BULLPRO

A Simple Bulletin Distribution Protocol

Tom Clark, W3IWI
6388 Guilford Road
Clarkmille, MD 21029

1. INTRODUCTION: This paper proposes a simple protocol for the efficient local distribution of bulletins by packet radio.

One of the interesting changes that packet radio has brought to the amateur community is the rapid national and international distribution of information for the entire amateur radio community. The introduction of the BID (Bulletin Identification Designator) by WA7MBL and its subsequent implementation by all the major BBS software writers has made it possible for a bulletin posted on a PBBS in one area to wend its way to hundreds of other PBBSes in a day or two. This capability has been used to provide wide and rapid circulation of AMSAT and ARRL bulletins, DX news, hamfest announcements as well as providing for a national "soapbox" for individuals. This paper will address neither the sociological implications of this capability nor problems associated misaddressed bulletins, duplicates and corrupted BIDs.

The intent of this contribution is to propose a more efficient distribution scheme for the local user in his local area network (LAN). Presently the local user gets his personal copy of each bulletin by logging onto his local PBBS, scanning the topics of interest and then reading particular bulletins. Other users in the area repeat the process, and a particular bulletin may be read many times.

The act of fetching bulletins is left to the end user who must log into his local BBS and manually initiate the read request for each bulletin he wants to read. Joe Kasser's (G3ZCZ/W3) *LAN-LINK* software frees the user from this manual operation by logging onto area PBBSes automatically to fetch personal mail and bulletins. Other individuals set up personal PBBSes (sometimes called Personal Mailboxes) so that the material is forwarded automatically. Whether the end user read the bulletins manually or lets his computer do the work, bulletins are transmitted repeatedly for each user; such activity consumes a large fraction of the available Local Area Network (LAN) channel time.

2. BROADCAST DATAGRAM PROTOCOLS (BDP): The obvious solution to this wasteful use of LAN resources is for the bulletin to be broadcast to many users at the same time. Conceptually, one LAN "superstation" sends the bulletins as unconnected packet frames (called <UI> frames or datagmms) and other stations in the LAN monitor the < UI > frames. In the AX.25 protocol < UI > frames are addressed from the originating station to a specific address (the UNPROTO address for most TNCs) and they are CRC checked for validity. Since they are broadcast to many users, individual recipients do not <ack> the individual datagmms.

By definition, <UI> datagmms are unnumbered and have no implicit sequencing. Any BDP must have added sequencing information for proper reassembly of the bulletin. In addition it is necessary to transmit some additional information so that the receiving station knows when the bulletin has been received in entirety.

3. BULLPRO -- A SPECIFIC BDP: In designing BULLPRO I had several objectives:

- The protocol should be simple enough so that a minimal user with only a TNC and line printer could make use of it.

- If an end-user does have a computer, then he should be able to use simple utilities he already has (for simplicity this document assumes a PC-clone running MS-DOS and some capture-to-disk terminal program).

  The BULLPRO protocol could easily be expanded so that "smart" software could handle tasks like eliminating duplicates, automatically filing the bulletins and even requesting "fills" of missing information.

To make BULLPRO work, the transmitted bulletins must meet the following criteria:

A. Bulletins are line-oriented with each line terminated by a < cr > . The TNC used to send the bulletins is set up with PACLEN at least 20 greater than the longest line. The TNC operates in CONVERSE mode with < cr > used as the SENDPAC trigger to send a frame. Thus each line of text corresponds to a separate < UI > frame.

B. Bulletins are N lines long and N < 1000.

C. Bulletins are identified by a unique character string, assumed to be a standard PBBS BID (the BID is l-12 alphanumeric characters, upper case only. Blanks, the punctuation characters @ < $ > and control characters are reserved and should not be used. The BID string is then prepended with a $ identifier.).

Consider the following N= 13 line long ARRL Bulletin which was distributed with the packet BID $ARLB026:

```
QST DE W1AW
ARRL BULLETIN 26 ARLB026
FROM ARRL HEADQUARTERS
NEUINGTON CT JULY 30, 1990
TO ALL RADIO AMATEURS

ON JULY 27 THE ARRL FILED ITS LEGAL BRIEF IN THE MATTER OF THE
PROPOSED REALLOCATION OF 220 TO 222 MHZ TO PRIVATE LAND MOBILE
SERVICES, FCC PR DOCKET 89-552, WITH THE UNITED STATES COURT OF
APPEALS FOR THE DISTRICT OF COLUMBIA CIRCUIT. ORAL ARGUMENTS BEFORE
THE COURT ARE SCHEDULED FOR NOVEMBER 16, 1990. THE CASE IS KNOWN AS
AMERICAN RADIO RELAY LEAGUE VERSUS FEDERAL COMMUNICATIONS COMMISSION
AND THE UNITED STATES OF AMERICA.
```

The bulletin distribution station using BULLPRO would send the bulletin by sending a beacon header with the following TNC commands:

```
MYCALL W3OBS-7                (as appropriate)
UNPROTO BULLTN VIA K9DOG
BEACON EVERY 60
```

and then identifying the bulletin currently being transmitted by sending an additional line of text (Line#00) which conveys the necessary descriptive material. This is done by setting setting the broadcast bulletin BTEXT to:

```
$ARLB026 L:00/13  SB ALL 3 ARRL  < W3OBS  900803  220 MHZ BRIEF FILED
```

where at least one blank separates the fields. The L:00/13 identifies this as line zero with 13 more lines to follow (a total of 14 lines). If a bulletin is more than 100 lines long, the L: length field would be like L:000/234. The additional information in Line #00 gives the user the same information he would have copied from his local PBBS and is adequate to re-introduce the bulletin into the packet system. The BEACON EVERY xx time interval should be chosen so that the Line #00 identification beacon is sent several times while the bulletin is being transmitted by BULLPRO.

The W3OBS-7 bulletin server's software then meters out bulletin text at a rate of one line (one frame) every 10-20 seconds (as appropriate to local conditions) and prepends sequencing information at the start of each line. The pattern of blanks in the prepended information should match that in Line #00 so that the most elementary character-oriented sort utilities (like MS-DOS SORT) can reassemble the bulletin.

```
$ARLB026 L:01 QST D E  W1AW
$ARLB026 L:02 ARRL BULLETIN 26 ARLB026
SARLB026 L:03 FROM ARRL HEADQUARTERS
$ARLB026 L:04 NEUINGTON CT JULY 30, 1990
$ARLB026 L:05 TO ALL RADIO AMATEURS
$ARLB026 L:06
SARLB026 L:07 ON JULY 27 THE ARRL FILED ITS LEGAL BRIEF IN THE MATTER OF THE
$ARLB026L:08 PROPOSED REALLOCATION (etcetera)
```

4. USER IMPLEMENTATION: At the user end, the minimal user can at least read the text and manually re-sequence it without ever logging onto the local PBBS.

A more sophisticated user could leave disk capture on overnight and save everything sent by W3OBS-7. An off-the-shelf utility like MSDOS's SORT could then be used to collect all lines with the BID $RLB026 together in order, and the FIND utility could be used to extract each bulletin into a separate file.

Assuming that bulletins are retransmitted several times, the user would be responsible for handling duplicated lines. The next step in sophistication would be to develop software to do this, to strip off the prepended Sequence information, to maintain a list with the status of receipt of different BIDs, and to file the bulletins based on their BID or other Line#00 criteria.

If copying the BULLPRO broadcast bulletins proves unreliable, then an additional feature could be added. The bulletin servers could listen for <UI> datagrams which request a specific line to be re-sent, something like:

```
K9DOG>REQBUL:??? $ARLB026 L:05
```

If a LAN has multiple BULLPRO servers, the user has the option of either selecting one with the MTO/MFROM or BUDLIST/LCALLS options in his TNC, or of accepting the multiple inputs and sorting out the duplicates in user's software. The latter option requires that all the servers adhere to a common BID standard. The uniform BID requirement is no more stringent than that imposed by the PBBSes to eliminate duplicates now.