked traffic accordingly.

These steps are not wholly independent. The way in which routes are identified in step (c) may constrain the kinds of route which can be selected in previous steps. The destination, inevitably, participates in the specification of source routes either by advertising the policies it is prepared to accept or, conceivably, by a negotiation process.

All of the proposals mark source routes by adding a chain of (perhaps partially-specified) intermediate addresses to each packet. None specifies the process by which a host might acquire the information needed to specify these intermediate addresses [not entirely unreasonably at this stage, but further information is expected]. The negative consequences of these decisions are:

- Packet headers can become quite long, depending on the number of intermediate addresses that must be specified (although there are mechanisms which are currently specified or which can be imagined to specify only the significant portions of intermediate addresses).

- The source route may have to be re-specified periodically if particular intermediate addresses are no longer reachable.

The positive consequences are:

- Inter-domain routers do not have to understand policies, they simply have to mechanically follow the source route.

- Routers do not have to store context identifying routes, since the information is specified in each packet header.

- Route servers can be located anywhere in the network, provided the hosts know how to find them.

2.4 Encapsulation

Encapsulation is the ability to enclose a network-layer packet within another one so that the actual packet can be directed via a path it would not otherwise take to a router that can remove the outermost packet and direct the resultant packet to its destination. Encapsulation requires:

a. An indication in the packet that it contains another packet.

b. A function in routers which, on receiving such a packet, removes the encapsulation and re-enters the forwarding process.

All the proposals support encapsulation. Note that it is possible to achieve the effect of source routing by suitable encapsulation by the source.

2.5 Multicast

The specification of addresses to permit multicast with various scopes can be accomodated by all the proposals. Internet-wide multicast is, of course, for further study!

2.6 Fragmentation

All the proposals support the fragmentation of packets by intermediate routers, though there has been some recent discussion of removing this mechanism from some of the proposals and requiring the use of an MTU-discovery process to avoid the need for fragmentation. Such a decision would effectively preclude the use of transport protocols which use message-count sequence numbering (such as OSI Transport) over the network, as only protocols with byte-count acknowledgement (such as TCP) can deal with MTU reductions during the lifetime of a connection. OSI Transport may not be particularly relevant to the IP community (though it may be of relevance to commercial suppliers providing multiprotocol services), however the consequences for the types of services which may be supported over IPng should be noted.

2.7 The End of Lifetime as We Know It

The old IPv4 "Time to Live" field has been recast in every case as a simple hop count, largely on grounds of implementation convenience. Although the old TTL was largely implemented in this fashion anyway, it did serve an architectural purpose in putting an upper bound on the lifetime of a packet in the network. If this field is recast as a hop-count, there must be some other specification of the maximum lifetime of a packet in the network so that a source host can ensure that network-layer fragment ids and transport-layer sequence numbers are never in danger of re-use whilst there is a danger of confusion. There are, in fact, three separate issues here:

1. Terminating routing loops (solved by hop count).

2. Bounding lifetime of network-layer packets (a necessity, unspecified so far) to support assumptions by the transport layer.

3. Permitting the source to place further restrictions on packet lifetime (for example so that "old" real-time traffic can be discarded in favour of new traffic in the case of congestion (an optional feature, unspecified so far).

3. WHAT THE PROPOSALS ONLY HINT AT

3.1 Resource Reservation

Increasingly, applications require a certain bandwidth or transit delay if they are to be at all useful (for example, real-time video and audio transport). Such applications need procedures to indicate their requirements to the network and to have the required resources reserved. This process is in some ways analogous to the selection of a source route:

a. The specification by the source of its requirements.

b. The confirmation that the requirements can be met.

c. Marking traffic with the requirement.

d. Routing marked traffic accordingly.

Traffic which is routed according to the same set of resource requirements is sometimes called a "flow". The identification of flows requires a setup process, and it is tempting to suppose that the same process might also be used to set up source routes, however, there are a number of differences:

- All the routers on a path must participate in resource reservation and agree to it.

- Consequently, it is relatively straightforward to maintain context in each router and the identification for flows can be short.

- The network can choose to reroute on failure.

By various means, each proposal could carry flow-identification, though this is very much "for future study" at present. No setup mechansisms are defined. The process for actually reserving the resources is a higher-order problem. The interaction between source-routing and resource reservation needs further investigation: although the two are distinct and have different implementation constraints, the consequence of having two different mechanisms could be that it becomes difficult to select routes which meet both policy and performance goals.

3.2 Address-Assignment Policies

In IPv4, addresses were bound to systems on a long-term basis and in many cases could be used interchangeably with DNS names. It is tacitly accepted that the association of an address with a particular system may be more volatile in IPng. Indeed, one of the proposals, PIP, makes a distinction between the identification of a system (a fixed quantity) and its address, and permits the binding to be altered on the fly. None of the proposals defines bounds for the lifetime of addresses, and the manner in which addresses are assigned is not necessarily bound to a particular proposal. For example, within the larger address space to be provided by IPng, there is a choice to be made of assigning the "higher order" part of the hierarchical address in a geographically-related fashion or by reference to service provider. Geographically-based addresses can be constant and easy to assign, but represent a renewed danger of degeneration to "flat" addresses within the region of assignment, unless certain topological restrictions are assumed. Provider-based address assignment results in a change of address (if providers are changed) or multiple addresses (if multiple providers are used). Mobile hosts (depending on the underlying technology) can present problems in both geographic and provider-based schemes.

Without firm proposals for address-assignment schemes and the consequences for likely address lifetimes, it is impossible to assume that the existing DNS model by which name-to-address bindings can be discovered remains valid.

Note that there is an interaction between the mechanism for assignment of addresses and way in which automatic configuration may be deployed.

3.3 Automatic Configuration

Amongst the biggest (user) bugbears of current IP services is the administrative effort of maintaining basic configuration information, such as assigning names and addresses to hosts, ensuring these are refelected in the DNS, and keeping this information correct. Part of this results from poor implementation (or the blind belief that vi and awk are network management tools). However, a lot of the problems could be alleviated by making this process more automatic. Some of the possibilities (some mutually-exlusive) are:

- Assigning host addresses from some (relative) invariant, such as a LAN address.

- Defining a protocol for dynamic assignment of addresses within a subnetwork.

- Defining "generic addresses" by which hosts can without preconfiguration reach necessary local servers (DNS, route servers, etc.).

- Have hosts determine their name by DNS lookup.

- Have hosts update their name/address bindings when their configuration changes.

Whilst a number of the proposals make mention of some of these possibilities, the choice of appropriate solutions depends to some extent on address-assignment policies. Also, dynamic configuration results in some difficult philosopical and practical issues (what exactly is the role of an address?, In what sense is a host "the same host" when its address changes?, How do you handle dynamic changes to DNS mappings and how do you authenticate them?).

The groups involved in the proposals would, I think, see most of these questions outside their scope. It would seem to be a failure in the process of defining and selecting candidates for IPng that "systemness" issues like these will probably not be much discussed. This is recognised by the participants, and it is likely that, even when a decision is made, some of these ideas will be revisited by a wider audience.

It is, however, unlikely that IP will make an impact on proprietary networking systems for the non-technical environment (e.g., Netware, Appletalk), without automatic configuration being taken seriously either in the architecture, or by suppliers. I believe that there are ideas on people’s heads of how to address these issues - they simply have not made it onto paper yet.

3.4 Application Interface/Application Protocol Changes

A number of common application protocols (FTP, RPC, etc.) have been identified which specifically transfer 32-bit IPv4 addresses, and there are doubtless others, both standard and proprietary. There are also many applications which treat IPv4 addresses as simple 32-bit integers. Even applications which use BSD sockets and try to handle addresses opaquely will not understand how to parse or print longer addresses (even if the socket structure is big enough to accommodate them).

Each proposal, therefore, needs to specify mechanisms to permit existing applications and interfaces to operate in the new environment whilst conversion takes place. It would be useful also, to have (one) specification of a reference programming interface for (TCP and) IPng (which would also operate on IPv4), to allow developers to begin changing applications now. All the proposals specify transition mechansisms from which existing application-compatibility can be inferred. There is no sign yet of a new interface specification independent of chosen protocol.

3.5 DNS Changes

It is obvious that there has to be a name to address mapping service which supports the new, longer, addresses. All the proposals assume that this service will be provided by DNS, with some suitably-defined new resource record. There is some discussion ongoing about the appropriateness of returning this information along with "A" record information in response to certain enquiries, and which information should be requested first. There is a potential tradeoff between the number of queries needed to establish the correct address to use and the potential for breaking existing implementations by returning information that they do not expect.

There has been heat, but not light, generated by discussion of the use of DNS for auto-configuration and the scaling (or otherwise) of reverse translations for certain addressing schemes.

4. WHAT THE PROPOSALS DON’T REALLY MENTION

4.1 Congestion Avoidance

IPv4 offers "Source Quench" control messages which may be used by routers to indicate to a source that it is congested and has or may shortly drop packets. TUBA/PIP have a "congestion encountered" bit which provides similar information to the destination. None of these specifications offers detailed instructions on how to use these facilities. However, there has been a substantial body of analysis over recent years that suggests that such facilities can be used (by providing information to the transport protocol) not only to signal congestion, but also to minimise delay through the network layer. Each proposal can offer some form of congestion signalling, but none specifies a mechanism for its use (or an analysis of whether the mechanism is in fact useful).

As a user of a network service which currently has a discard rate of around 30% and a round-trip-time of up to 2 seconds for a distance of only 500 miles I would be most interested in some proposals for a more graceful degradation of the network service under excess load.

4.2 Mobile Hosts

A characteristic of mobile hosts is that they (relatively) rapidly move their physical location and point of attachment to the network topology. This obviously has signficance for addressing (whether geographical or topological) and routing. There seems to be an understanding of the problem, but so far no detailed specification of a solution.

4.3 Accounting

The IESG selection criteria require only that proposals do not have the effect of preventing the collection of information that may be of interest for audit or billing purposes. Consequently, none of the proposals consider potential accounting mechanisms.

4.4 Security

"Network Layer Security Issues are For Further Study". Or secret.

However, it would be useful to have it demonstrated that each candidate could be extended to provide a level of security, for example against address-spoofing. This will be particularly important if resource-allocation features will permit certain hosts to claim large chunks of available bandwidth for specialised applications.

Note that providing some level of security implies manual configuration of security information within the network and must be considered in relationship to auto-configuration goals.

5. WHAT MAKES THE PROPOSALS DIFFERENT?

Each proposal is about as different to the others as it is to IPv4 - that is the differences are small in principle, but may have significant effects (extending the size of addresses is only a small difference in principle!). The main distinct characteristics are:

PIP:

PIP has an innovative header format that facilitates hierarchical, policy and virtual-circuit routing. It also has "opaque" fields in the header whose semantics can be defined differently in different administrative domains and whose use and translation can be negotiated across domain boundaries. No control protocol is yet specified.

SIP:

SIP offers a "minimalist" approach - removing all little-used fields from the IPv4 header and extending the size of addresses to (only) 64 bits. The control protocol is based on modifications to ICMP. This proposal has the advantages of processing efficiency and familiarity.

TUBA:

TUBA is based on CLNP (ISO 8473) and the ES-IS (ISO 9542) control protocol. TUBA provides for the operation of TCP transport and UDP over a CLNP network. The main arguments in favour of TUBA are that routers already exist which can handle the network-layer protocol, that the extensible addresses offer a wide margin of "future-proofing" and that there is an opportunity for convergence of standards and products.

5.1 PIP

PIP packet headers contain a set of instructions to the router’s forwarding processor to perform certain actions on the packet. In traditional protocols, the contents of certain fields imply certain actions; PIP gives the source the flexibility to write small "programs" which direct the routing of packets through the network.

PIP addresses have an effectively unlimited length: each level in the topological hierarchy of the network contributes part of the address and addresses change as the network topology changes. In a completely hierarchical network topology, the amount of routing information required at each level could be very small. However, in practice, levels of hierarchy will be determined more by commercial and practical factors than by the constraints of any particular routing protocol. A greater advantage is that higher-order parts of the address may be omitted in local exchanges and that lower-order parts may be omitted in source routes, reducing the amount of topological information that host systems are required to know.

There is an assumption that PIP addresses are liable to change, so a further quantity, the PIP ID, is assigned to systems for the purposes of identification. It isn’t clear that this quantity has any purpose which could not equally be served by a DNS name [it is more compact, but equally it does not need to be carried in every packet and requires an additional lookup]. However, the problem does arise of how two potentially-communicating host systems find the correct addresses to use.

The most complex part of PIP is that the meaning of some of the header fields is determined by mutual agreement within a particular domain. The semantics of specific processing facilities (for example, queuing priority) are registered globally, but the actual use and encoding of requests for these facilities in the packet header can be different in different domains. Border routers between two domains which use different encodings must map from one encoding to another. Since routers may not only be adjacent physically to other domains, but also via "tunnels", the number of different encoding rules a router may need to understand is potentially quite large. Although there is a saving in header space by using such a scheme as opposed to the more familiar "options", the cost in the complexity of negotiating the use and encoding of these facilities, together with re-coding the packets at each domain border, is a subject of some concern. Although it may be possible for hosts to "precompile" the encoding rules for their local domain, there are many potential implementaion difficulties.

Although PIP offers the most flexibility of the three proposals, more work needs to be done on "likely use" scenarios which make the potential advantages and disadvantages more concrete.

5.2 SIP

SIP is simply IP with larger addresses and fewer options. Its main advantage is that it is even simpler that IPv4 to process. Its main disadvantages are:

- It is far from clear that, if 32 bits of address are insufficient, 64 will be enough for the forseeable future;

- although there are a few "reserved" bits in the header, the extension of SIP to support new features is not obvious.

There’s really very little else to say!

5.3 TUBA

The characteristics of ISO CLNS are reasonably well known: the protocol bears a strong cultural resemblance to IPv4, though with 20-byte network-layer addressing. Apart from a spurious "Not Invented Here" prejudice, the main argument againt TUBA is that it is rather too like IPv4, offering nothing other than larger, more flexible, addresses. There is proof-by-example that routers are capable of handling the (very) long addresses efficiently, rather less that the longer headers do not adversely impact network bandwidth.

There are a number of objections to the proposed control protocol (ISO 9542):

- My early experience is that the process by which routers discover hosts is inefficient and resource consuming for routers - and requires quite fine timer resolution on hosts - if large LANs are to be accomodated reasonably. Proponents of TUBA suggest that recent experience suggests that ARP is no better, but I think this issue needs examination.

- The "redirect" mechanism is based on (effectively) LAN addresses and not network addresses, meaning that local routers can only "hand-off" complex routing decisions to other routers on the same LAN. Equally, redirection schemes (such as that of IPv4) which redirect to network addresses can result in unnecessary extra hops. Analysis of which solution is better is rather dependent on the scenarios which are constructed.

To be fair, however, the part of the protocol which provides for router-discovery provides a mechanism, absent from other proposals, by which hosts can locate nearby gateways and potentially automatically configure their addresses.

6. Transition Plans

It should be obvious that a transition which permits "old" hosts to talk to "new" hosts requires:

Either:

(a) That IPng hosts can also use IPv4 or
(b) There is translation by an intermediate system

and either:

(c) The infrastructure between systems is capable of carrying both IPng and IPv4 or (d) Tunneling or translation is used to carry one protocol within another in parts of the network

The transition plans espoused by the various proposals are simply different combinations of the above. Experience would tend to show that all these things will in fact happen, regardless of which protocol is chosen.

One problem of the tunneling/translation process is that there is additional information (the extra address parts) which must be carried across IPv4 tunnels in the network. This can either be carried by adding an extra "header" to the data before encapsulation in the IPv4 packet, or by encoding the information as new IPv4 option types. In the former case, it may be difficult to map error messages correctly, since the original packet is truncated before return; in the latter case there is a danger of the packet being discarded (IPv4 options are not self-describing and new ones may not pass through IPv4 routers). There is thus the possibility of having to introduce a "new" version of IPv4 in order to support IPng tunneling.

The alternative (in which IPng hosts have two stacks and the infrastructure may or may not support IPng or IPv4) of course requires a mechanism for resolving which protocols to try.

7. Random Comments

This is the first fundamental change in the Internet protocols that has occurred since the Internet was manageable as an entity and its development was tied to US government contracts. It was perhaps inevitable that the IETF/IESG/IAB structure would not have evolved to manage a change of this magnitude and it is to be hoped that the new structures that are proposed will be more successful in promoting a (useful) consensus. It is interesting to see that many of the perceived problems of the OSI process (slow progress, factional infighting over trivia, convergence on the lowest-common denominator solution, lack of consideration for the end-user) are in danger of attaching themselves to IPng and it will be interesting to see to what extent these difficulties are an inevitable consequence of wide representation and participation in network design.

It could be regarded either as a sign of success or failure of the competitive process for the selection of IPng that the three main proposals have few really significant differences. In this respect, the result of the selection process is not of particular significance, but the process itself is perhaps necessary to repair the social and technical cohesion of the Internet Engineering process.

8. Further Information

The main discussion lists for the proposals listed are:

TUBA:	tuba@lanl.gov
PIP:	pip@thumper.bellcore.com
SIP:	sip@caldera.usc.edu
General:	big-internet@munnari.oz.au

(Requests to: <list name>-request@<host>)

Internet-Drafts and RFCs for the various proposals can be found in the usual places.

Security Considerations

Security issues are not discussed in this memo.

Author's Address

Tim Dixon
RARE Secretariat
Singel 466-468
NL-1017AW Amsterdam
(Netherlands)

Phone: +31 20 639 1131 or + 44 91 232 0936
EMail: dixon@rare.nl or Tim.Dixon@newcastle.ac.uk