Network Working Group
Request for Comments: 1575
Obsoletes: 1139
Category: Standards Track

S. Hares
Merit/NSFNET
C. Wittbrodt
Stanford University/BARRNet

February 1994

An Echo Function for CLNP (ISO 8473)

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Abstract

This memo defines an echo function for the connection-less network layer protocol. The mechanism that is mandated here is in the final process of being standardized by ISO as "Amendment X: Addition of an Echo function to ISO 8473" an integral part of Version 2 of ISO 8473.

Table of Contents

1. Conventions

The following language conventions are used in the items of specification in this document:

- MUST, SHALL, or MANDATORY -- the item is an absolute requirement of the specification.
- SHOULD or RECOMMENDED -- the item should generally be followed for all but exceptional circumstances.
- MAY or OPTIONAL -- the item is truly optional and may be followed or ignored according to the needs of the implementor.

2. Introduction

The OSI Connection-less network layer protocol (ISO 8473) defines a means for transmitting and relaying data and error protocol data units, (PDUs) or preferably, packets through an OSI internet. Unfortunately, the world that these packets travel through is imperfect. Gateways and links may fail. This memo defines an echo function to be used in the debugging and testing of the OSI network layer. Hosts and routers which support the OSI network layer MUST be able to originate an echo packet as well as respond when an echo is received.

Network management protocols can be used to determine the state of a gateway or link. However, since these protocols themselves utilize a protocol that may experience packet loss, it cannot be guaranteed that the network management applications can be utilized. A simple mechanism in the network layer is required so that systems can be probed to determine if the lowest levels of the networking software are operating correctly. This mechanism is not intended to compete with or replace network management; rather it should be viewed as an addition to the facilities offered by network management.

The code-path consideration requires that the echo path through a system be identical (or very close) to the path used by normal data. An echo path must succeed and fail in unison with the normal data path or else it will not provide a useful diagnostic tool.

Previous drafts describing an echo function for CLNP offered two implementation alternatives (see RFC 1139). Although backward compatibility is an important consideration whenever a change is made to a protocol, it is more important at this point that the echo mechanisms used on the Internet interoperate. For this reason, this memo defines one implementation mechanism (consistent with one of the previous drafts).

3. The Generic Echo Function

The following section describes the echo function in a generic fashion. This memo defines an echo-request entity. The function of the echo-request entity is to accept an incoming echo-request packet, perform some processing, and generate an echo-response packet. The echo implementation may be thought of as an entity that coexists with the network layer. Subsequent sections will detail the implementation mechanism.

For the purposes of this memo, the term "ping" shall be used to mean the act of transmitting an echo-request packet to a remote system (with the expectation that an echo-response packet will be sent back to the transmitter).

3.1. The Echo-Request

When a system decides to ping a remote system, an echo-request is built. All fields of the packet header are assigned normal values (see implementation specific sections for more information). The address of the system to be pinged is inserted as the destination NSAP address. The rules of segmentation defined for a data (DT) packet also apply to the echo-request packet.

The echo-request is switched through the network toward its destination. (An echo packet must follow the same path as CLNP data packet with the same options in the CLNP header.) Upon reaching the destination system, the packet is processed according to normal processing rules. At the end of the input processing, the echo-request packet is delivered to the echo-request entity.

The echo-request entity will build and dispatch the echo-response packet. This is a new packet. Except as noted below, this second packet is built using the normal construction procedures. The destination address of the echo-response packet is taken from the source address of the echo-request packet. Most options present in the echo-request packet are copied into the echo-response packet (see implementation notes for more information).

3.2. The Echo-Response

The entire echo-request packet is included in the data portion of the echo-response packet. This includes the echo-request packet header as well as any data that accompanies the echo-request packet. The entire echo-request packet is included in the echo-response so that fields such as the echo-request lifetime may be examined when the response is received. After the echo-response packet is built, it is transmitted toward the new destination (the original source of the

echo-request). The rules of segmentation defined for a data packet also apply to the echo-response packet.

The echo-response packet is relayed through the network toward its destination. (A echo response packet must follow the same path as a CLNP data packet with the same options in the CLNP header.) Upon reaching its destination, it is processed by the packet input function and delivered to the entity that created the echo-request.

4. The Implementation Mechanism

The implementation mechanism defines two new 8473 packet types: ERQ (echo-request) and ERP (echo-response). With the exception of a new type code, these packets will be identical to the date packet in every respect.

4.1. The Echo-Request

The type code for the echo-request packet is decimal 30.

4.2. The Echo-Response

The type code for the echo-response packet is decimal 31.

5. Implementation Notes

The following notes are an integral part of memo. It is important that implementors take heed of these points.

5.1. Discarding Packets

The rules used for discarding a data packet (ISO 8473, Section 6.9 - Section 6.10) are applied when an echo-request or echo-response is discarded.

5.2. Error Report Flag

The error report flag may be set on the echo-request packet, the echo-response packet, or both. If an echo-request is discarded, the associated error-report (ER) packet will be sent to the echo-request source address on the originating machine. If an echo-response is discarded, the associated error-report packet will be sent to the echo-response source address. In general, this will be the destination address of the echo-request entity. It should be noted that the echo-request entity and the originator of the echo-request packet are not required to process error-report packets.

5.3. Use of the Lifetime Field

The lifetime field of the echo-request and echo-response packets should be set to the value normally used for a data packet. Note: although this memo does not prohibit the generation of a packet with a smaller-than-normal lifetime field, this memo explicitly does not attempt to define a mechanism for varying the lifetime field set in the echo-response packet. This memo recommends the lifetime value that would under normal circumstances by used when sending a data packet.

5.4. Echo-request function

This function is invoked by system management to obtain information about the dynamic state of the Network layer with respect to (a) the reachability of specific network-entities, and (b) the characteristics of the path or paths that can be created between network-entities through the operation of Network layer routing functions. When invoked, the echo-request function causes an echo-request (ERQ) packet to be created. The echo-request packet shall be constructed and processed by ISO 8473 network-entities in end systems and intermediate systems in exactly the same way as the data packet, with the following caveats:

a) The information available to the packet composition function (ISO 8473) consists of current state, local information, and information supplied by system management.

b) The source and destination address fields of the echo-request packet shall contain, respectively, a Network entity title (NET) of the originating network-entity and a Network entity title of the destination network-entity (which may be in either an end system or an intermediate system). NOTE: A Network entity title is syntactically indistinguishable from an NSAP address. The additional information in an NSAP address, if any, beyond that which is present in a Network entity title, is relevant only to the operation of the packet decomposition function in a destination end system, and therefore is not needed for the processing of an echo-request packet (from which no N-UNITDATA indication is ever produced). The fact that the source and destination address fields of the echo-request packet contain NETs rather than NSAP addresses therefore does not affect the processing of an echo-request packet by any network-entity.

c) When an echo-request packet has reached its destination, as determined by the Header processing (call HEADER FORMAT Analysis function in ISO 8473), the echo-response function shall handle this Network Protocol Data Units (NPDU) instead of the packet

decomposition function. In ISO 8473, the packet decomposition function is like a decomposing fish on the sea shore - it takes a packet down to its bare bones and processes it.

Also, it is up to each individual system whether or not handling echo-request packets involves system management. One example of involving system management is the reporting reception of the echo packets as some systems do with the ping packet. Some systems find this of value if they are being pinged to death.

d) The maximum length of the echo-request packet is equal to the maximum length of the echo-response packet minus the maximum length of the echo-response packet header. This ensures that the entire echo-request packet can be contained within the data field of the echo-response packet (see ISO 8473).

e) The data part of the echo-request packet may, as a local matter, contain zero or more octets with any values that fit within the echo-request packet. (see (d) above for maximum length of the echo-request packet). If the first octet of data is binary 1000 0001, then an echo-response header is contained in the echo-request packet. The existence of this header insures that a router can formulate a standard echo-response packet.

Normally, the "more segmentation" flag in the encapsulated echo-response packet header shall be zero, and the segmentation portion of the encapsulated packet shall not be included. The segmentation length in the echo-response packet header shall be zero.

If the "more segmentation" flag is set in the encapsulated echo-response packet header, then a segmentation length shall be filled in and the segmentation part of the echo-response packet header will be present in the echo-response header. This same segmentation function shall be present in the echo-response sent by the router.

NOTE: However, this formulated echo-response is not required between any two systems. With a common format for an echo-request packet used in an environment such as the Internet, the echo-response header may not be needed, and may in fact be unnecessary overhead.

5.5. Echo-response function

This function is performed by a network-entity when it has received an echo-request packet that has reached its destination, as determined by the Header format analysis function (ISO 8473 clause 6.3) that is, an echo-request packet which contains, in its destination address field, a Network entity title that identifies the network-entity. When invoked, the echo-response function causes an

echo-response (ERP) packet to be created. The echo-response packet shall be constructed and processed by ISO 8473 network-entities in end systems and intermediate systems in exactly the same way as the data packet, with the following caveats:

a) The information available to the packet composition function consists of current state, local information, and information contained in the corresponding echo-request packet.

b) The source address field of the echo-response packet shall contain the value of the destination address field of the corresponding echo-request packet. The destination address field of the echo-response packet shall contain the value of the source address field of the corresponding echo-request packet.

c) The echo-request packet, in its entirety, shall be placed into the data part of the echo-response packet. The data part of the echo-response packet shall contain only the corresponding echo-request packet.

d) If the data part of the echo-request packet contains an echo-response header, the packet composition function may, but is not required to, use some or all of the information contained therein to select values for the fields of the echo-response packet header. In this case, however, the value of the lifetime field contained in the echo-response packet header in the echo-request packet data part must be used as the value of the lifetime field in the echo-response packet. The values of the segment length and checksum fields shall be computed by the network-entity regardless of the contents of those fields in the echo-response packet header in the data part of the echo-request packet.

e) The options part of the echo-response packet may contain any (or none) of the options described in ISO 8473 (but see Section 5.7 of this RFC). The values for these options, if present, are determined by the network-entity as a local matter. They may be, but are not required to be, either identical to or derived from the corresponding options in the echo-request packet and/or the echo-response packet header contained in the data part of the echo-request packet (if present). The source routing option in the echo-response packet shall not be identical to (copied from) the source routing option in the echo-request packet header. If the recording of route option in the echo-response packet is identical to (copied from) the recording of route option in the echo-request packet header, the second octet of the parameter value field shall be set to the value 3.

f) It is a local matter whether or not the destination network-entity performs the lifetime control function on an echo-request packet before performing the echo-response function. The destination network-entity shall make the same decision in this regard that it would make, as a local matter, for a data packet in accordance with ISO 8473.

5.6. Use of the Priority Option

The 8473 priority function indicates the relative priority of packet. 0 is normal and 14 is the highest. Packets with higher values will be transmitted before lower values. The specific action upon receiving a 8473 packet with the priority field set is a "LOCAL MATTER". These means, any two systems could do it differently.

Hopefully, in the future, Internet routers will handle this as a priority queueing function. Some implementors consider the priority queueing function to be a cap. For example, if a router is congested, all those packets with priorities higher than 20, will be allowed through, and those with priority less than 20 will be dropped.

In short, the basic function of priority has wide latitude in the ISO specification. This wide latitude of implementation needs to be narrowed for implementations within a common network environment such as the Internet. The 8473 priority function is rarely implemented in today's Internet. The transmission of an echo-request packet with a priority set may provided unexcepted results until a more wide spread deployment of the priority feature in 8473 capable routers and end systems.

However, if the priority function must be used it is the safest value may be the value 0 - which indicates Normal priority. It most likely this value will follow the 8473 pathways.

In the future, as the implementation of the priority function further Internet documents will need to deal with its expected use.

5.7. Use of the Source Route Option

Use of the source route option in ISO 8473 may cause packets to loop until their lifetime expires. For this reason, this memo recommends against the use of the source route option in either an echo-request or echo-response packets. If the source route option is used to specify the route that the echo-request packet takes toward its destination, this memo does not recommend the use of an

automatically generated source route on the echo-response packet.

5.8. Transmission of Multiple Echo-Requests

The echo function may be utilized by more than one process on any individual machine. The mechanism by which multiple echo-requests and echo-responses are correlated between multiple processes on a single machine is a local matter and not defined by this memo.

6. Security Considerations

Security issues are not discussed in this memo.

7. Authors' Addresses

Susan K. Hares
MERIT/NSFNET
Internet Engineering
1075 Beal Avenue
Ann Arbor, MI 48109-2112

Phone: (313) 936-3000
EMail: skh@merit.edu

Cathy J. Wittbrodt
Stanford University/BARRNet
Networking Systems
Pine Hall 115
Stanford, CA 94305

Phone: (415) 725-5481
EMail: cjw@magnolia.Stanford.EDU

8. References

[1] ISO/IEC. Protocol for Providing the Connectionless-mode Network Service. International Standard 8473, ISO/IEC JTC 1, Switzerland, 1986.

[2] Hagens, R., "An Echo Function for ISO 8473", RFC 1139, IETF-OSI Working Group, January 1990.

[3] ISO 8473-1993 Protocol for providing the connectionless-mode network service, edition 2 (IS under preparation).