                TN3270 Extensions for LUname and Printer Selection

Status of this Memo

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

Abstract

   This document describes protocol extensions to TN3270. There are two
   extensions outlined in this document. The first defines a way by
   which a TN3270 client can request a specific device (LUname) from a
   TN3270 server. The second extension specifies how a TN3270 printer
   device can be requested by a TN3270 client and the manner in which
   the 3270 printer status information can be sent to the TN3270 server.
   Discussions and suggestions for improvements to these enhancements
   should be sent to the TN3270E Working Group mailing list
   TN3270E@list.nih.gov . These extensions will be called TN3287 in this
   document. This information is being provided to members of the
   Internet community that want to support the 3287 data stream within
   the TELNET protocol.

1. INTRODUCTION

   The need to communicate with IBM mainframe systems has a number of
   unique requirements associated with it. This document addresses
   those needs in a TCP/IP communications network.

   IBM terminals are generically referred to as 3270's which includes a
   broad range of terminals and devices,not all of which actually begin
   with the numbers 327x.

   The 3270 family of terminals and the IBM mainframe applications
   systems are VERY closely coupled and it is the nature of the way the
   3270s and the applications interact which require that this document
   be available to provide a consistent way for the TCP/IP environment
   to interact effectively with the 3270 applications of the IBM
   mainframe world.

IBM mainframe applications systems have existed for almost two
decades now and are used to serve tens of thousands of users daily.
For this reason it is usually the need of a mainframe environment to
add TCP/IP network support WITHOUT writing new applications to run
with the TCP network. The TN3270 series of documents addresses how
this can be done and maintain compatibility with those mainframe
application systems.

One of the unique characteristics of the 3270 terminals is their
ability to communicate status information in an out-of-band data
flow. These status's are in turn used by the applications systems to
support error recovery, and conflict resolutions, examples of these
are printer out of paper, and terminal powered up. The terminals are
also half duplex and block mode in their operations, which results in
the need to communicate when blocks are being sent, when they end,
and when they cannot be sent. This document describes these
characteristics in IBM VTAM/SNA terms. Some VM mainframe application
systems do not use VTAM, so for those systems these terms don't
apply. For any systems which use VTAM these terms apply and are
dealt with in some way by the TCP/IP to VTAM interface.

VTAM/SNA is a hierarchical network and some of that hierarchy needs
to be addressed by the TCP network attaching to it if the
applications systems are to continue to provide the same applications
support that they have provided to the 3270 terminals.

The 3270 terminal environment consists of a terminal controller with
terminals attached to that controller. In VTAM/SNA this controller
is called a PU (Physical Unit) and the terminals called LUs (Logical
Units). The PU is used to communicate management information to the
VTAM/SNA system, and the LU is used by the application to communicate
with the terminal. VTAM/SNA identifies each LU and PU in a network
by a unique name. These names are referred to as LUnames and
PUnames, and is how the network is managed and the applications
identify what terminals are being communicated with in the network.
The actual connection between a terminal and the applications is
referred to as a session, and it is this session which has both in-
band and out-of-band information flows sent between the applications
and the terminals.

VTAM/SNA 3270 terminals actually have two sessions when communicating
with the applications. One session is directly connected with the
application and the other session is connected directly to VTAM. It
is the session with VTAM, also called the SSCP, that is used to
communicate the out-of-band information flows. This session is
called the SSCP-LU session, and the session with the application is
called the LU-LU session (in VTAM an applications is just another
Logical Unit).

One such out-of-band flow is the LUSTAT message which tells the
application that the status of the terminal has changed, and is how a
printer or screen tells the application that it is ready, or is not
ready to receive data.

There are also flows which must be able to flow in the LU-LU session
to help control the use of the terminal by applications. The block
of information sent in a session is called an RU (Request Unit) and
it tells what type of data this block contains, how long it is and if
more data (RUs) is coming along. This is a gross over simplification
of what RUs are and do, but it should help understand their use in
the TN3270 documents. Some of the VTAM/SNA terms used to describe
what an RU is requesting are: Chains/chaining which tell a session
partner that another RU is being sent or not being sent in this
transmission. Brackets which are used to indicate that a unit of
work is complete, such as when a printout of a file is complete.

The determination of what part of the VTAM/SNA protocols such as
brackets and chaining are to be used are managed by VTAM tables
called LOGMODE tables. These tables are selected when an LU-LU
session is started and set up such things as bracket, and/or chaining
protocols; and the type of terminal data contained in the RUs, such
as printer data without screen formatting data (LU type 1), 3270
screen formatted data (LU type 2) and 3270 screen formatted data for
a printer (LU type 3). The LOGMODE tables also contain the size of
the RU to be sent and received. These tables also communicate the
screen size of 3270 terminals such as 24X80 (Model 2), 27X132 (Model
5), etc. Each LU has a LOGMODE table entry hard assigned to it as
part of the VTAM configuration (often called a GEN). The selection
of these table entries can't be controlled by the terminal LU or PU.
They can only be selected by the user at connection/logon time or by
the application when the connection is established. The actual
LOGMODE entries to be used during a session are sent at session logon
time, in a special type of RU called a BIND. Once the bind has been
sent then the rules for the use of the session have been set, can't
be changed, and must be followed.

The purpose of the TN3287 protocol is to provide a general IBM 3270
host printer communications facility. Its primary goal is to allow a
method of connecting printer devices and printer-oriented processes
to each other. This protocol will allow a TN3270 Client to process
3287 print data streams.

This memo supplements and extends the STD 8, RFC 854, TELNET Protocol
Specification. This memo also presents an example of the correct
implementation.

2. GENERAL CONSIDERATIONS

A TELNET connection is a Transmission Control Protocol (TCP) connection used to transmit data with interspersed TELNET control information.

The companion document, STD 8, RFC 854 -- "TELNET Protocol Specification" should be consulted for further information about the TELNET command, codes and code sequences referenced in this specification.

3. CLIENT-SERVER NEGOTIATION

The TN3270 Client and Server require a specific negotiation protocol. After the negotiation is complete, all transmission between the Client and Server is in TELNET Binary format with a TELNET "End-Of-Record(EOR)" sequence at the end of each data stream.

Support for the TN3287 data stream requires that both sides:

   A. Are able to exchange binary data.

   B. Can establish the agreement between client and server on the terminal type that will be used.

   C. Agree to use the TELNET IAC EOR as a delimiter for inbound and outbound TN3287 data streams

This implementation requires the options: TERMINAL-TYPE and BINARY be successfully negotiated between the Client and Server before processing of any print data streams.

This implementation supports host applications that can mix LU 1 and LU 3 type data in the data stream.

3.1 TN3287 SERVER

The maximum Request Unit (RU) size is server specific, but should not exceed 4 kilobytes.

The LU type is determined by the bind from the mainframe application. The server, when bound, must remember LU 1 or LU 3 type.

The server will automatically unbind the session upon receipt of a TELNET CLOSE command. The printer will be reported to VTAM as powered down until a new TELNET connection is established.

3.2 TN3287 CLIENT

   The TN3287 Client is a TN3270 client created specifically to print
   mainframe 3270 print data. The client emulates the IBM device type
   that it identifies itself to the TN3270 server as, in this case, an
   IBM 3287 model 1 type printer. The design of this printer protocol
   is aligned with the way printing occurs in the IBM host and how 3270
   printers function. These printer extensions DO NOT support a 3270
   printer client that cannot accept both types LU 1 and LU 3 printer
   streams. No IBM printer operates in this fashion, and as a result,
   no TN3270 server could function properly with mainframe applications
   if it didn't allow for a mixing of LU 1 and LU 3 data streams. The
   common way in which this can occur is printer sharing between
   multiple IBM host applications, such as CICS and JES. Since there is
   no restriction, the JES can be configured to output LU 1 data
   streams, and the CICS can be configured for LU 3 data streams.
   Therefore, the server will identify what LU type the current
   application connected to the server is using. If that type is LU 1,
   ALL message records sent to the Client will be preceded by one byte
   of binary zeros (0x00). If the first byte is not zeros, then that
   byte will be a valid LU type 3 Write-Command-Code(WCC), which can
   NEVER be zeros. Thus, the client can tell the LU type of data as
   each record is received.

   This protocol does allow for the client to shutdown if the client
   does not wish to support both LU types. This is accomplished by
   detecting an invalid data type from the received record, and
   notifying the user that the mainframe application has sent LU type x
   print data and should be configured for LU type y printing.

4.   COMMAND STRUCTURE

       1. All TELNET commands consist of at least a two-byte sequence:
          the "Interpret-as-Command(IAC)" escape character followed by
          the code for the command.

   NOTE: Since the TELNET IAC character (255 decimal) is used as a
   delimiter (together with EOR) in the inbound and outbound data
   streams, a data byte within the data stream itself that has the same
   value as the IAC command is sent as two bytes (255, 255) and one byte
   is discarded.

4.1 TELNET COMMANDS

   Command meaning - WILL and DO commands are used to obtain and grant
   permission for the subsequent subnegotiation. Both sides must
   exchange WILL TERM-TYPE and DO TERM-TYPE before subnegotiation.

The actual exchange of information is done within the option
subcommand.

<IAC DO TERMINAL-TYPE> Sender requests that the other party begin
terminal-type sub-negotiation.

<IAC WILL TERMINAL-TYPE> Sender is willing to send terminal-type
information in a subsequent sub-negotiation.

<IAC SB TERMINAL-TYPE SEND IAC SE> Sender requests the receiver to
transmit his terminal-type.

<IAC SB TERM TYPE IS IBM-3287-1 IAC SE>   Sender is stating the name
of his terminal-type. The code for <IS> is 0. Optionally, a
specific Logical Unit (LU) can be requested by using the TERMINAL-
TYPE string below.   If no LUname is specified, the first available
3287 LU is selected.

    IAC SB TERM-TYPE IS IBM-3287-1 @ LUNAME IAC SE

<IAC DO BINARY> Sender requests that sender of the data starts
transmitting or confirms that the sender of data is expected to
transmit characters that are to be interpreted as 8 bits of binary
data by the receiver.

<IAC WILL BINARY>   Sender requests permission to begin transmitting,
or confirms it will now begin transmitting binary data.

An <EOR> is sent at the end of each SNA Request Unit (RU) end of
chain, in either direction.   The first byte following the <EOR> is a
Write-Command-Code(WCC) for LU 3 data streams.

An <AO> is sent at the end of the SNA RU and end of bracket. This
signifies the end of the print output or file by the IBM host
application and possibly a change of LU type.

4.2   COMMAND VALUES

TELNET COMMAND	CODE
IAC Interpret as Command	255
DO	253
WILL	251
SB SuBnegotiation option	250
SE Subnegotiation End	240
TERMINAL-TYPE	24
SEND	1
IS	0

```
                    EOR  End-Of-Record              25
                    BINARY                          0
                    AO   Abort Output               245
                    IP   Interrupt Process          244
                    AYT  Are You There              246
                    BREAK                           243
```

   NOTE: The above codes and code sequences have the indicated meaning
   only when immediately preceded by an "Interpret as Command (IAC)".

5. TN3270 Printer Status Message

   The status message can be sent at any time. It must be sent every
   time the TN3270 Server sends an End-of-Record(EOR) indicator to the
   TN3270 Client, or when a printing error occurs at the Client. The
   Printer Status Message is only sent by the TN3270 Client. Once the
   End-Of-Record IAC is processed, the TN3270 Client sends the status
   message to the server when it is ready to receive more print data.

       MESSAGE DESCRIPTION: SOH % R S1 S2 IAC EOR

                        SOH = 0X01
                        % = 0X6C
                        R = 0XD9
                        S1 = Status/Sense Byte 0
                        S2 = Status/Sense Byte 1
                        IAC = Telnet IAC Character
                        EOR = Telnet EOR Character

5.1   Status/Sense Byte description

5.1.1. S/S Byte 0:

      High Order   Low Order

          _____
         | |
         | 0   1   2   3   4   5   6   7 |
         |_____|

         Bit Number: Bit Definition:

             0   Always Zero

             1   Always Zero

2	Always Zero
3	Always Zero
4	Always Zero
5	Unit Specify - is set due to an error condition. The reason for the error condition will be indicated in S/S Byte 1. See Note 1*.
6	Device End - when this bit sent in response to a data message it indicates the client has successfully processed the data message from the server and notifies the server to send a new data message to the client when available. See Note 2*.
7	Always Zero

   Note 1*:   A negative response to the Server's data message would be
   S/S Byte 0 Bit 5 "Unit Specify condition". The possible Unit Specify
   conditions are listed below. (See Section 3.2 for bit settings for
   the Unit Specify conditions listed below.)

           Unit Specify Condition: SNA Sense Code sent to host:

           Command Rejected   0X10030000
           Intervention Required   0X08020000
           Data Check   0X10010000
           Operation Check   0X10050000
           Component Disconnected (LU)   0X08020000

   Note 2*:   Device End - A positive response to the Server's data
   message would be the "Device End" bit (S/S Byte 0 Bit 6) to indicate
   a ready to receive data from the host condition. This will also be
   sent after clearing a previous Unit Specify condition of
   "Intervention Required".

5.1.2. S/S Byte 1:

High Order   Low Order

```
        _____
       |                                                                |
       |     0       1       2       3       4       5       6       7   |
       |_____|
```

Bit Number: Bit Definition:

        0   Always Zero

        1   Always Zero

        2   Command Rejected (CR) -- This bit
                              indicates an invalid 3270 command
                              generated.

        3   Intervention Required - Printer Not Ready.
                              See Note 3*.

        4   Component Disconnected - Printer is powered
                              off or printer cable not connected. See
                              Note 4*.

        5   Data Check - Invalid print data

        6   Always Zero

        7   Operation Check - An illegal buffer address
                              or incomplete order sequence

   Note 3*: The Intervention Required is cleared by sending an S/S
   message with the "Device End" bit (Bit 6 of S/S byte 0). The LUSTAT
   sent to the host is 0X00010000. The IBM host interprets this as a
   "printer now ready" condition.

   Note 4*: The Component disconnected is cleared by sending an S/S
   message with the "Device End" bit (Bit 6 of S/S byte 0). The LUSTAT
   sent to the host is 0X082B0000. The IBM host interprets this as a
   "printer now ready -- presentation space integrity may be lost"
   condition.

6. The following is an example of the Client-Server negotiation
    process.

        Server: IAC DO TERMINAL-TYPE
        Client: IAC WILL TERMINAL-TYPE
        Server: IAC SB TERMINAL-TYPE SEND IAC SE
        Client: IAC SB TERMINAL-TYPE IS IBM-3287-1 IAC SE

        Note: To request a specific LU the TERMINAL-TYPE string would be:
        IAC SB TERMINAL-TYPE IS IBM-3287-1 @ LUNAME IAC SE
        (The client has specified its terminal type is an IBM-3287-1)

        Server: IAC DO END-OF-RECORD
        Client: IAC WILL END-OF-RECORD
        Server: IAC WILL END-OF-RECORD
        Client: IAC DO END-OF-RECORD

        (The Server and Client have both agreed to transmit End-Of-Record
        (EOR)).

        Server: IAC DO TRANSMIT-BINARY
        Client: IAC WILL TRANSMIT-BINARY
        Server: IAC WILL TRANSMIT-BINARY
        Client: IAC DO TRANSMIT-BINARY
        (The Server and Client have both agreed to use binary
        transmission)

        Server: 0x00 (3270 PRINT DATA)
        Client: (S/S with DEV END) IAC EOR
        Server: 0x00 (3270 PRINT DATA) IAC EOR

        NOTE: LU 1 type data is prefaced with a 0x00 character. LU 3
        type data is not prefaced with a special character. This
        character will precede print data in each chain, and should be
        discarded before the print data is processed.   An <IAC EOR> must
        be received before changing to LU 1 or LU 3 type data.

        Client: (S/S with IR) IAC EOR (This indicates a paper jam
                  at printer.)
        Client: (S/S with DE) IAC EOR (This indicates the clearing
                  of above condition.)
        Server: 0x00 (3270 PRINT DATA) (This indicates start of LU 1
                  data)

        Server: (3270 PRINT DATA)

```
        Server:    (3270 PRINT DATA)
        Server:    (3270 PRINT DATA) IAC EOR
        Client:        (S/S with DE) IAC EOR
        Server:    0x00 (LAST 3270 PRINT DATA) IAC EOR


        Client:          (S/S with DE) IAC EOR
        Server:    IAC AO
        (The Abort Output <AO> signifies the end-of-bracket -- end of
        print job)
```

7. SECURITY CONSIDERATIONS

This document does not specify a security methodology to insure that
the client requesting a printer LU name is authorized to access that
LU. Currently, this is left up to individual server implementations.
The design of the protocols described in this document allow for the
future incorporation of the RFCs regarding encryptions and
authentication protocols and services. However, before this may
occur, certain extensions may be required to the protocols defined in
this document or to the encryptions and authentication services and
protocols.

8. ERROR CONDITIONS

After a client and server have successfully completed negotiation, a
number of potential error conditions may be detected by the server
which require notifying the client and aborting the connection.

When an error condition is detected by the server, the client must be
negotiated back into NVT mode by the server sending a "WONT/DONT
BINARY" TELNET sequence and the client responding with the
appropriate "DONT/WONT BINARY" TELNET sequence.

The server should immediately send the appropriate error message to
the client as an ASCII string and then close the connection. The
error message should be prefixed by a numeric identifier to precisely
notify the client of the specific error condition. The error message
sent to the client should be routed to the proper console or log for
corrective action.

Below is a list of error conditions identified by numeric value,
error text, meaning of the error and recovery procedure.

    Message: "01 No LU's of the type configured"

        Meaning: The configuration definition on the server
                 does not include the LU type requested.

          Recovery: Notify your Systems Administrator as this
                    is a permanent error condition.

       Message: "02 Requested LU unavailable"

          Meaning: The requested LU is not available at
                   this time.

          Recovery: This may be a temporary error and may
                    be retried periodically. If the condition
                    persists contact your Systems Administrator.

       Message: "03 Requested LU type is inconsistent with configuration"

          Meaning: The LU requested does not match the terminal
                   type in the server configuration.

          Recovery: Notify your Systems Administrator as this
                    is a permanent error condition.

       Message: "04 Requested LU is not configured"

          Meaning: The LU is not defined in server configuration.

          Recovery: Notify your Systems Administrator as this
                    is a permanent error condition.

    When a client receives a message not defined in the above list, the
    message should be displayed to a console or log and the connection to
    the server should be closed. No other recovery should be attempted
    as this is most likely a fatal error condition. (Notify your Systems
    Administrator.)

9. REFERENCES

   [1] Postel, J., and J. Reynolds, "TELNET Protocol Specification", STD
       8, RFC 854, USC/Information Services Institute, May 1983.

   [2] VanBokkeln, J., "TELNET Terminal-Type Option" RFC 1091, FTP
       Software Inc., February 1989.

   [3] Postel, J., and J. Reynolds, "TELNET Binary Transmission", STD
       27, RFC 856, USC/Information Services Institute, May 1983.

Authors' Addresses

> **Cleve Graves**
> **2711 LBJ Freeway**
> **Dallas, Texas 75234**
>
> **Phone: (214) 484-5200**
> **EMail: cvg@oc.com**
>
>
> **Thomas Butts**
> **2711 LBJ Freeway**
> **Dallas, Texas 75234**
>
> **Phone: (214) 484-5200**
> **EMail: tom@oc.com**
>
>
> **Michelle Angel**
> **2711 LBJ Freeway**
> **Dallas, Texas 75234**
>
> **Phone: (214) 484-5200**
> **EMail: angel@oc.com**