Corrections to RFC 560: Remote Controlled Transmission & Echoing TELNET Option

(Postel)

Network Working Group
Request for Comments: # 581

NIC # 19860

References: RFC 560, RFC 563
Categories: Protocols, TELNET, RCTE

19860

D. Crocker (UCLA-NAG)
J. Postel (MIT/WHO-TIP)
2 Nov 73

(RECEIVED)

NOV 2 1973

Corrections to RFC 560
Remote Controlled Transmission & Echoing TELNET Option

(This RFC contains corrections to RFC 560 (NIC -- 18192,) which
described the Remote Controlled Transmission and Echoing TELNET
Option. A completely updated version of 18192 has been journalized
and will be included in the Protocols Notebook. These new
specifications for RCTE are in NIC document (19659).)

Section 1 of the RCTE option specification (18192,2a:gy) was supposed
to include the name and code for the option. The code was
accidentally left out. That statement should read:

RCTE 7

Section 2 should include the End of Subnegotiation Parameter, at the
end of the subnegotiation parameter specification (18492,2b:gy). All
examples in the option specifications, showing RCTE SB commands,
should also show the IAC SE parameter. (The revised RCTE
specifications have been so changed.) Section 2 should be changed so
that it reads:

IAC SB RCTE <cmd> [OC1 BC2] [TC1 TC2] IAC SE

The sample scenario, in Section 5.D (18492,2el:gy), should be
modified to reflect the kind of asynchrony of events that can occur
with the RCTE protocol. The updated RCTE specifications (in --
19859,1eh:gy) now reflect this.

In RFC 563 (18753,) John Davidson criticizes RCTE's apparent failure
to allow Net I/O and server computation to overlap.

I agree with John's criticisms and feel that the following should fix
the problem:
1. Change 5.A (16492,2e1)

from:

Overview of Interaction

to:

Overview of User Terminal Printing Action & Control

2. Change 5.B.5.a (16492,2e2e1)

from:

A Transmission character is one which REQUIRES the User Host to transmit all text accumulated up to and including its occurrence. (For Net efficiency, User hosts are DISCOURAGED from sending before the occurrence of a Transmission character).

to:

A Transmission character is one which RECOMMENDS that the Using Host transmit all text accumulated up to and including its occurrence. (For Net efficiency, Using hosts are DISCOURAGED from sending before the occurrence of a Transmission character, as defined at the moment the character is typed).

3. Change 5.B.5.b (16492,2e2e2)

from:

A Break character has the effect of a Transmission character, but also causes the Using host to stop its print/discard action upon the User's input text, until directed to do otherwise by another IAC SE <cmo> IAC SE command from the Serving host. Break characters therefore define printing units. "Break character" as used in this document does NOT mean Telnet Break character.

to:

A Break character has the effect of a Transmission character, but also causes the Using host to stop its print/discard action upon the User's input text, until directed to do otherwise by another IAC SE <cmo> IAC SE command from the Serving host. Break characters therefore define printing units. "Break character" as used in this document does NOT mean Telnet Break character.
Corrections to RFC 560: Remote Controlled Transmission & Echoing

TELNET Option

to:

A Break character REQUIRES that the using host transmit all text accumulated up to and including its occurrence and also causes the using host to stop its print/discard action upon the User's input text, until directed to do otherwise by another IAC 58 RCTL <cmd> IAC 58 command from the Serving host. Break characters therefore define printing units. "Break character" as used in this document does NOT mean Telnet Break character.

4. Change 5.B.6 (10492,2e2f)

from:

Input from the terminal is (hopefully) buffered up to the occurrence of a Transmission or Break character; and the input text is echoed or not echoed, up to the occurrence of a Break Character. The most recent RCTL command determines the echo, Transmission and Break actions.

to:

Input from the terminal is (hopefully) buffered into units ending with a Transmission or Break character; and echoing of input text is suspended after the occurrence of a Break Character and until receipt of a Break Reset command from the Serving host. The most recent RCTL Break reset command determines the Break actions.

5. Change 5.C.4 (10492,2e3d)

FROM;

A severe (User) site-dependent problem will be buffering type-ahead input from the terminal. It is possible, especially in the case of TIPS, that the input buffer will overflow often. If the receiving (serving) host will permit, the accumulated text should be transmitted at this point. If the text cannot be transmitted and further typing by the user will result in lost text, the user should be notified.

to:

Buffering Problems and Transmission vs. Printing Constraints:
There are NO mandatory transmission constraints. The Using host is allowed to send a character a time, though this would be a waste of RTTE. The Transmission Classes commands are GUIDELINES, so deviating from them, as when the User's buffer gets full, is allowed.

Additionally, the Using host may send a Break Class character, without knowing that it is one (as with type-ahead).

The problem with buffering occurs when printing on the user's terminal must be suspended, after the user has typed a currently valid Break Character and until a Break Reset command is received from the serving host. During this time, the user may be typing merrily along. The text being typed may be SAVED, but may not yet be PRINTED.

The more standard problem of filling the transmission buffer, while awaiting an ALLOC from the Serving host, may also occur, but this problem is well known to implementors and in no way special to RTTE.

In any case, when the buffer does fill and further text typed by the user will be lost, the user should be notified.

6. And add 5.0.5, 5.0.6, 5.0.7, 5.0.8, and 5.0.9 as follows: 

(5) The Serving and Using hosts must carefully synchronize Break Class Reset commands with the transmission of break characters. Except at the beginning of an interaction, the Serving host MAY ONLY send a Break Reset command in response to the User host's having sent a Break character as defined at that time. This should establish a one-to-one correspondence between them. (A <cmd> value of zero, in this context, is interpreted as a Break Classes reset to the same class(es) as before.) The Reset command may be preceded by terminal output.

(6) Text should be buffered by the User host until the user types a character which belongs to the transmission class in force at THE MOMENT THE CHARACTER IS TYPED.

(7) Transmission Class Reset commands may be sent by the Serving host at ANY TIME. If they are frequently sent separate from Break Class Reset commands, it will probably be better to exit from RTTE and enter regular character at a time transmission.
(8) It is not immediately clear what the Using host should do with currently buffered text, when a Transmission Classes Reset command is received. The buffering is according to the previous Transmission Classes scheme.

The Using host clearly should NOT simply wait until a Transmission character (according to the new scheme) is typed.

Either the buffered text should be rescanned, under the new scheme;

Or the buffered text should simply be sent as a group. This is the simpler approach, and probably quite adequate.

(9) It is possible to define NO BREAK CHARACTERS except Telnet commands (IAC ...). This might actually be useful, as in the case of transmitting on carriage-return, with the Using host echoing (a controlled half-duplex).

Having the using host send a Telnet Command will allow the serving host to know when he may reset the Break classes, but the mechanism is awkward and probably should be avoided.