Network Working Group
Request for Comments: 40

E. Harslem
J. Heafner
RAND
March 1970

# More Comments on the Forthcoming Protocol

We have recently discussed NWG/RFC Nos. 36 and 39 with Steve Crocker, UCLA. Steve has asked that we elaborate on the errors, queries, and HOST status that were mentioned in NWG/RFC #39.

Please voice your opinions soon in order to affect the forthcoming protocol specifications.

## ERROR MESSAGES

<ERR> <Code> <Command length> <Command in error>

<Code> is an eight-bit field that specifies the error type. The assigned codes are shown below. <Command length> is a 16-bit integer that indicates the length of the <Command in error> in bits. The <Command in error> is the spurious command.

The ranges of <Code> are shown below in hexidecimal.

| | |
|---|---|
| 00 | Unspecified error types |
| 10-0F | Resource errors |
| 10-1F | Status errors |
| 20-2F | Content errors |
| 30-3F | Unused |

Specific values of <Code> are shown below with their meaning.

| <Code> value | Semantics |
|---|---|
| 00 | Unspecified errors. |
| 01 | Request for an invalid resource. |
| 02 | Request for an exhausted resource, try later. |
| 03-0F | Unused. |
| 10 | Invalid <RSM>, i.e., link connected but unblocked. |
| 11 | Invalid <SPD>. |
| 12 | Invalid <ASG>, i.e., connected but no <RDY> received. |

| <Code> value | Semantics |
|---|---|
| 13 | Message received on blocked link. |
| 14-1F | Unused. |
| 20 | Unknown command code. |
| 21 | Message received on unconnected link. |
| 22 | Invalid <RFC>. |
| 23 | Invalid <CLS>. |
| 24 | Invalid <RSM>, i.e., link not connected. |
| 25 | Invalid <FND>. |
| 26 | Invalid <END>. |
| 27 | Invalid <RDY>. |
| 28 | Invalid <ASG>, i.e., not connected. |
| 29-2F | Unused. |
| 30-FF | Unused. |

QUERIES

```
     <QRY> <My Socket>
or   <RPY> <Your Socket> <Text>
```

The <QRY> is the query indicated in NWG/RFC #39 and <RPY> is the reply. The format of <Text> is shown below; also refer to NWG/RFC #36, p. 3.

```
<Text>::= <16 bit count of relevant connection table entries>
          <relevant connection table entries>

<relevant connection table entries>::=
                                      <relevant connection table entries>
                                      <a relevant connection table entry>
                                      <a relevant connection table entry>

<a relevant connection table entry>::= <local socket> <foreign socket>
                                       <link> <connection state>
                                       <flow state and buffer control>
                                       <reconnection control state>
```

HOST STATUS

<NOP>

An NCP may be up, down, pending, etc. When an NCP changes its state to UP it should send a <NOP> to each remote NCP which indicates the NCP is available. The sending NCP can then construct a vector of HOST status from the RFNMs it receives. An NCP receiving a <NOP> can update the availability of the sending NCP in its HOST status vector.