TITLE: A MODEL FOR SECURE CALL-LIKE SESSIONS WITHIN THE INTERNET

<draft-davies-broker-model-00.txt>

- September 1996 -

AUTHORS: P.J.Williams) GPT Limited, UK.
Ian Davies)

Status of this document

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as ``work in progress.''

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt'' listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

ABSTRACT

The Internet is increasingly being used for commercial and industrial purposes which, apart from causing an explosion in traffic, will require that the future network provides a highly efficient "on-demand" service. The provision of an efficient on-demand service will require Internet "sessions" to use virtual-message-paths that must be opened and closed in a manner similar to the establishment of calls in a telephone network; enabling the network to refuse traffic which exceeds the available capacity and ensuring that established sessions will not be violated. It is felt that the RSVP mechanism will not fully meet these requirements for the commercial services described below, and this paper describes an alternative.

As the Internet becomes the ultimate source of information, the "Global Net", users will need intermediary "Broker" services to clarify their requirement, select the appropriate service and arrange delivery of the information. It is proposed that Broker-type "special-service" sessions should be

established in reverse. The Broker or Receptionist will be given identifiers enabling the virtual-message-path to the user to be picked-up at the user's Internet Router. As the user's requirements are recognised, these identifiers can be passed from Broker to Broker and from Server to Server enabling other Brokers or Servers to join the session, or enabling the user to be transferred from one to the other until service delivery is complete.

The proposed method of service delivery is very user-friendly. From the user's point of view, every session appears to be a simple session with a single Internet host. Whatever starting point is chosen (even the wrong one), the transfer activity will eventually deliver the required information. The method is also commercially secure. The user does not participate in the transfer activity and cannot short-circuit the Broker on future occasions. Servers deliver information directly to the user - whose identity is verified, thereby easing if not completely eliminating the "hacking" problem.

TABLE OF CONTENTS.

1. INTRODUCTION 0004

2. REQUIREMENTS OF THE FUTURE INTERNET. 0004
  2.1. The "connection-oriented" requirement. 0004
  2.2. The simple-session requirement 0005
  2.3. The "special-service" requirement. 0005

3. OUTLINE OF SPECIAL-SERVICE IMPLEMENTATION. 0006

4. INTERNET NAMES AND ADDRESSES 0006
  4.1. Internet names 0006
  4.2. Internet addresses 0007
    4.2.1. User addresses 0007
    4.2.2. Service-delivery addresses 0007
    4.2.3. Special-service networks 0008

5. TRANSPORT LAYER PROTOCOL MIGRATION STRATEGY. . . .0008

6. CREATING A VIRTUAL-MESSAGE-PATH. 0008
  6.1. The host/Router link 0008
    6.1.1. The host/Router link header. 0009
  6.2. The OPEN message 0009

6.3. The message-path through the Internet. 0010
6.3.1. VCN allocation 0010
6.3.2. Switching tables 0010
6.3.3. The OPEN_DONE message. 0011
6.3.4. Reserving capacity 0012
6.3.5. Using the message-path 0012
6.3.6. Diverting a message-path 0013
6.3.7. Closing the message-path 0013

7. SPECIAL-SERVICES 0013
7.1. Invoking special-service delivery. 0013
7.1.1. The SERVICE_ACK message. 0014
7.1.2. The SERVICE_REQUEST message. 0014
7.2. Establishing the service delivery path 0015
7.2.1. The OPEN_SERVICE message 0015
7.2.2. The OPEN_DONE messages 0016
7.2.2.1. OPEN-DONE to the Server. 0016
7.2.2.2. OPEN_DONE to the user. 0017
7.2.3. Control message interface. 0017
7.2.4. The REQUEST_DONE message 0018
7.3. Delivering the service 0018
7.4. Service transfer 0018
7.4.1. The TRANSFER_REQUEST message 0018
7.4.2. The OPEN_TRANSFER message. 0018
7.4.3. The OPEN_DONE and CLOSE_REQUEST messages . . .0018
7.4.4. The REQUEST_DONE message 0019
7.5. Subsequent transfers 0019

8. ACCOMMODATING EXISTING EQUIPMENT 0021
8.1. A simple session in the unknown Internet 0021
8.1.1. The OPEN message 0022
8.1.2. The OPEN_OLD message 0022
8.1.3. Conducting the session 0023
8.1.3.1. Forward messages 0023
8.1.3.2. Backward messages. 0023
8.1.3.3. Closing the session. 0024
8.2. A special-service session in the unknown Internet.0024
8.2.1. Invoking the service 0024
8.2.2. SERVICE_REQUEST via old network. 0024
8.2.2.1. UDP/IP header. 0024
8.2.2.2. The ACK_OLD message. 0024
8.2.3. Service delivery via old network 0025
8.2.3.1. The OPEN_SERVICE message 0025
8.2.3.2. The OPEN_OLD message 0025
8.2.4. Conducting the session 0027
8.2.4.1. Forward messages 0027
8.2.4.2. Backward messages. 0028

8.2.4.3. Closing the previous message-path. 0028
8.2.5. Closing the session. 0028

9. MULTI-SESSIONS 0029

10. THE FUTURE INTERNET - CHARGING 0029
10.1. Basic charges. 0029
10.2. Special-service charges. 0029
10.3. Collecting Service Providers charges 0030

11. SECURITY 0030
11.1. User identity verification 0030
11.2. Breeching security 0030

1. INTRODUCTION.

This paper describes a very simple, very user-friendly and commercially secure means of providing access to the endless variety of services and sources of information that is and will be available via the Internet. The proposals are an adaptation of the principles employed in the "Enhanced Intelligent Network" proposed for the telephone network and are equally applicable to ATM, X25 or similar networks.

To gain access to even the most remote and obscure source of information, a user simply opens a session with a Broker or Enquiry type service. As the user's needs are identified, the session will be transferred from Broker to Broker and from Server to Server until the final objective is reached. The user is not required to participate in the transfer activity.

Brokers do not reveal any information to their client users as they transfer sessions to other Brokers or Servers. Having gained access to a service via a Broker, a user cannot regain access to that service without again going via the Broker.

Servers deliver information directly to the user whose identity is verified. The Broker cannot copy the information.

The paper also shows how the proposals can interwork with the existing Internet architecture and protocols.

2. REQUIREMENTS OF THE FUTURE INTERNET.

The Internet is increasingly being used for commercial and industrial purposes which, apart from causing an explosion in traffic, will require that the future network provides a highly efficient and commercially secure "on-demand" service.

The current arrangement, in which users benevolently handle and switch one-another's traffic (and do not quarrel about the performance), cannot continue. Internet Providers must be independent of and impartial to the users. Users must know that their messages are not handled in a competitor's network.

The future Internet Providers will be similar to the telephone network Providers. Users will pay a fixed charge for Internet access and will almost certainly pay an additional usage charge. In some cases, a further charge may be levied by the distant terminal for the actual information delivered. The charging arrangements are reviewed in Section 10.

2.1. The "connection-oriented" requirement.

The provision of an efficient on-demand service will require Internet "sessions" to use virtual-message-paths that must be opened and closed in a manner similar to the establishment of calls in a telephone network; enabling the network to refuse

traffic which exceeds the available capacity and ensuring that established sessions will not be violated.

Charging is another aspect of the future Internet that will require sessions to be opened and closed. Internet Providers will record the traffic at network terminals and at gateways to other networks for charging purposes, but will need to know which terminal originated the session to which the messages belong; - that is "which end is paying?"

2.2. The simple-session requirement.

Perhaps the majority of Internet usage will continue to be where a user establishes a virtual-message-path to another user to conduct a simple one-to-one session.

The present method of gaining access to remote information during a simple session is for the terminating end to open a session with the remote source and relay information as required. This method is known as "chaining".

Alternatively, the terminating end may send to the user the address of the remote information. The user is required to end the current session and open a session with the new address to obtain the information, Returning an Internet address in this manner is known as "referral".

"Chaining" and "referral" will continue to be used in the future but in many cases it will be preferable to use the proposed means of "special-service" delivery.

2.3. The "special-service" requirement.

In many cases, an Internet user may merely know that he wishes to buy software, insurance, or contact a large multi-national business organisation and will need to open an initial session with a broker or receptionist in order to proceed.

Having interrogated the user to identify the service required, the Broker/Receptionist may not need to participate in the ensuing activity but may be unwilling to use "referral" as this would give information to the user that could be used, or misused on future occasions.

e.g. the user could bypass the Broker on future occasions.

the user could pass the information to other users.

the user could amend the information to gain access to other services provided by the Server.

the information may enable privileged access to the service and privileged charges to be applicable.

"Chaining" may also be unacceptable. Apart from the additional handling and resources involved, the Server may be unwilling to deliver its service via a Broker and may require to deal directly with the user.
Service delivery requires that the user can be transferred from Broker to Broker and from Server to Server as service delivery proceeds, without being involved in the transfer activity.

3. OUTLINE OF SPECIAL-SERVICE IMPLEMENTATION.

"Special-service" sessions will be established in reverse.

When a user attempts to open a session with a special-service, the special-service Server will be given identifiers enabling it to establish a virtual-message-path through the Internet to pick-up the virtual-message-path from the user at the user's Internet Router (a message switching node of the Internet).

As the session proceeds, the identifiers may be passed from Server to Server enabling the user to be transferred from one Server to another, or enabling other Servers to be brought into the session.

This basic concept of special-service delivery is very simple, but because the present Internet is not connection-oriented, a detailed description of its implementation requires that the creation of a virtual-message-path for a simple session is described and that the method of interworking with existing equipment is included.

Consequently, the implementation is detailed in three parts.

Part 1. Creating a virtual-message-path. (Section 6.)

Part 2. Special-service delivery. (Section 7.)

Part 3. Accommodating existing equipment. (Section 8.)

4. INTERNET NAMES AND ADDRESSES

4.1. Internet names.

To send a message or open a session in the Internet, a user provides the Internet name of the distant end. The initial task in any activity is to consult a cache or Domain Name Server to obtain the Internet address. As a result the choice of an Internet name is, and will remain, rather arbitrary.

4.2. Internet addresses.

4.2.1. User addresses. (See Figure 1)

For the majority of Internet users, the more significant bits of the user's address (netID) will identify the user's Router; the lesser bits (hostID) identifying the user's terminal.

The user's Internet address will not identify a particular Router if the user is connected to an elongated private network that gains access to the Internet via several Routers.

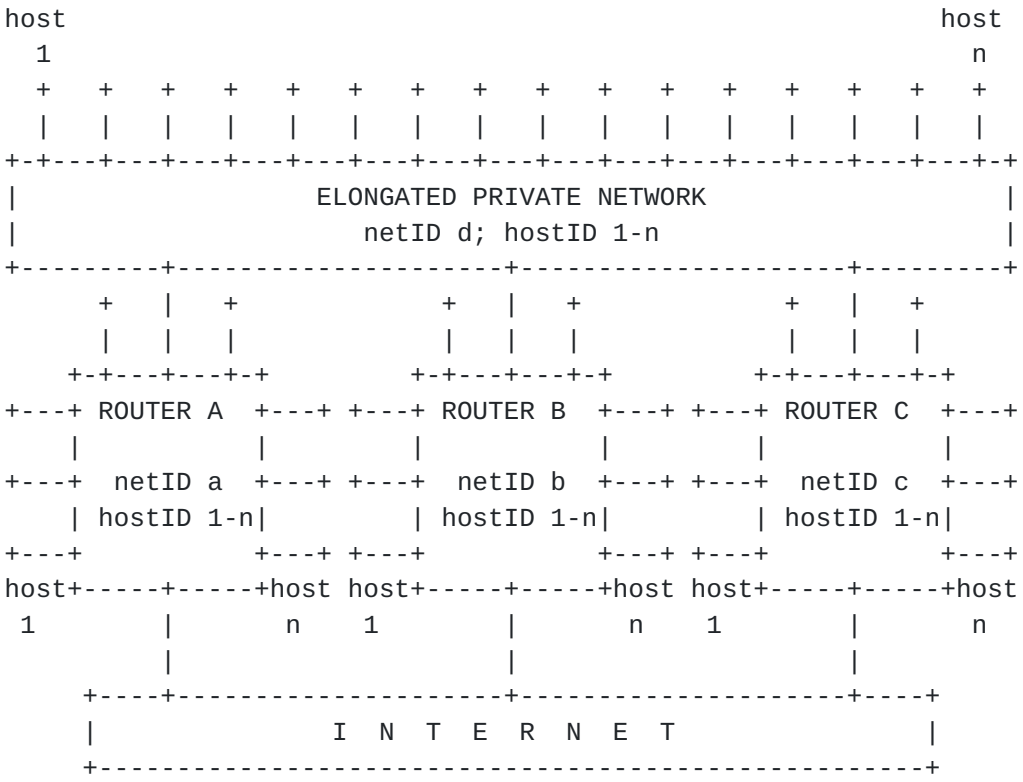


FIGURE 1. INTERNET ADDRESSES

Such users present no problems for simple Internet sessions; whichever Router is used to establish a virtual-message-path at the start of a session will be the Router used throughout the session. However, when users have to be picked-up for special-service delivery, the address used must identify the Router that invoked the service.

4.2.2. Service Delivery addresses.

To deliver a "special-service", a Server establishes a virtual message-path through the Internet to pick-up the message-path to the user at the user's Internet Router. To establish the path, the Server uses a "service delivery address" allocated

by the Router at the time of service invocation. The netID part of the address identifies the Router, the hostID part enables the Router to identify the current session-record.

4.2.3. Special-service networks.

The future Internet address structure will include "networks" (netIDs) that identify Brokers and Services but have no relation to the location of the Routers and Servers. A single address (hostID) within such a network might identify all the worldwide services provided by General Motors of America, adjacent addresses identifying Siemens of Germany, Reuters News Agency or Mitsubishi of Japan.

Thus, the netID part of the address is merely a "special-service" identifier and the hostID part is no more than an index number identifying the service.

5. TRANSPORT LAYER PROTOCOL MIGRATION STRATEGY.

In order to accommodate existing Internet equipment, all new equipment (hosts and Routers) will continue to handle the existing protocols (IP/TCP,etc.).

As the penetration of new style equipment increases, new transport layer protocols will be introduced which are more suited to a connection-oriented environment, but some time will elapse before such protocols are universally available.

The old protocols will continue to be used when no other protocol is mutually available. When old protocols are used in a connection-oriented manner, the IP header is redundant and should be ignored or even omitted. (It would be used only to make a checksum?)

To accommodate the introduction of new protocols, when opening a session, a user will indicate in the OPEN message the old and new protocols available for the transport layer session. The distant end will indicate in the OPEN_DONE message the chosen mutually available protocol. (If old network equipment is encountered, the session will use old network protocols.)

6. CREATING A VIRTUAL-MESSAGE-PATH. (SEE FIG.2, page 11)

6.1. The host/Router link.

Internet terminals are called "hosts". A host may be a PC capable of little more than one task at a time; an ISDN type terminal with several ports, each capable of several sessions; or a mainframe computer with innumerable hardware and software "ports", each capable of countless simultaneous sessions.

Whatever the configuration, a session within a host can be

uniquely identified by a "port" and "session" number.

A session-reference number identifies a session on the link between a host and its parent Router. All messages sent or received during a session will have the session-reference number in the host/Router link header. The reference may be supplemented by a sub-session number during multi-sessions.

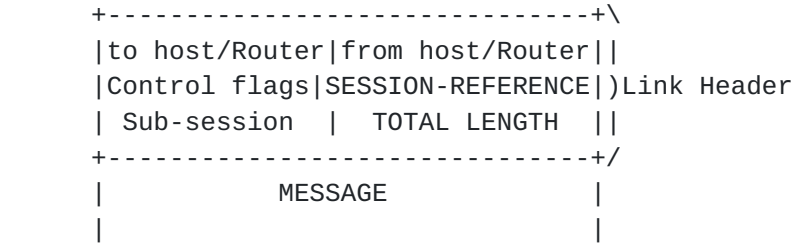
The host relates the session-reference to the appropriate port and session, the Router relates the host/session-reference to the route and virtual-channel-number (VCN) that forms the next link in the chain toward the distant end.

The session-reference number will be allocated by the host on originating sessions or by the Router on terminating sessions.

A host passes control messages (OPEN/CLOSE) received from the Internet to its control port. On hosts which deliver special-services, this port will require a standard port number in order to allow new version SERVICE-REQUEST messages to be received via old version Internet equipment. (See 8.2.2.1.)

6.1.1. The host/Router link header.

The host/Router link header might appear as follows:



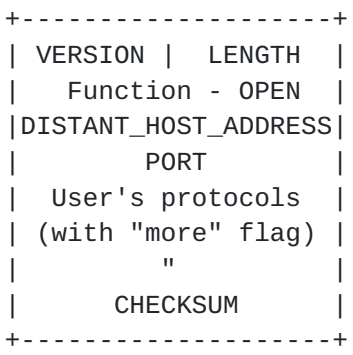
The control flags may be used as follows:

00 - ordinary message packet (new version),
10 - session control message (OPEN/CLOSE etc.)
01 - (not used)
11 - old version message packet (no session-reference)

6.2. The OPEN message.

To open a virtual-message-path through the Internet, a host sends an OPEN message to its Internet Router. The message contains the distant host's address and port number which are derived from the Internet name and protocol indicator (ftp, http, etc) specified by the user.

As new transport layer protocols are introduced, more than one protocol indicator might indicate the same port (ftp, newftp). The OPEN message will list the different protocols that are available at the user end and the distant end will indicate the chosen protocol in the OPEN_DONE message.



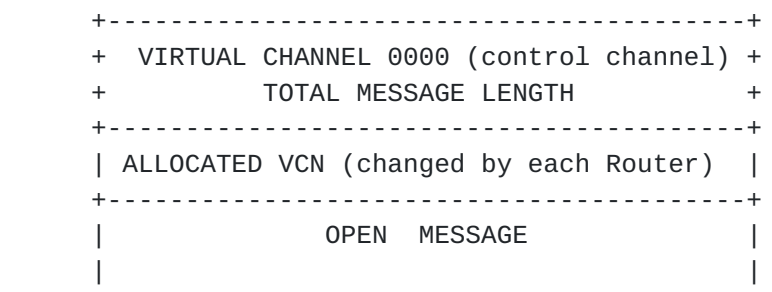
TYPICAL OPEN MESSAGE

6.3. The message-path through the Internet.

6.3.1. VCN Allocation.

Each Router uses the distant-host-address in the OPEN message to identify the route towards the destination and allocates a VCN to identify the session on the link to the next Router. The OPEN message is then passed on the control channel (VCN 0000?) to the next Router in a message packet carrying the allocated VCN where the process will be repeated.

At the link level, an OPEN message being passed from Router to Router might appear as follows:



Messages received on a link's control channel (VCN 0000) will be passed to the Router's session processer for attention.

To avoid VCN collisions, the Router at one end of a link may allocate VCNs in the range 0001 - 7FFF while the Router at the other end allocates 8000 - FFFF. (This assumes that VCN 0000 is used in both directions for control messages.)

To accommodate the existing and other network-layer protocols, link implementation may find it convenient to allocate a separate control VCN for each protocol. Messages received on such VCNs will be passed directly to the appropriate handler.

6.3.2. Switching tables.

As sessions are opened and closed, Routers will update tables which, for each incoming route (link) and VCN, will indicate

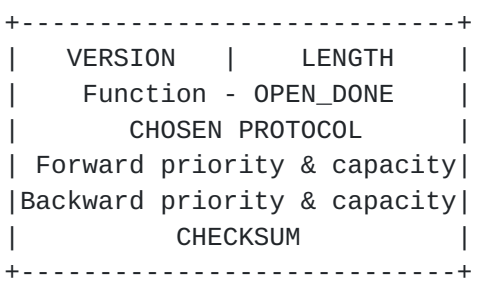
the corresponding outgoing route, VCN and message switching priority. Each session will have two complimentary entries in

the tables; one for each direction of transmission.

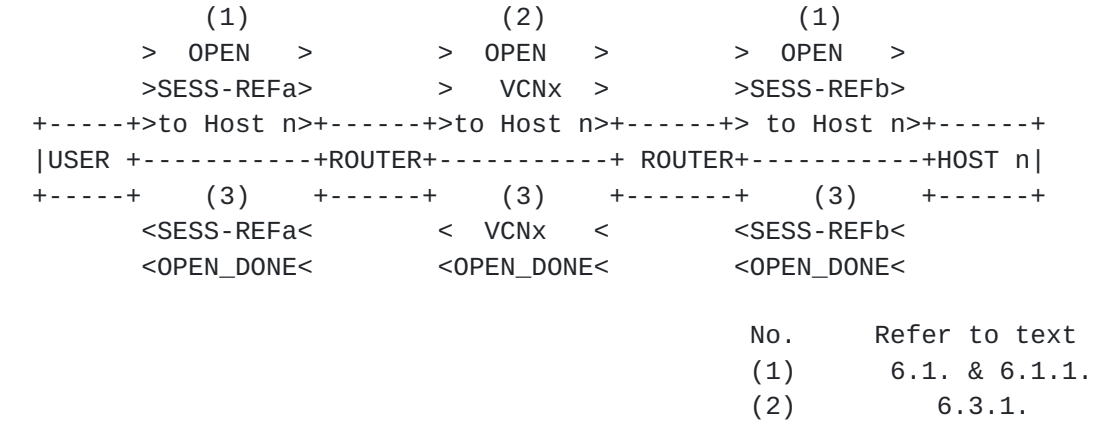
For charging or traffic recording purposes, Routers will also record which party established each session (which party is paying). Whilst this information could be derived from the session records, it may be convenient to have a flag in the switching tables.

6.3.3. The OPEN_DONE message.

When the message-path is complete, the distant host will return an OPEN_DONE message indicating the chosen protocol, and the capacity and message switching priority to be reserved in the network in both directions.



TYPICAL OPEN_DONE MESSAGE



No.	Refer to text
(1)	6.1. & 6.1.1.
(2)	6.3.1.

(3) 6.3.3.& 6.3.4.

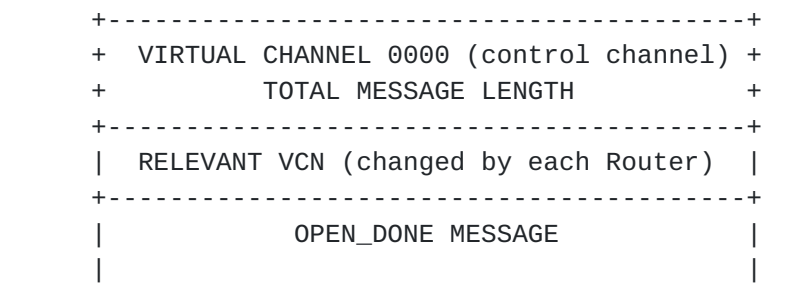
ESTABLISHING A VIRTUAL-MESSAGE-PATH

FIGURE 2

The OPEN_DONE or FAILURE message will be returned link by link

using the control channel (VCN 0000); each Router amending the RELEVANT VCN to indicate the session to which the message applies.

At the link level, an OPEN_DONE message being passed from Router to Router might appear as follows:



To accommodate hosts which require use of a wake-up procedure, the final Router may return an OPEN_ACK message indicating that the message-path is complete but that there will be a delay before the OPEN_DONE message can be sent.

All subsequent control messages - in either direction - will be sent on the control channel in a similar manner, with the link header being followed by the RELEVANT VCN and the control message itself.

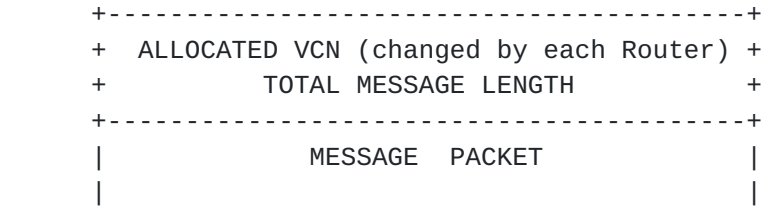
6.3.4. Reserving capacity.

Routers will reserve capacity (in terms of anticipated mean bits per second) on the links employed as indicated in the OPEN_DONE message, and will refuse traffic which exceeds the capacity available.

A Router will convert an OPEN_DONE message into a FAILURE message if it finds that it cannot provide the capacity indicated in the OPEN_DONE message. Upon receiving a FAILURE message the originating host will be required to send a CLOSE message to release the established part of the message-path.

6.3.5. Using the message-path.

Once a message-path is established, the originating host will open a transport layer session. Message packets will be passed from Router to Router with the Link Header indicating the allocated VCN. Each Router will replace the incoming VCN with the outgoing VCN as the message packet is switched.



6.3.6. Diverting a message-path. (2nd OPEN_DONE message.)

To OPEN a simple Internet session, a user may need to OPEN an initial session with the distant host's Port Mapper in order to learn the appropriate port number.

With new version equipment, the distant host should be able to divert the Port Mapper session to the required port without requiring the user to re-open the session.

There may then be need for a control message to change the chosen protocol and reserve appropriate network capacity.

The message content will be seen to be identical to the OPEN_DONE message. Users and Routers should be prepared to accept a second such message, including that a Router may change the message to a FAILURE message if it finds that it cannot provide the required capacity.

An OPEN_DONE message may also be sent to a user more than once during special-service delivery when a session is transferred from one Server to another.

6.3.7. Closing the message-path.

When the session ends, the transport layer session will be closed followed by closure of the virtual-message-path.

Message-paths are controlled by the originating terminal and closed by the exchange of CLOSE and CLOSE_ACK messages. The

CLOSE_ACK message being returned on a per link basis. The terminating end may invite closure with a CLOSE_REQUEST message.

7. SPECIAL-SERVICES. (See Figures 3 & 4. PAGES 16 & 17)

7.1. Invoking special-service delivery.

To open a special-service session in the Internet, a host sends an OPEN message to its Internet Router as normal. The message includes the distant-host-address derived from the Internet name specified by the user.

When the user specifies the name of a special-service, the netID part of the distant-host-address will be a "special-service" identifier. (See 4.2.3.)

Upon receiving a "special-service" address, the Router returns a SERVICE_ACK message to the host and sends a SERVICE_REQUEST message to the nearest SORTER. (A Sorter is located with any convenient Router and has an address in the range used by any other host connected to that Router.)

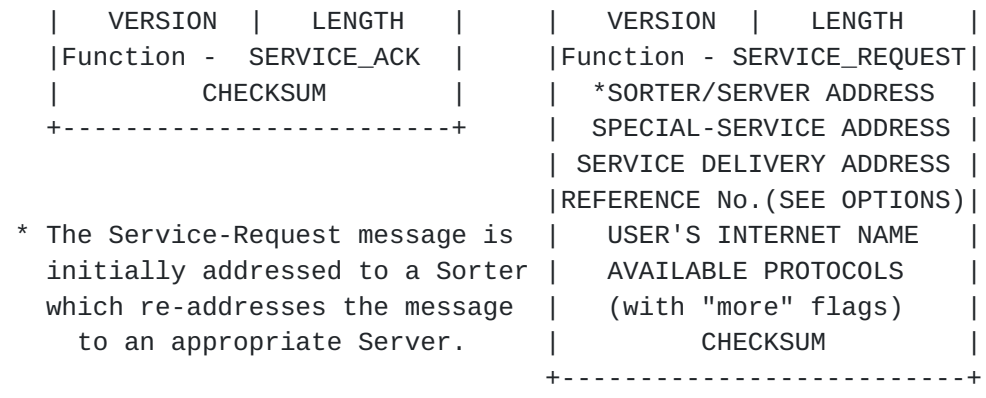
The SORTER opens the SERVICE_REQUEST message to identify the special-service address and uses a look-up table to re-address the message to the appropriate Broker or Server.

In the case of services which are provided by other network operators or in other countries, the Sorter will re-address the message to a Sorter in that network or country; there is no need for internetwork management of Sorters.

7.1.1. The SERVICE_ACK message.

The SERVICE_ACK message contains no parameters. It informs the user that the session may involve more than one transport layer session and that the initiative to open and close those sessions will be at the distant end. The user must assume the default situation, - that messages will be sent and received via old network equipment and will have IP/TCP type headers.

All messages sent or received during the session (old protocol or otherwise) will use the current message-path; link headers quoting the SESSION_REFERENCE and the control bits set to 00 (or 10 for new version session control messages). See 6.1.1.



TYPICAL SERVICE_ACK & SERVICE_REQUEST MESSAGE

7.1.2. The SERVICE_REQUEST message.

The SERVICE_REQUEST message contains;
(i) the special-service address
(ii) the service-delivery address & reference no. See OPTIONS
(iii) the user's Internet name (prime user's name)
(iv) the transport layer protocols available.
Items i and iv are taken from the OPEN message, items ii & iii are provided by the Router.
OPTION 1 This option enables the service-delivery address to be used as the DESTINATION-ADDRESS in IP headers when services are delivered via old network equipment, but requires a block of hostIDs to be reserved for this purpose. The SERVICE DELIVERY ADDRESS is an ordinary Internet address, the netID identifies the Router, the

hostID is a reference number enabling the Router to identify the active session-record.
OPTION 2 Enables address economy by using a single SERVICE DELIVERY ADDRESS and placing the reference number in another field. This requires a field to be found in IP Headers to carry the reference when services are delivered via old network equipment. (Not difficult?)

SERVICE_REQUEST messages use the control channel like any other control message and create a message-path as they are passed via Routers and Sorters to the Server; each Router and Sorter allocating a VCN or session reference as necessary.

The message-path created by the SERVICE_REQUEST message is used only to return a REQUEST_DONE or FAILURE message from the Server via the Sorter to the originating Router. The Router will CLOSE the path when the returned message is received. No

capacity is reserved for the SERVICE_REQUEST session.

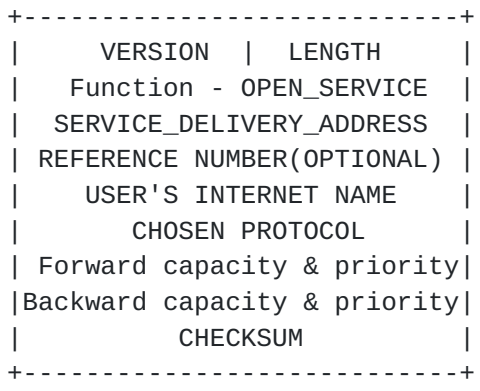
7.2. Establishing the service delivery path.

7.2.1. The OPEN_SERVICE message.

Upon receiving the SERVICE_REQUEST message, the Server will choose the protocol to be used and open a message-path to the user via the Internet with an OPEN_SERVICE message containing:

(i) the SERVICE-DELIVERY ADDRESS as DISTANT-HOST-ADDRESS
(ia) the REFERENCE NUMBER (OPTIONAL. See 7.1.2. OPTIONS)
(ii) the USER'S INTERNET NAME
(iii) the transport layer protocol to be used
(iv) the capacity and switching priority to be reserved

Items iii and iv are valid only if a complete virtual-message-path can be established and are used only after being transferred to an OPEN_DONE message.



TYPICAL OPEN_SERVICE MESSAGE

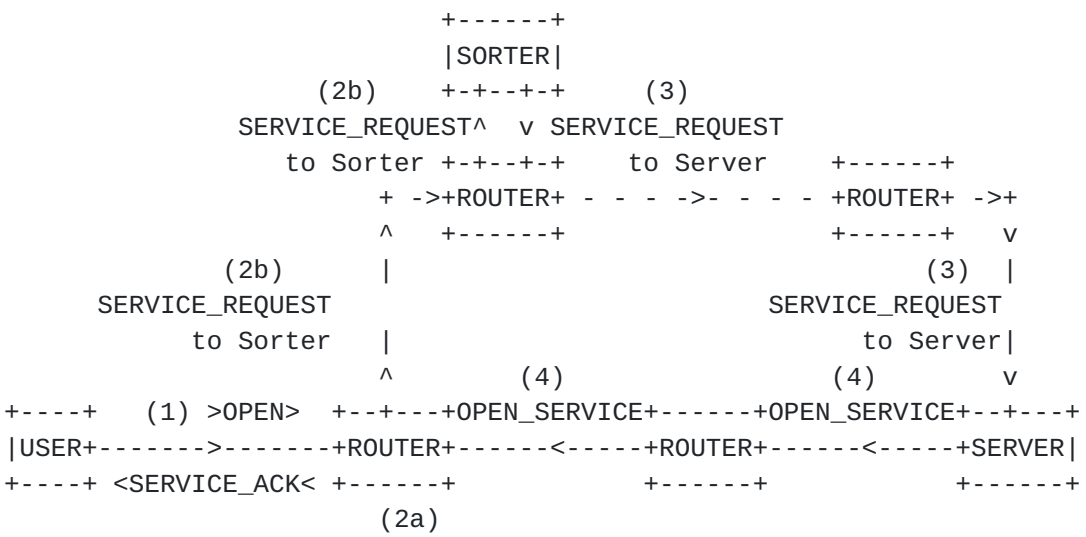
The OPEN_SERVICE message will be treated as a normal OPEN message by all Routers except the final Router, which uses the SERVICE_DELIVERY_ADDRESS (and REFERENCE NO.? See 7.1.2.) to find the session record, verify the USER'S INTERNET NAME and

pick-up the message-path to the user.

7.2.2. The OPEN_DONE messages.

7.2.2.1. OPEN_DONE to the Server.

The Router sends an OPEN_DONE message to the Server using the protocol, capacity and priority fields from the OPEN_SERVICE message. It is used by all Routers to reserve capacity. To the Server, it confirms the protocol to be used and indicates that a complete virtual-message-path has been established.

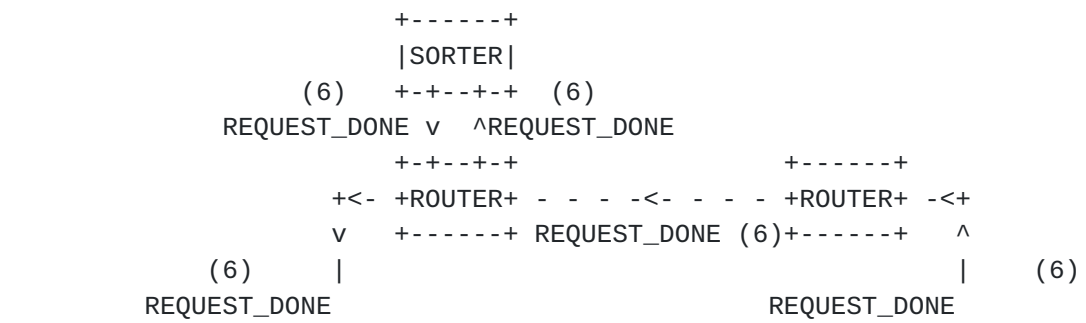


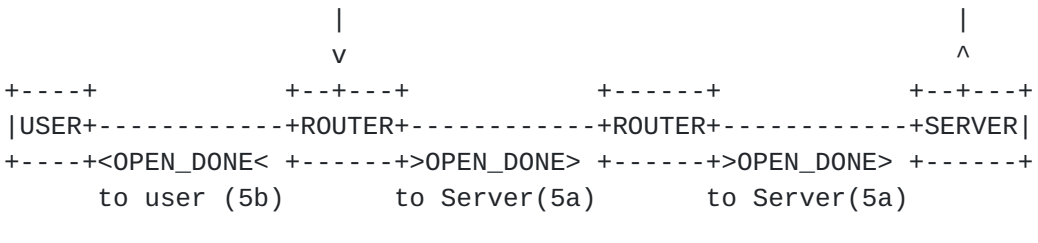
No.	Message	Refer to Text
(1)	OPEN	7.1
(2a)	SERVICE_ACK	7.1.1.
(2b)	SERVICE_REQUEST	7.1.2.
(3)	SERVICE_REQUEST	7.1.2.
(4)	OPEN_SERVICE	7.2.1.

- ->- - path of SERVICE_REQUEST message
---<--- virtual-message-path & direction of establishment

CREATING A SPECIAL-SERVICE MESSAGE-PATH

FIGURE 3.





No.	Message	Refer to Text
(5a)	OPEN_DONE to Server	7.2.2.1.
(5b)	OPEN_DONE to User	7.2.2.2.
(6)	REQUEST_DONE	7.2.4.

- - - - - - - - - - SERVICE_REQUEST message path
---------- established virtual-message-path

After receiving REQUEST_DONE the user's Router closes the SERVICE_REQUEST message path

PATH-COMPLETE MESSAGES

FIGURE 4.

7.2.2.2. OPEN_DONE to the user.

The same OPEN_DONE message will be sent to the user but with the forward and backward capacity and priority fields reversed (the capacity and priority may be required by a LAN Gateway?)

The message tells the user which transport layer protocol will be used for service delivery and overrides the default old network protocol established by the SERVICE_ACK message. At the end of the transport layer session, the user must revert to the default. (See 7.1.1.)

7.2.3. Control message interface.

The user's Internet Router will be required to provide an interface for subsequent control messages as both the user and Server will consider themselves to be the session originator. For charging purposes, the session must be considered to be originated by the Server.

7.2.4. The REQUEST_DONE message.

Having received OPEN_DONE indicating that the message-path is complete, the Server will return a REQUEST_DONE message to the

originating Router using the message-path created by the SERVICE_REQUEST message; the Router will then CLOSE that path.

7.3. Delivering the service.

Using the established message-path, the Server will open a transport layer session with the user to commence service delivery. This will often involve interrogating the user to further identify the service required after which the user may be transferred to another Server.

7.4. Service Transfer. (See Figures 5 & 6, pages 20 & 21)

7.4.1. The TRANSFER_REQUEST message.

A Server transfers a user to another Server by closing the transport layer session and sending a TRANSFER_REQUEST message to the other Server. The TRANSFER_REQUEST message contains the same information as the original SERVICE_REQUEST message, but indicates that the existing message-path must be diverted.

The message will usually be addressed directly to the other Server but may be addressed via a Sorter as before, in which case the SPECIAL_SERVICE_ADDRESS field must be updated.

The message packet may include information for the new Server obtained during the earlier session (regarding payment etc.). All such information is of no consequence to the Internet.

The TRANSFER_REQUEST message will be forwarded to the new Server and a message-path created to return the REQUEST_DONE or FAILURE message. (Similar to a SERVICE_REQUEST message.)

7.4.2. The OPEN_TRANSFER message.

The new Server will use a OPEN_TRANSFER message to establish a message-path via the Internet to the user. The message content is the same as an OPEN_SERVICE message and is treated as an ordinary OPEN message by all Routers except the final Router that connects to the user or user's LAN. This Router uses the SERVICE_DELIVERY_ADDRESS (and REFERENCE_NO. See 7.1.2.) to find the session record, verify the USER'S NAME and complete the virtual-message-path to the user.

7.4.3. OPEN_DONE and CLOSE_REQUEST messages.

Having completed the new message-path, the Router will send an OPEN_DONE message to the new Server and to the user as before (See 7.2.2) and will send a CLOSE_REQUEST message to the old

Server on the old message-path. (If the previous session used old network procedure, no CLOSE_REQUEST message can be sent.)

The previous Server will CLOSE the old message-path when the CLOSE_REQUEST message is received. (See also 8.2.4.3.)

Upon receiving the OPEN_DONE message, the user will cancel the default situation (restored when the previous transport layer session was closed. See 7.2.2.2) and await the opening of a new transport layer session using the protocol indicated in the OPEN_DONE message.

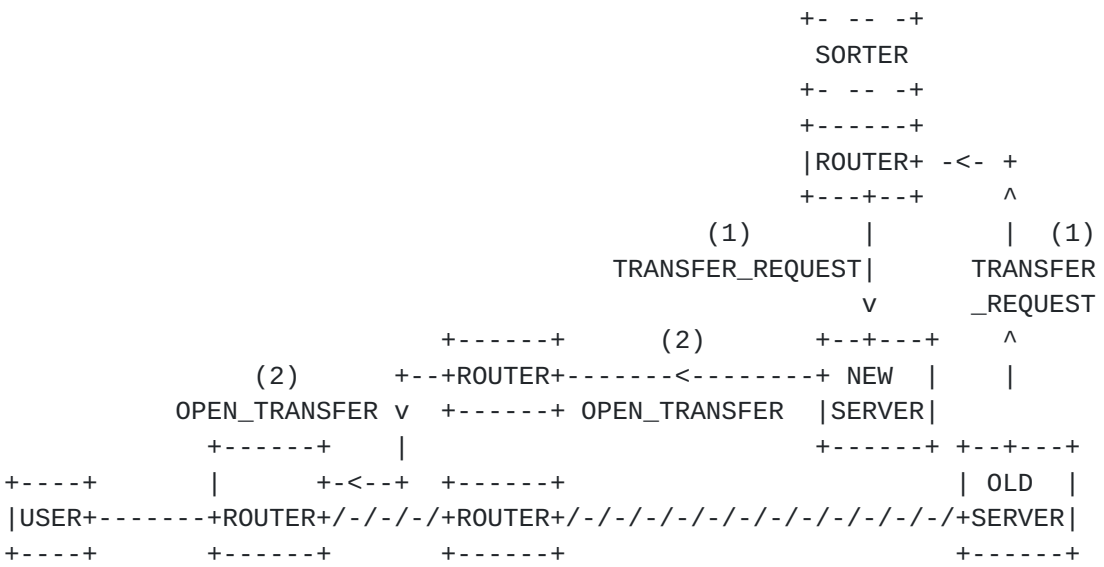
7.4.4. The REQUEST_DONE message.

When the new Server receives the OPEN_DONE message, it will send REQUEST_DONE to the old Server on the TRANSFER_REQUEST message-path and will commence service delivery.

After receiving REQUEST_DONE, the old Server will close the TRANSFER_REQUEST message-path.

7.5. Subsequent transfers.

The session transfer activity may be repeated any number of times.

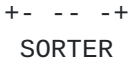


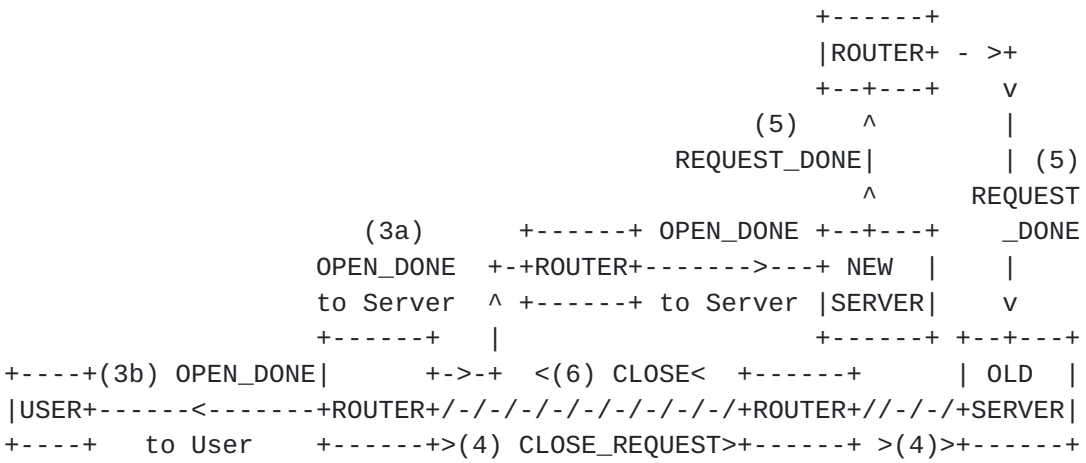
No.	Message	Refer to text
(1)	TRANSFER_REQUEST	7.4.1.
(2)	OPEN_TRANSFER	7.4.2.

- - ->- -path of TRANSFER_REQUEST message
-----<---new virtual-path & direction of establishment
-/-/-/-/-old virtual-path

INITIATING SERVICE TRANSFER

FIGURE 5.





No.	Message	Refer to text
(3a)	OPEN_DONE to Server	7.4.3.
(3b)	OPEN_DONE to User	7.4.3.
(4)	CLOSE_REQUEST	7.4.3.
(5)	REQUEST_DONE	7.4.4.
(6)	CLOSE	7.4.3.

- - - - - path of TRANSFER_REQUEST message
--------- new virtual-path
-/-/-/-/- old virtual-path

After receiving REQUEST_DONE the old Server closes the REQUEST message-path

PATH-COMPLETE MESSAGES

FIGURE 6.

8. ACCOMMODATING EXISTING EQUIPMENT.

To interwork with existing equipment, new version hosts and new version Routers must be able to handle the existing procedures and must be able to receive and send old version messages. Control flags indicate the presence of old version messages on host/Router links (See 6.1.1.); on links between Routers, old version messages use a special VCN. (See 6.3.1)

8.1. A simple session in the unknown Internet. See Fig 7, p22)

Although capable of old network procedures, new version hosts will never initiate old version sessions. Even when a session is known to involve old version equipment, a new version host will attempt to establish a virtual-message-path; ensuring as far as possible, the continuity of the session. (If only the distant host were old version equipment a virtual-message-path could be established right across the Internet.)

8.1.1. The OPEN message.

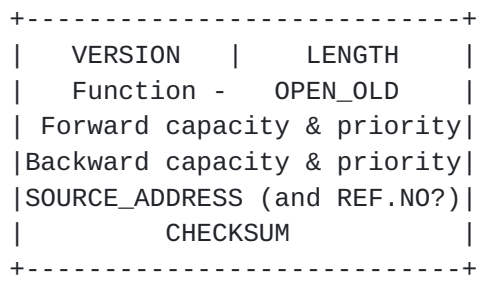
When a new version host sends an OPEN message into the Internet, a message-path will established to the distant host or to the point that the path meets old version equipment.

If the path meets old version equipment, the Router at the interface to the old equipment completes the path to the old network route and returns OPEN_OLD to the originating end.

8.1.2. The OPEN_OLD message.

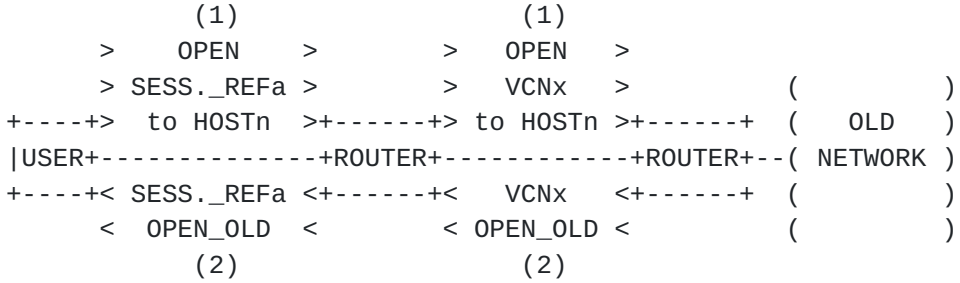
The OPEN_OLD message informs the user that the message-path is open but is not complete and that old network procedures must be used. The message includes a SOURCE_ADDRESS for use in the IP Headers; it identifies the interface Router and its active session-record. (An "option" achieves address economy by using an additional REFERENCE_NO field. See SERVICE_DELIVERY_ADDRESS in SERVICE_REQUEST messages. Section 7.1.2.)

The OPEN_OLD message also indicates nominal capacities and priorities to be reserved by the Routers.



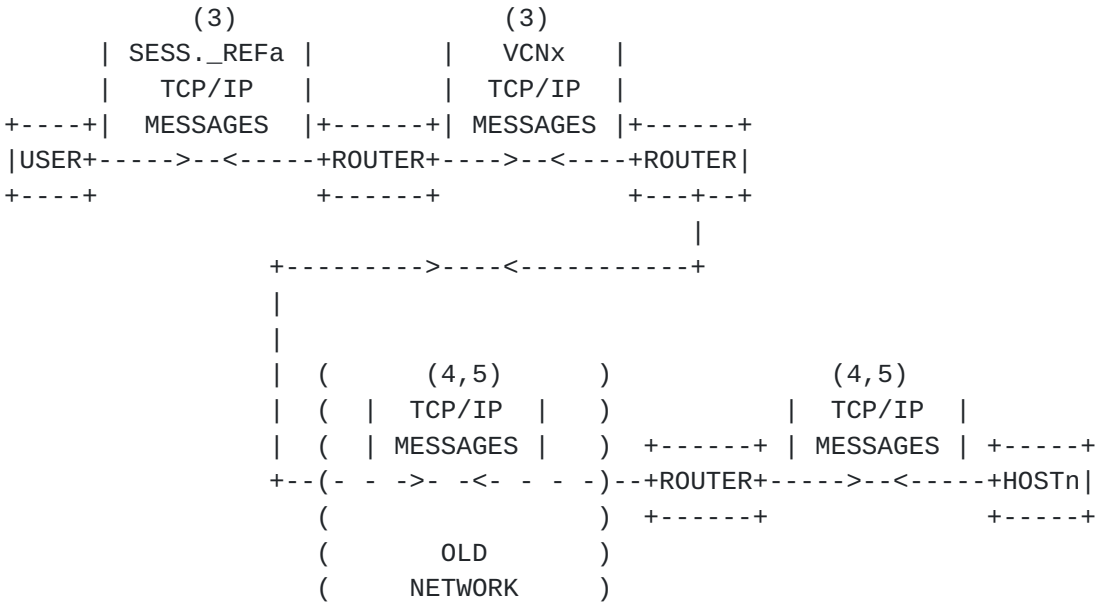
TYPICAL OPEN_OLD MESSAGE

Once an interface to old version equipment has been provided, it is not possible to return to the new version procedures if later equipment has new version capability.



No.	Refer to text
(1) OPEN	8.1.1. ESTABLISHING AN INCOMPLETE MESSAGE-PATH
(2) OPEN_OLD	8.1.2. FIGURE 7.

[page 22



No.	Refer to text
(3)	8.1.3.
(4)	8.1.3.1.
(5)	8.1.3.2.

USING AN INCOMPLETE MESSAGE-PATH

FIGURE 8.

8.1.3. Conducting the session. (See Figure 8)

The user will send and receive messages with IP/TCP type headers on the established message-path. (i.e. With the link-header using the SESSION-REFERENCE and the control bits set to 00.) The DESTINATION_ADDRESS in the IP header will be the same as the distant-host-address in the OPEN message, the SOURCE_ADDRESS (and REFERENCE NO?) will be that provided in the OPEN_OLD message.

8.1.3.1. Forward messages.

Upon receiving forward messages from the user, the Router at the interface will pass the messages to the old network route. The messages will continue to their destination using old network procedures.

8.1.3.2. Backward messages.

Messages returned from the distant end will be addressed to the SOURCE_ADDRESS given in the forward messages, which (with an optional REFERENCE NO.?) identifies the interface Router and its session-record.

Upon receiving the returned messages, the interface Router

will find the session-record and return the messages, with headers, to the user on the established message-path.

8.1.3.3. Closing the session.

When the session ends, the originating host will CLOSE the message-path to the interface Router in the normal manner.

8.2. A special-service session in the unknown Internet.

8.2.1. Invoking the service.

When a new version host attempts to OPEN a session with a special-service address, the Internet Router addresses a SERVICE_REQUEST message to the nearest Sorter which re-addresses the message to an appropriate Server.

If an old version host attempts to gain access to such a service, the special-service address will appear in the DESTINATION_ADDRESS of an IP header. A variety of treatments are possible; but are not detailed in this document.

8.2.2. SERVICE_REQUEST via old network. (See Figure 9, p 26)

8.2.2.1. UDP/IP Header.

If the message path from the Router via the Sorter to the chosen Server encounters old version equipment, the Router at the interface will prefix the SERVICE_REQUEST message with an IP/UDP Header and forward it via the old version equipment.

For this purpose, new version Routers will have a ready-made IP/UDP Header. The Router will be required to complete the

DESTINATION_ADDRESS, CHECKSUM and LENGTH fields each time the header is used. The ready-made header may also be used to forward TRANSFER_REQUEST messages.

A Sorter will be required to accept IP/UDP messages only when old network equipment exists in the path between the Sorter and the Routers which it serves; or when SERVICE_REQUEST messages are received from Sorters in other networks or other countries (See 7.1. last para). Having received a request message using IP/UDP, the message should be forwarded to the Server in the same manner.

The SERVICE_REQUEST message will be delivered using the IP/UDP protocol to the Server's control port (a standard port number on special-service equipment. See 6.1. last para.).

8.2.2.2. The ACK_OLD message.

Servers do not acknowledge SERVICE_REQUEST messages delivered by IP/UDP. Having forwarded the message, the interface Router

will return ACK_OLD to the originating Router.

The ACK_OLD message has no parameters. The originating Router will CLOSE the partly established REQUEST message-path (used only to return DONE or FAILURE messages from the Server) and await the commencement of service delivery; a time-out and escape procedure being available if no response arises.

8.2.3. Service delivery via old network. (See Figure 10, p26)

8.2.3.1. The OPEN_SERVICE message.

Having received the SERVICE_REQUEST message (by whatever means), the Server will attempt to open a message-path to the user by sending a OPEN_SERVICE message into the Internet.

If the message-path meets old version equipment, the Router at the interface to the old equipment will complete the path to the old network route and return OPEN_OLD to the Server.

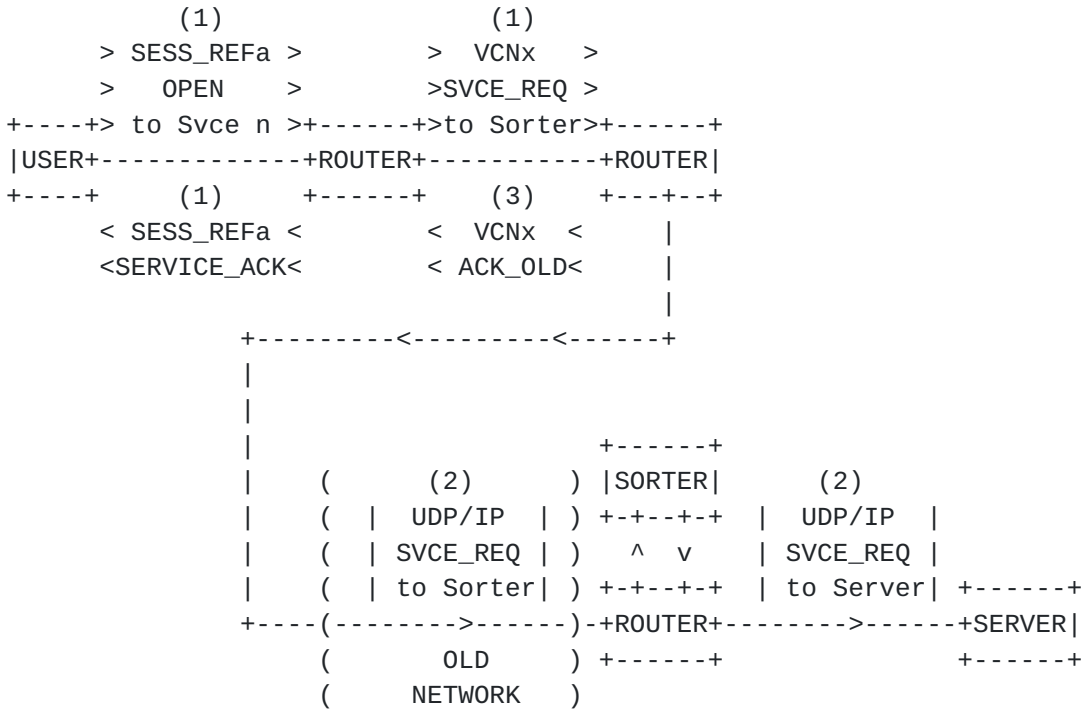
8.2.3.2. The OPEN_OLD message. (See 8.1.2.)

The OPEN_OLD message informs the Server that the message-path is not complete and that old network procedure must be used.

The message gives nominal capacity and priority reservations and includes a SOURCE_ADDRESS to be used by the Server in the

IP Headers. The address (including an optional REFERENCE_NO.) identifies the interface Router and its session-record.

Upon receiving the OPEN_OLD message, the Server will return REQUEST_DONE to the request message source. (If the request message delivery had also encountered old-network equipment, it will not be possible to return a REQUEST_DONE message.)



No. Refer to text
(1) 7.1; 7.1.1; & 7.1.2.

(2) 8.2.2.1.
(3) 8.2.2.2.

FIGURE 9. SENDING SERVICE_REQUEST VIA OLD NETWORK

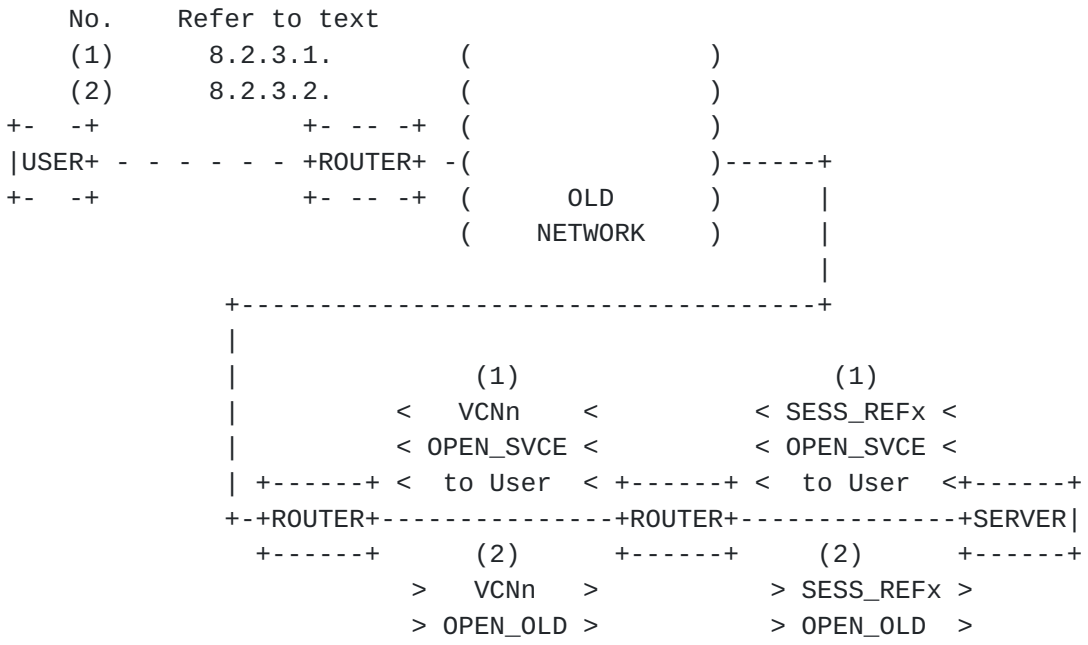
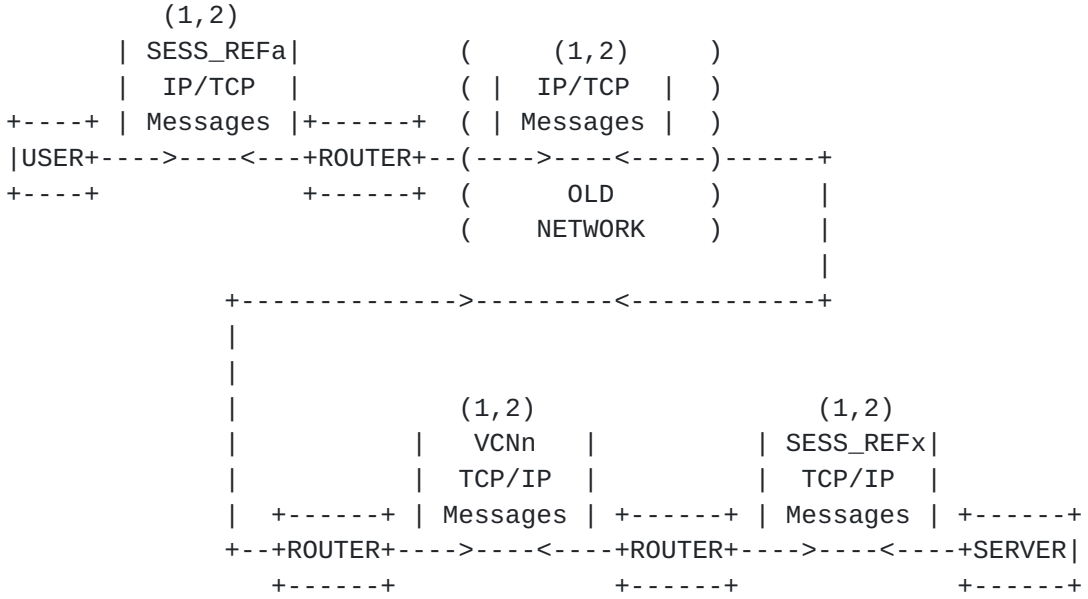


FIGURE 10. ESTABLISHING AN INCOMPLETE SERVICE-DELIVERY MESSAGE-PATH



No.	Refer to text
(1)	8.2.4.1.
(2)	8.2.4.2.

FIGURE 11. DELIVERING SERVICE VIA INCOMPLETE MESSAGE-PATH

8.2.4. Conducting the session. (See Figure 11, above)

8.2.4.1. Forward messages.

The Server will commence service delivery using the part established message-path. (i.e. The link header will hold the session_reference number with the control bits set to 00.)

Each message packet will have an old version IP/TCP header. The SOURCE_ADDRESS (and REFERENCE NO.?) in the header will be as received in the OPEN_OLD message, identifying the interface Router and its session-record. The DESTINATION_ADDRESS (and REFERENCE NO.?) will be the SERVICE_ DELIVERY_ADDRESS from the SERVICE/TRANSFER_REQUEST message, identifying the originating Router and its session-record.

The interface Router passes messages received from the Server into the old network which forwards the messages to the originating Router as identified by the DESTINATION_ADDRESS.

The originating Router recognises that the DESTINATION_ADDRESS in the forward messages is a SERVICE_DELIVERY_ADDRESS. It uses the address (and REFERENCE_NO?) to find the session-record and send the messages to the user on the established message-path, albeit only one link. (Link headers hold the session-reference number with the control bits set to 00.)

The user will be in the default situation and will be

expecting service delivery using old network protocols. The port and session to which the messages belong will be identified by the session-reference number in the link header; the IP Header in the forward messages will be used only for checksums and to provide the address for returned messages.

8.2.4.2. Backward messages.

To simplify the handling of backward messages, the originating Router may record in its switching table the route from which forward messages are received. The table must be updated each

time a forward message is received as the route may change without notice if service delivery is transferred to a Server which also needs to use old network procedure.

Messages returned to the Router from the user will have IP/TCP type headers and will use the established message-path (the link-header holding the session-reference and the control bits set to 00). The DESTINATION_ADDRESS will be the SOURCE_ADDRESS from the forward messages, which (with a REFERENCE NO.?) identifies the interface Router and its session-record.

The Router processes the messages using old-network procedure.

When the interface Router receives the messages it uses the DESTINATION_ADDRESS (and REF.NO?) to find the session-record and return the messages to the Server on the established message-path.

8.2.4.3. Closing the previous message-path (if any).

If the service delivery were the result of a service transfer, the originating Router's records would hold references to the path used for the earlier session. The records will be updated and a CLOSE_REQUEST message sent on the previous message-path.

The previous Server will CLOSE the previous message-path when it receives the CLOSE_REQUEST message.

If the previous session encountered old-network equipment, no CLOSE_REQUEST message could be sent. The previous Server would CLOSE the previous incomplete message-path after receiving REQUEST_DONE from the new Server; or after receiving ACK_OLD in response to its TRANSFER_REQUEST message indicating that REQUEST_DONE would not be received.

8.2.5. Closing the session.

When service-delivery is complete, the final Server will CLOSE the part established message-path in the normal manner. The user will CLOSE the host/Router link when no more transfers are expected.

9. MULTI-SESSIONS

A user may establish several simultaneous sub-sessions in the pursuit of a session. No special handling is required in the Internet as the user would be able to relate the different

Internet sessions as being part of one master-session.

A multi-session may also be formed by a special-service Server adding other Servers instead of transferring to other Servers; or by a mixture of user and Server created sub-sessions.

A Server adds another Server by sending an ADD_REQUEST message to the new Server containing the same information that would be contained in a TRANSFER_REQUEST message.

The new Server will establish a message-path to the user with an OPEN_ADD message containing the same information that would be contained in an OPEN_TRANSFER message.

The user's Router will return an OPEN_DONE message to the new Server which will send REQUEST_DONE to the Server that initiated the ADD.

The user's Router will also allocate a sub-session number before sending an ADD_DONE message to the user containing similar information to that contained in the OPEN_DONE message. (See 7.2.2.2.)

Upon receiving the ADD_DONE message, the user's host will open a sub-session as indicated by the SESSION_REFERENCE and sub-session number in the link header.

The ADD facility is dependant upon new version equipment. There is no means of separating the different sub-session windows if old version equipment is encountered.

10. THE FUTURE INTERNET - CHARGING.

10.1. Basic charges.

Internet users will pay a fixed-fee to their Internet Provider and will probably pay an additional usage charge, each session being charged to the session originator.

Session charges will depend upon the time that a session remains open (and upon the number and size of the messages carried?). Session charges should be sufficient to deter users from blocking network resources by keeping sessions open unnecessarily.

10.2. Special-service charges.

Sessions originated by Brokers or Servers in response to

service or transfer request messages will be charged to the Broker or Server. Some Service Providers may choose to pay the session charges (like Freefone in the telephone network) while others may not only pass-on the session charges to the user but levy an additional charge for the actual service delivered (like Premium Rate services).

10.3. Collecting Service Provider's charges.

To deliver special-services, Brokers and Servers are given the USER'S INTERNET NAME. Most simply, the charge for the service should be collected from that name.

Collecting service delivery charges on behalf of the Service Providers could be a very attractive and very profitable service provided by the Internet Operators.

Recording the charges levied by Servers for service delivery will be the responsibility of the Service Providers, - not a problem for the Internet! (A Broker may elect to pay for a session between a Server and a user - then add commission before charging the user.)

11. SECURITY.

11.1. User identity verification.

The user-friendliness and commercial security of this proposal are interdependant and lie in the fact that services are delivered directly to the user who has no knowledge of the service's source or the detail of its invocation. The Server is given the user's identity which is verified by the user's Internet Router from the session records. (See 7.2.1.)

11.2. Breeching security.

The commercial security of this proposition is in part attributable to the fact that services are delivered to a verifiable user name.

This verification is not possible when services are delivered via old version equipment as the DESTINATION_ADDRESS in the IP header is an address which identifies a session-record and the user's identity is not included.

Also, when message-paths are established via old version equipment, the user is given an address which identifies a session-record to use as the SOURCE_ADDRESS in IP headers.

Having established a message-path via old version equipment, a knowledgeable and meddlesome user could extricate this number and use it to build SERVICE or TRANSFER_REQUEST messages with a bogus user-name.

The user may also cheat the system when services are known to be delivered via old version equipment, by sending REQUEST messages to Servers containing a false user's name but with the true user's address in the SERVICE_DELIVERY_ADDRESS field. This method may not be so attractive to a dishonest user as it requires revealing the true address.

Consequently, Servers must be aware that the user's identity cannot be verified when services are delivered via old network equipment.

********DOCUMENT EXPIRES MARCH 1997**********

Authors' addresses

P.J. Williams
IN / R&S
GPT Limited
PO Box 53
New Century Park
Coventry CV3 1HJ
UK

Ian Davies
Central Engineering
GPT Limited
PO Box 53
New Century Park
Coventry CV3 1HJ
UK

Please note - Pip Williams (who did all the work!) is now a consultant to GPT and not usually available except on Wenesdays. He is not directly accessible by e-mail. Please address comments to Ian Davies (daviesic@ncp.gpt.co.uk) tel. +44 1203 562755 or fax +44 1203 562566.