Network Working Group IETF-OSI Working Group
Request for Comments: 1139
R. Hagens
January 1990

An Echo Function for ISO 8473

Status of this Memo

This memo defines an echo function for the connection-less network layer protocol. This memo is not intended to compete with an ISO standard. This is a Proposed Elective Standard for the Internet. Distribution of this memo is unlimited.

Abstract

This memo defines an echo function for the connection-less network layer protocol. Two mechanisms are introduced that may be used to implement the echo function. The first mechanism is recommended as an interim solution for the Internet community. The second mechanism will be progressed to the ANSI X3S3.3 working group for consideration as a work item.

When an ISO standard is adopted that provides functionality similar to that described by this memo, then this memo will become obsolete and superceded by the ISO standard.

1. Introduction

The OSI Connection-less network layer protocol (ISO 8473) defines a means for transmitting and relaying data and error report PDUs 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.

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.

There are three important issues to consider when defining an echo extension to ISO 8473: complexity, code-path divergence, and backward compatibility. The complexity of the echo facility must be kept low. If it is not, then there is a good chance that the facility will not be universally provided. The code-path consideration requires that the echo path through a system is 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.

Backward compatibility is an important consideration whenever a change is made to a protocol. For this reason, this memo defines two implementation mechanisms: the short term approach and the long term approach. The short term approach will produce echo packets that are indistinguishable from normal data ISO 8473 PDUs. These echo packets may be switched through ISO 8473 routers that do not implement the echo function. The short term approach will be adopted as an Elective Internet Standard because it is backward compatible with ISO 8473. However, due to its nature, the short term approach will never be incorporated into future versions of ISO 8473.

The long term approach will produce echo packets that are not compatible with the existing standard. However, the long term approach may be acceptable by ISO as an addendum to ISO 8473. In this event, backward compatibility will no longer be an issue. At that juncture, the short term approach defined by this memo will be obsolete and superseded by the ISO addendum.

2. The Generic Echo Function

The following section will describe 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 PDU, perform some processing, and generate an echo-reply PDU. Depending on the echo implementation, the echo-request entity may be thought of as an entity that exists above the network layer, or as an entity that co-exists with the network layer. Subsequent sections will detail the short and long term implementation mechanisms.

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

2.1 The Echo Request

When a system decides to ping a remote system, an echo-request is built. All fields of the PDU 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 DT PDU also apply to the echo-request PDU.

The echo-request is switched through the network toward its destination. Upon reaching the destination system, the PDU is processed according to normal processing rules. At the end of the input processing, the echo-request PDU is delivered to the echo-request entity.

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

2.2 The Echo Reply

The entire echo-request PDU is included in the data portion of the echo-reply PDU. This includes the echo-request PDU header as well as the any data that accompanies the echo-request PDU. The entire echo-request PDU is included in the echo-reply so that fields such as the echo-request lifetime may be examined when the reply is received. After the echo-reply PDU is built, it is transmitted toward the new destination (the original source of the echo-request). The rules of segmentation defined for a DT PDU also apply to the echo-reply PDU.

The echo-reply PDU is relayed through the network toward its destination. Upon reaching its destination, it is processed by the PDU input function and delivered to the entity that created the echo-request.

3. The Short Term Implementation Mechanism

The short term implementation mechanism will use an ISO 8473 normal data PDU as the echo-request and echo-reply PDU. A special NSAP selector value will be used to identify the echo-request and insure that it reaches the echo-request entity. This selector value is known as the echo-request selector. In addition, an echo-reply selector is defined so that the echo-reply PDU may be identified at the destination system. It is important to note that (except for the NSAP selector) the echo-request PDU and the echo-reply PDU are indistinguishable from a DT PDU.

This approach has the advantage that it is simple and does not allow any code-path divergence. In addition, this approach requires that only the systems which wish to generate an echo-reply PDU must change. Systems that do not adhere to this memo will not generate an echo-reply PDU, but will still switch other echo-request and echo-reply PDUs.

3.1 The Echo Request

An echo-request is built using the normal DT PDU construction procedures. All fields of the PDU header are assigned normal values (see implementation notes). The address of the system to be pinged is inserted as the destination NSAP address. The selector field of the destination NSAP address must contain the echo-request selector. The selector field of the source NSAP address must contain the echo-reply selector.

3.2 The Echo Reply

Except as noted below (see implementation notes), an echo-reply is built using the normal DT PDU construction procedures. The destination NSAP address is taken from the source address of the echo-request PDU.

3.3 Use of NSAP Selectors

The choice of echo-request and echo-reply NSAP selectors is a local matter. However, to insure interoperability, and as an interim measure until use of the directory service becomes widespread, this memo will recommend the following default values (specified in decimal):

Echo Request Selector - 30
Echo Reply Selector - 31

4. The Long Term Implementation Mechanism

The long term implementation mechanism will define two new 8473 PDU types: ERQ (echo-request) and ERP (echo-reply). With the exception of a new type code, these PDUs will be identical to the DT PDU in every respect.

4.1 The Echo Request

The type code for the ERQ PDU is decimal 30.

4.2 The Echo Reply

The type code for the ERP PDU 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 PDUs

The rules used for discarding a DT PDU (8473, sec 6.9 - sec 6.10) are applied when an echo-request or echo-reply is discarded.

5.2 Error Report Flag

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

5.3 Use of the Lifetime Field

The lifetime field of the echo-request and echo-reply PDU should be set to the value normally used for a DT PDU. Note: although this memo does not prohibit the generation of a PDU 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-reply PDU. This memo recommends that the normal DT lifetime value should be set in the echo-request and echo-reply PDU.

5.4 Transfer of Options from the echo-request PDU to the echo-reply PDU

With two exceptions, all options present in the echo-request header are copied directly into the echo-reply header. The two exceptions are the record route option and the source route option. A record route option present in an echo-request PDU is copied into the echo-reply PDU, but the routes recorded in the option are "erased" by resetting the second octet of the option to 3. This allows the entire record route option space to be used by the echo-reply PDU. Note: the record route present on the echo-request is not lost because the echo-request PDU is wholly contained in the data part of the echo-reply PDU.

The second exception concerns the source route option. A source route option present on the echo-request PDU is not copied into the echo-reply PDU.

5.5 Use of the Priority Option

If the priority option is included, it will normally be set to value 0 (default priority). This memo allows for priority values higher than 0 to be set in the echo-request or echo-reply header, but cautions against this practice.

5.6 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-reply PDU. If the source route option is used to specify the route that the echo-request PDU takes toward its destination, this memo does not recommend the use of an automatically generated source route on the echo-reply PDU.

5.7 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-replies are correlated between multiple processes on a single machine is a local matter and not defined by this memo.

Security Considerations

Security issues are not addressed in this memo.

Author's Address

Robert A. Hagens
Computer Science Department
1210 West Dayton Street
Madison, WI 53706

Phone: (608) 262-1204

EMail: hagens@CS.WISC.EDU