

NSCOM-01 MC01 Documentation

1. Message Format

a. WRQ / RRQ

```
+-----+-----+-----+-----+-----+-----+-----+-----+
| Opcode | Filename | 0 | Mode | 0 | Opt1 | 0 | Val1 | 0 | OptN | 0 | ValN |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- Opcode: Can be either be a value of 1 (RRQ) or 2 (WRQ)
- Filename: A string containing the filename to be uploaded or downloaded.
- Mode: Indicates the mode of file transfer. Possible values can be “netascii”, “octet”, and “mail”.
- Opt1/OptN: A string containing the name of the option to be added such as Blksize, Tsize, and Timeout.
- Val1/ValN: The value associated with the option.

The WRQ / RRQ packet is the first tftp packet to be sent by the client and the first to be received by the server. The standard format of this packet is opcode, filename, 0, mode, and another 0. The packet can be extended to accommodate optional fields such as block size, tsize, and timeout to further enhance the performance of the speed of the file transfer and provide additional information to the client and server program.

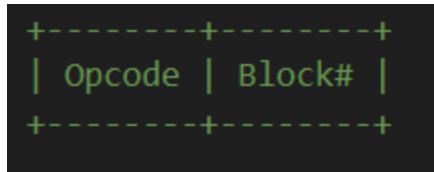
b. DATA

```
+-----+-----+-----+
| Opcode | Block# | Data |
+-----+-----+-----+
```

- Opcode: Contains value of 3.
- Block#: Contains the block number of the data to be transferred.
- Data: The data of the file stored in bytes.

The Data packet contains the data that is being uploaded or downloaded alongside the block number, to indicate what block of data is being transferred. The Data field has a default length of 512 bytes, but it can be adjusted by using the blksize option when sending the RRQ or WRQ packet.

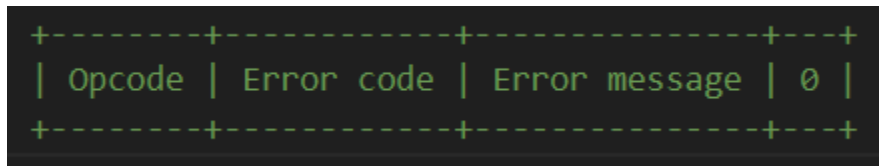
c. ACK



- Opcode: Contains value of 4.
- Block#: Contains the block number of the data to be acknowledged.

The ACK packet are given by the server or the client as an acknowledgement that the packet sent by the other side has been received and acknowledged. Similar to the Data packet, it also contains the block number to indicate what packet is being acknowledged.

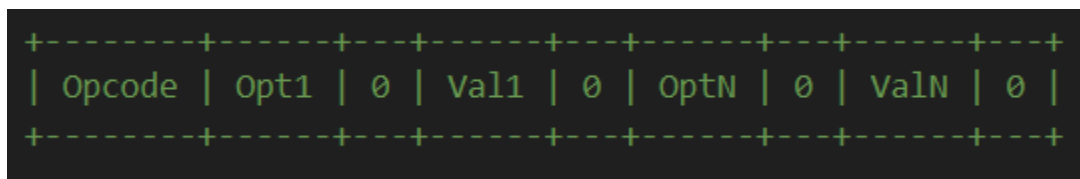
d. ERROR



- Opcode: Contains a value 5.
- Error code: Contains the error code which indicates the type of error that occurred.
- Error message: Contains detailed information about the error that occurred.

The Error packet is given by the server or client if a request by the other side cannot be granted or if some error occurred. It is stated by the RFC 1350 that the Error packet should not be acknowledged by an ACK packet. It may be therefore lost during the packet transmission process.

e. OACK

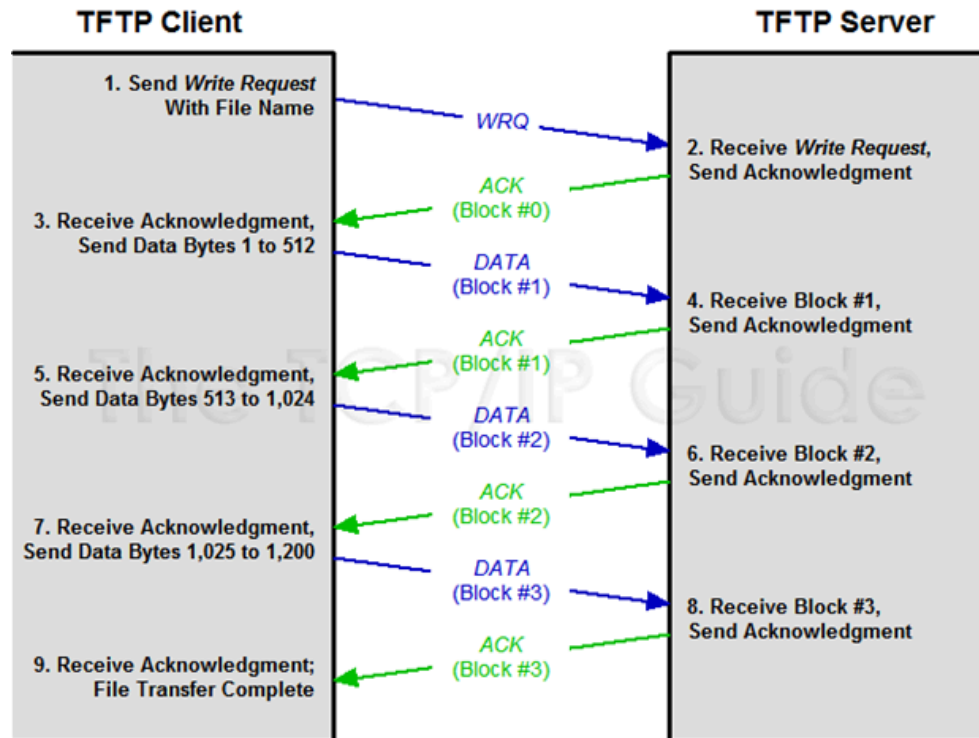


- Opcode: Contains a value of 6.
- Opt1/OptN: A string containing the option to be acknowledged by the server Blksize, Tsize, and Timeout.
- Val1/ValN: The value associated with the option.

The OACK packet is given as a reply by the server to the client's WRQ or RRQ request, if the client chose to include the optional fields along with their respective values. This packet means that the server acknowledges the client's negotiation for the optional fields.

2. Sequence Diagrams

a. TFTP RRQ/WRQ Diagram



The TFTP client program follows the RFC 1350 wherein the program will first send a TFTP packet that is either a RRQ or WRQ packet to the server. The server responds with an ACK or an OACK to the previous packet sent by the client. Depending on the first type of message sent by the client, it will start receiving Data packets and replying with ACK packets for RRQ. While for WRQ, it will start sending Data packets and receiving ACK packets. This process will continue to happen up until the client sends/receives the last packet from the server.

b. TFTP ERROR Diagram



In some cases, a TFTP client/server may send an ERROR packet, indicating that it has encountered an error during transmission. Because TFTP does not have any kind of error checking, all error handling must be done on the client-side. This includes resending the data/packet if it was ever damaged before or after transfer, displaying the error message on the client-side, or terminating the program altogether. If improper error handling is implemented, an ongoing TFTP transmission may encounter what's known as the Sorcerer's Apprentice bug. This bug causes the server-side to return the data twice, and will continue to do so until the program has timed out or if the transmission has terminated.



TFTP Connection (Cont'd)

