

Telnet Remote Serial Port (RSP) Option
<[draft-rfced-info-bennett-00.txt](#)>

Status of This Memo

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-abstract.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

This document is a submission to the Internet Engineering Task Force (IETF) as an Internet draft standard.

Distribution of this memo is unlimited.

1. Command Name and Code

RSP XX

2. Command Meanings

IAC WILL RSP

Sender is willing to send RSP information in a subsequent sub-negotiation.

IAC WON'T RSP

Sender refuses to send RSP information.

IAC DO RSP

Sender is willing to receive RSP information in a subsequent sub-negotiation.

IAC DON'T RSP

Sender refuses to accept RSP information.

IAC SB RSP SEND IAC SE

IAC SB RSP SEND item1 item2 ... IAC SE

Client requests server to transmit the current state of the given item(s). The code for SEND is 1.
If no items are given, the server is requested to return all known items.
(See below for valid 'item's)

IAC SB RSP IS item1 value1 item2 value2 ... IAC SE

Server states the current value of the given items.
May be in response to SEND, or may be due to a change in the value of the items. The code for IS is 0.
(See below for valid 'item's)

IAC SB RSP SET item1 value1 item2 value2 IAC SE

Client requests that the server sets the value of 'item1' to 'value1' etc. The code for SET is 2.
(See below for valid 'item's)

2.2 Valid RSP items

MSR 0

8250-compatible modem status register
Value is 32 bit value in network byte order, with msr in the least significant 8 bits.

BAUD-RATE 1

Baud rate
Value is 32 bit number in network byte order, greater than 0.

STOP-BITS 2

Stop bits
Value is 32 bit number in network byte order, where 0 represents 1 1/2.

DATA-BITS 3

Data bits in a character

Value is 32 bit number in network byte order.

PARITY 4

Parity of characters

Value is 32 bit number in network byte order with one of the following values.

NONE 0
ODD 1
EVEN 2
MARK 3
SPACE 4

DTR 5

State of DTR

Value is a 32 bit number in network byte order with one of the following values.

OFF 0
ON 1

RTS 6

State of RTS

Value is a 32 bit number in network byte order with one of the following values.

OFF 0
ON 1

TXBUFFER 7

Characters in the the transmit buffer

Value is a 32 bit number in network byte order with one of the following values.

EMPTY 0
NOT-EMPTY non-zero

FLOW-CONTROL 8

Input and output flow control setting.

Value is a 32 bit number in network byte order which represents a bit mask.

The bitmask may contain any combination of the following values.

RTSFLOW	1	(Input flow control - drop RTS to stop receiving)
CTSFLOW	2	(Output flow control - stop transmitting when CTS drops)
IXON	4	(Output flow control - start/stop sending on XON/XOFF)
IXOFF	8	(Input flow control - send XON/XOFF to start/stop receiving)
DCDFLOW	16	(Output flow control - stop transmitting when DCD

drops)

DTRFLOW	32	(Input flow control - drop DTR to stop receiving)
DSRFLOW	64	(Output flow control - stop transmitting when DSR

drops)

SEND-CHANGED 32

Specifies which items, if any, for which the server should send values when the value of the item changes.

Value is a 32 bit number in network byte order which represents a bit mask. If a bit is 1, then the server should send values for that item, and if it is a 0, it should send values only upon request. For example, if 0x00000001 is set as a value for SEND-CHANGED, then the server should send the value for MSR whenever that value changes.

PURGE 33

Purges receive and/or transmit queues.

Value is a 32 bit number in network byte order with one of the following values.

RX-QUEUE	1	
TX-QUEUE	2	
RXTX-QUEUE	3	(Purge both the RX and TX queues)

3. Default

WON'T RSP
DON'T RSP

RSP information will not be exchanged.

SEND-CHANGED 0

Server only sends updated information as requested.

4. Motivation for the Option

Many (if not all) terminal servers support the ability to set up a telnet listener on a serial port. This allows a connection to a port to be made via the network, however only (two way) data can be transferred between the client and the port.

By using the Remote Serial Port (RSP) telnet option, the client is able to control non-data aspects of the port such as baud rate and modem signals. This is especially important where the port is attached to a modem and modem signals are required to hangup the modem.

This document defines a simple protocol which allows terminal servers (and other systems which provide access to serial ports via a network connection) to provide access to these features.

5. Description of the Option

Willingness to exchange RSP information is agreed upon via conventional Telnet option negotiation. WILL and DO are used only to obtain and grant permission for future discussion. The actual exchange of status information occurs within option subcommands (IAC SB RSP...).

Once the two hosts have exchanged a WILL and a DO, the sender of the DO RSP (the client) is free to request and set RSP information. A successful DO RSP is required before SEND and SET requests can be used by the client and IS responses can be sent by the server. The negotiation indicates acceptance in one direction only. (It would generally not be appropriate to negotiate the RSP option in both directions).

All RSP items are 1 octet in length and all RSP values are 32 bit values in network byte order. This allows multiple items or multiple {item, value} pairs to be specified in a single request or response without requiring that all items be understood by the recipient.

6. Implementation Issues

This specification allows for various items to be set or requested by a client. Support for all items is optional. Also it may be inappropriate to set or send a given item at any time. Thus, a server responds to requests as follows.

When a server receives a SET request, it should immediately make the change for each item in the list. If any item is not supported, not understood or not appropriate (doesn't make sense or the value is inappropriate), that item is ignored. All items in a request are independent. That is, if one item fails, then the remaining items in the request should continue to be processed.

Similarly, when a server receives a SEND request, it should respond with a IS request for each item in the SEND request which is understood and makes sense. Other items are ignored.

No response is sent by the server as a result of a SET request. It is the responsibility of the client to issue a SEND request if it wishes to determine whether a SET request has succeeded.

SEND-CHANGED is an item like any other including the fact that support for it is optional. If it is supported, an IS request is sent by the server to the client whenever the value for an item changes value spontaneously (that is, other than in response to a SET request) for which a 1 bit is set in the SEND-CHANGED value.

Note: No requirements are placed on the server in determining how soon after the change occurs that the IS request should be

sent. This allows a server to combine multiple changes into a single request if multiple changes occur within a short time of one another. Also, a server may choose to delay sending of an IS request if the value of an item changes very quickly to avoid flooding the connection. Every change of an item need not be (and should not be) sent in these circumstances. Rather, only the "last" change should be sent.

Note: Only changes to the first 32 RSP items are supported by SEND-CHANGED. This should be sufficient for all realistic implementations of remote serial ports.

7. Examples

Note: In these examples, the C notation, 0x, is used to denote a hexadecimal number.

In this example, the client gets the current values for all items known by the server.

Client: IAC DO RSP

Server: IAC WILL RSP

(Client may now request or set RSP information at any time.
Server is SEND-CHANGED 0)

Client: IAC SB RSP SEND IAC SE

Server: IAC SB RSP IS SEND-CHANGED 0 MSR 0x00 DTR ON BAUD-RATE
9600 IAC SE

In this example, the client turns on SEND-CHANGED and checks to see if it is supported by the server. The server sends change notifications.

Client: IAC DO RSP

Server: IAC WILL RSP

Client: IAC SB RSP SET SEND-CHANGED 0x0000000F IAC SE

Client: IAC SB RSP SEND SEND-CHANGED IAC SE

Server: IAC SB RSP IS SEND-CHANGED 0x0000000F IAC SE

Server: IAC SB RSP IS MSR 0x33 IAC SE

Server: IAC SB RSP IS MSR 0x03 BAUD-RATE 38400 IAC SE

9. References:

- [1] Postel, J., and J. Reynolds, "Telnet Protocol Specification", [RFC 854](#), USC Information Sciences Institute, May 1983.
- [2] Postel, J., and J. Reynolds, "Telnet Option Specification", [RFC 855](#), USC Information Sciences Institute, May 1983.
- [3] Reynolds, J., and J. Postel, "Assigned Numbers", [RFC 1700](#), USC Information Sciences Institute, October 1994.

Author's Address

Murray Bennett
Stallion Technologies
33 Woodstock Rd
Toowong QLD 4066
Australia

Phone: +61 7 3270 4277

Email: murray@stallion.oz.au

Murray Bennett	Stallion Technologies
Phone: +61 7 3270 4277	33 Woodstock Rd
Fax: +61 7 3270 4245	Toowong QLD 4066
Email: murray@stallion.oz.au	Australia

INTERNET-DRAFT

EXPIRES: AUGUST 1997

INTERNET-DRAFT