IEN-67

TCP Meeting Notes

TCP Meeting Notes
30 & 31 January 1978

Agenda

Monday, 30 January
  8:30- 8:40   Introduction and Objectives - Cerf
  8:40- 8:45   Arrangements - Cohen
  8:45-10:30   Status Reports
    1. BBN TENEX & TOPS-20 status.
    2. BBN UNIX TCP (and FTP/TELNET)
    3. UCLA 360 TCP (and FTP/TELNET)
    4. MIT Multics TCP
    5. SRI LSI-11 TCP
    6. CCA RSX-11M TCP
 10:30-11:00   TCP-3 Comments Review - Postel
    A list of issues brought up by the review of the TCP-3 Draft
    Specification is to be circulated by Postel before the meeting. A
    working party will be assigned to sort out final editing on
    Tuesday.
 11:00-11:30   Fragmentation Issues and Choices - Cerf
    Object here is to get fragmentation issues out on the table and
    assign a small working group to resolve them on Tuesday.
 11:30-12:00   Sequencing -- Octets vs. Segments & Rubber EOS - Cerf
    Object here is to get issues these out on the table and assign a
    small working group to resolve them on Tuesday.
 12:00- 1:00   Lunch
  1:00- 1:30   CCA PSMF & TCP - Kou-Mei Chuang
    Packet Speech Measurement Facility/TCP Measurement
  1:30- 2:30   NSW Protocols and their Requirements - Thomas
    The object here is to understand how TCP might have to change to
    support the NSW MSG protocols.
  2:30- 3:30   Multidestination Addressing, Broadcast, Datagrams, and
Emmission Control Mode - Cerf
    Discuss the multidestination and broadcast topic, set up a working
    group on it, and introduce the issue of incorporating a datagram
    mode and an emmission control mode of operation into TCP.
  3:30- 4:00   Types of Services Revisited - Cohen
    Can we get a specific proposal for the first cut set of services
    and set up a working party to worry about how they are coded,
    characterized, etc. Includes discussion of Plummer's ideas and
    Craighill's ideas.
  4:00- 4:30   Real Time Conferencing - Cohen
  4:30- 5:00   TCP Demo - Plummer

Tuesday, 31 January
 8:30-12:00   Working Groups
     (a) TCP 3 specification editing
     (b) Fragmentation
     (c) Multidestination/broadcast
     (d) Speech/real-time
     (e) Sequence Counting and Rubber Everythings
12:00- 1:00   Lunch
 1:00- 2:30   Reports from Working Groups
 2:30- 3:30   Specific Plans for 1978 - Cerf + inputs from others
     Experiments, milestones, demonstrations, developments, analysis,
     etc.
 3:30- 4:00   Agenda for Next Meeting - Cerf

Introduction and Objectives - Cerf

The main objective is to get TCP-3 straightened out, and to discuss
extensions.

Arrangements - Cohen

The arrangements are: see Debe.

Uses this time to complain about TCP-3 becomming all things to all
people. Also illustration of the approach to voice service via an
"unreliability" package on top of TCP.

Status Reports

    1. BBN TENEX & TOPS-20 status.

       Bill Plummer reports that both Tenex and Tops20 TCPs are running.
       These are most similat to TCP 2. (For example, Urgent and Rubber
       EOL are not implemented.) Both are monitor versions (not user
       code). Some documentation has been done.

    2. BBN UNIX TCP

       Ray Tomlinson reports that the BBN Unix implementation is
       essentially Jim Mathis' TCP11 encapsulated into the Unix
       environment, with user to TCP communication via the RAND port
       mechanism.

    3. UCLA 360 TCP

       Bob Braden reports that work is in progress, right now focusing on
       cleaning up the network interface environment, still planning on a
       1-June date to begin testing TCP-3 with other sites.

Postel

4. MIT Multics TCP

   Dave Reed reports that TCP-3 for Multics is now being coded but
   that testing of the revised 1822 interface has higher priority.
   Dave expects that Multics may be ready to test its TCP-3 against
   other sites as soon as a month from now.

5. SRI LSI-11 TCP

   Jim Mathis reports that the running version of TCP in his MOS and
   ELF systems is version 2.5, and has been distributed to several
   other sites.

6. CCA RSX-11M TCP

   Kou-Mei Chuang reports that CCA is building a packet speech
   measurement facility which is to include a TCP measurement
   facility. CCA plans to use Mathis TCP11 on a RSX system.

7. DTI Unix+ TCP

   Gary Grossman reports that DTI is working on a TCP-3
   implementation funded by a DCEC contract.

TCP-3 Comments Review - Postel

   Jon reviews the main points on the list of issues circulated before
   the meeting. A brief discussion of the European use of the interrupt
   facility and the relation of that to TCPs urgent mechanism took
   place. A working party is to sort out final editing on Tuesday.

Fragmentation Issues and Choices - Cerf

   Vint proposes that fragmentation be removed from TCP and be
   designated an internet protocol task. John Shoch presented a brief
   summary of his memo on this topic. A working group is to resolve this
   issue on Tuesday.

Flow Control - Cerf

   Some discussion of sequencing of octets vs. sequencing of segments.
   Arguments about the unit of work of a TCP being a segment. Other
   arguments about when a TCP is required to send ACKs. A suggestion
   that there be a per connection segment size negotiated at connection
   set up. Talk of allocations of segment counts and octet counts
   (shades of NCP !). Result of all this ==> DO NOTHING (thank
   goodness)!

Protocol Specification Techniques - Sunshine

Carl present some ideas on how to make better specifications of protocols. There were 5 catagories: Introduction, Design Goals, Pure Specification, Implementation Suggestions, and Analysis of Specification. These five catagories are to be applied to the protocol itself and to each of its interfaces. Others suggested that there should be a functionality specification, an algorithmic specifcation, and an interface specification. A users guide was mentioned. There is to be a working group on this too.

CCA PSMF & TCP - Kou-Mei Chuang

Packet Speech Measurement Facility/TCP Measurement

Kou-Mei made a presentation about the CCA Packet Speech Measurement Facility and showed some of the analysis that can be performed. The facility can record streans of speech packets and play them back, a pass through feature is now being added.

NSW Protocols and their Requirements - Thomas

Bob gave a presentation of the MSG protocol and especially its role in the NSW. There was some discussion of how TCP could help and how MSG could use a datagram mode.

Multidestination Addressing, Broadcast, Datagrams, and Emmission Control Mode - Cerf

Vint led a general discussion of these topics which turned up a lot of questions and few answers.

Types of Services Revisited - Cohen

Earl Craighill made a brief presentation of the material in his note. Danny made a presentation of the type of information he sees needed on the outside of the envelope and on the inside of the envelope. This led to a discussion of the dependcies between protocol function, which led to having a working group on the topic on Tuesday.

TCP Demo - Plummer

Bill ran a demo of TCPs on various Tenexs and Top20s talking with each other.

Working Groups

    (a) Protocol Specification Techniques
    (b) Fragmentation
    (c) Type of Service Dependices
    (d) TCPs for PDP-11s
    (e) TCP 3 Document Editing

Working groups a, d, and e met in parallel sessions for about an hour
and a half, then groups b and c met in parallel until lunch time.
After lunch the fragmentation discussion continued briefly, with the
summary presented by Ray Tomlinson.

Reports from Working Groups

    (a) Protocol Specification Techniques -- Sunshine

        The following outline for improved TCP documentation was developed
        by a working group at the Jan 30-31 1978 TCP meeting. It suggests
        dividing the current single document (Version 3.0 spec) into five
        sections: a brief introduction; more detailed philosophy,
        explanation, and justification of TCP design; a formal
        specification of the TCP-TCP mechanisms and the functional
        interfaces to users (above) and network or internet (below); a
        rigorous verification; and implementation suggestions based on the
        experience of the several TCP implementations to date.

        The introduction, philosophy, and implementation sections would be
        based largely on material already in the Version 3.0 spec.
        Techniques for formal specification appear to require further
        study, although section 4.2.9 of the current spec again provides a
        basis. Verification techniques are definitely a research topic.
        Carl Sunshine will be heading a continuing working group on these
        topics, and any comments on the suggested outline are welcome.

        OUTLINE FOR IMPROVED TCP DOCUMENTATION

        1) Introduction (about 10 pages)

            Brief history, context
            Scope, purposes and goals of TCP
            Mention rest of documentation
            Emphasize 3 levels (User-TCP, TCP-TCP, TCP-net) Brief
            operation of protocol-philosophy
            Glossary

        2) Philosophy:

            Detailed history - lessons learned

Compare with others (NCP, INWG 96)
Explain mechanisms, intuitive description of
3-way handshake, addressing, termination,
no Resynch, urgent, letters, etc.
Functional spec of User-TCP, TCP-net interfaces Problems,
shortcomings
Future Directions
Examples - scenarios
Glossary

3) Formal Specification

Explain formalisms used
Discuss other techniques
Potential techniques:
Formal Language: BNF, W-grammars
State Transitions (Events-->Actions)
Program Description
Processing of Events in prose
Module Definition
Illinois Event spec.
Requires further research

4) Rigorous Verification

May require different formal spec. techniques
Based on combination of state reachability and
assertion proofs
Requires further research

5) Implementation Hints

What not to leave out (see functional interface spec) User
interfaces
TCP mechanisms
Data structures
Program sizes, performance
Test sequences, procedures, exerciser
Parameter values: timeouts, segment sizes, quit time, ACK
times, buffering strategies, windows
Debugging
Separate general from specific

(b) Fragmentation -- Cerf

This group had a lengthly discussion. One immeadiate result was
the decision to remove fragmentation from TCP and place it in the
internet protocol, subsequent discussion focused on how to do
fragmentation. This discussion resulted in the definition of a

series of numbers and relations between the numbers regarding segment, fragment, and packet sizes.

M0 := a segment of this size or smaller will not need to be fragmented by any network in the internet system. That is, all networks will handle segments of this size (and smaller) as complete units.

M1 := all internet fragment reassemblers will handle segments of this size.

M2 := the largest segment a particular internet module is able to receive from the network, independent of reassembly. This a local implementation parameter.

M3 := the largest segment a particular higher level module (e.g. TCP) is able to accept from an internet module. This a local implementation parameter.

M4 := the largest segment that can be handled by the internet system.

MF := the unit of fragmentation.

and

$$MF <= M0 <= M1 <= M2 <= M3$$

$$M2 <= M4$$

M0 = 126 octets (1008 bits).
M1 = to be determined.
M2 = local option.
M3 = local option.
M4 = 2^{14} octets (2^{17} bits).
MF = 64 octets (512 bits).

To aid in the reassembly of fragments each segment will carry an identifier (16 bits). A reassembler checks the four fields identifier, source, destination, and format to decide if two fragments are from the same segment. The value of the identifier is set by the sending higher level protocol. To indicate where in a segment a fragment belongs there is a fragment offset field (8 bits) that specifies the location of the fragment as measured in MF units from the begining of the segment. There is a "not-last-fragment" bit that is set on all but the last fragment. Finally there is a "don't-fragment" bit that indicates a segment with this bit set is not to be fragmented in any case.

It should be noted that all of the above is with respect to
internet fragmentation, that is, fragmentation visible at the
internet level. It is permitted that networks or pairs of
gateways use intranet fragmentation as long as the operation is
invisible at the internet level.

(c) Type of Service Dependices -- Grossman

A set of potentially optional facilities in the present TCP were
identified. Each of these facilities may be present or absent on
an individual connection basis.

Octet Ids: A facility which provides a distinct id for each octet
              of data.

Sequence Numbers: Octet ids which in addition indicate the order
              of the data octets.

Reject Duplicates: A facility to detect and discard duplicate
              data octets. Uses octet ids.

Ack-window: An acknowledgment which defines the left window edge.

Sort: A facility that restores the relative order of the data
         octets, but does not fill holes in the data stream.

Fill: A facility that fills holes in the data stream.

Flow Control: A facility that allows the receiver to control the
               flow of data.

Ack-delivery: The acknowledgment by the receiver of all data
           octets up to a given point in the data stream.

Retransmit: A facility used by data senders to ensure arrival of
           data at the receiver.

A dependency matrix and graph were developed to show the
dependency relations between these facilities. It seems that
there are some facilities which are independent of most other
facilities.

TCP Dependency Matrix

```
                              R                 S
                              e                 e
                              j                 q
                              e                 u
                      A       c       F         e
                      c       t       I         n
                      k   A R D       o   O     c
                      c   c e u       w   c     e
                      D   k t p           c
                      e   r l i       C N t
                      l   W a i       o u e
                      i   i n c       n m b
                      v   n s a S F   t o i r e l
                      e   d m t o i   r i r e l
                      r   o i e r l   o r l o r d
                      y   w t s l l   l s l s s

                   ---------------------------------------
Ack-delivery       !\! !2!2!A!X! !X! !\!
                   ---------------------------------------
Ack-window         ! !\! ! ! ! ! ! !X!
                   ---------------------------------------
Retransmit         !B!B!\!3!2!2! ! !X!
                   ---------------------------------------
Reject Duplicates  ! ! !\! ! ! ! ! !X!
                   ---------------------------------------
Sort               ! ! ! ! !\! ! ! !X!
                   ---------------------------------------
Fill               !2!2!X!X!X!X!\! !X!
                   ---------------------------------------
Flow Control       ! !X! ! ! ! ! !\!X! !
                   ---------------------------------------
Sequence Numbers   ! ! ! ! ! ! ! !\!X!
                   ---------------------------------------
```

Rows depend on Columns.

Notes:
    A - need not be delivered to the user in order.
    B - One of Ack-delivery or Ack-window is reqired for
    termination.
    2,3 - 2nd and 3rd order dependency.

A facility was proposed for synchronizing activity between
multiple data streams. This facility was relegated to a higher
level protocol.

Three basic questions were generated:

1.   To what extent are TCP mechanisms independent enough to permit reduced TCP implementations to provide reliable service when used on a relatively reliable transmission media?

E.G., can a TCP using flow control alone provide reliable service on a virtual circuit?

2.   To what extent will reduced service version s of TCP work, provided that errors are handled by RSTs?

E.G., can partial implementation TCPs work at all?

3.   What additions to or modifications of TCP mechanisms would improve the independence and/or performance of the existing TCP mechanisms?

E.G., when TCP is used on a transmission medium which delivers data in order but may lose data, a selective retransmission request might improve performance.

(d) TCPs for PDP-11s -- Mathis

Garry Grossman from DTI briefly presented their UNIX implementation of TCP v3, scheduled for completion in June. The TCP is built upon the Illinios UNIX InterProcess Communication facility. This facility provides: Events - a "small" message exchanged between processes Segments - data that is shared between address spaces by using virtual memory mapping facility.
The TCP will be interface with the standad UNIX I/O system so that existing programs can be made to use TCP connections with minimal changes.

List of operating systems which provide (or will provide) TCP support:

    MOS - SRI (v2.5)
    ELF - BBN (v2.5)
    UNIX - BBN (v2.5)
    UNIX - DTI (v3, June completion)
    RSX-11M - CCA (mid-1978 completion)

The current v2.5 TCP11 program is expected to be upgraded to v3 protocol by May.

The remainer of the session was used for discussions on converting TCP11 to run on various system configuartions.

(e) TCP 3 Document Editing -- Postel

It was decided to produce a slightly revised version of the document correcting the minor points, then to decide whether or not to go ahead with a major revision according to the results of the protocol specification working group.

Among the points to be covered in the minor revision are: reset (to be reviewed by Carl Sunshine and Jon Postel), fragmentation (see result of fragmentation working group), internet format (perhaps as a separate document), and the checksum procedure (with input from Dave Reed).

Specific Plans for 1978 - Cerf

Vint promises to circulated a note covering this topic.

Agenda for Next Meeting - Cerf

The next meeting will be 1&2 June 1978 at MIT.

Progress Reports
    BBN - Tops20
    BBN - Tenex
    BBN - Unix
    NDRE - NORD10
    UCLA - 360/91
    SRI - TCP11 MOS & ELF
    CCA - RSX-11
    MIT - Multics
    DTI - Unix+
    UCL - TCP in various machines
Summary of the Internet Protocol -- Cerf
TCP-3 Specification Issues -- Postel
TCP Facility Dependencies -- Grossman
Real-Time Protocol -- Cohen
Internet Type of Service -- Cohen
    Proposal for actual services.
Broadcast/Multidestination -- Cerf
    Description of the service desired, description of applications
    that would use it.
MSG - Response to the Internet Service Proposed
Protocol Specification -- Sunshine
Working Groups
    Protocol Specification
    Internet Protocol
    Real Time Protocol
    TCP Issues

Postel

Memos Distributed

  1) Transmission Control Procedures -- Postel
  2) A Quick Approach for TCP Type of Service -- Craighill
  3) Protocol Topics -- Postel
  4) Comments on TCP-3 (Jan-78) -- Cohen
  5) On Names, Addresses, and Routings [IEN 23] -- Cohen
  6) Design Considerations for a Real-time Text Conf. System -- Cohen
  7) Inter-Network Naming, Addressing, and Routing [IEN 19] -- Shoch
  8) Inter-Network Fragmentation and the TCP [IEN 20] -- Shoch
  9) More on Internet Fragmentation -- Shoch
 10) Assigned Numbers [RFC 739] -- Postel

Attendees

NAME	AFFILIATION	ADDRESS	TELEPHONE
Vint Cerf	ARPA	CERF@ISI	(202)694-3049
Bill Plummer	BBN	PLUMMER@BBN	(617)491-1850
Virginia Strazisar	BBN	STRAZISAR@BBN	(617)491-1850 ext. 632
Ray Tomlinson	BBN	TOMLINSON@BBN	(617)491-1850
Kou-Mei Chuang	CCA	KOU-MEI@CCA	(617)491-3670
Ed Cain	DCA	ECAIN@BBN	(703)437-2247
Ray McFarland	DOD	DEDWARDS@ISI	(301)796-6290
Gary R. Grossman	DTI	GRG@ILL-CAC	(217)384-8500
Danny Cohen	ISI	COHEN@ISI	(213)822-1511
Jon Postel	ISI	POSTEL@ISIB	(213)822-1511
David Reed	MIT-LCS	DPR@MIT-ML	(617)253-6004
Paal Spilling	NDRE	PAAL@SRI-KA	Norway: (02)712660 (229)
Carl Sunshine	RAND	CAS@RAND-UNIX	(213)393-0411
David Kaufman	SDC	WEISSMAN@ISI	(213)829-7511 ext. 2627
Earl Craighill	SRI	CRAIGHILL@SRI	(415)326-6200
Jim Mathis	SRI	MATHIS@SRI-KL	(415)326-6200 ext. 5150
Sunil Das	UCL	KIRSTEIN@ISI	UK: (01)387-7050
Peter Kirstein	UCL	KIRSTEIN@ISI	UK: (01)387-7050
Bob Braden	UCLA	BRADEN@CCN	(213)825-7518
David Boggs	XEROX-PARC	BOGGS@PARC-MAXC	(415)494-4365
Yogen Dalal	XEROX-PARC	DALAL@PARC	(415)494-4259
John Shoch	XEROX-PARC	SHOCH@PARC-MAXC	(415)494-4384