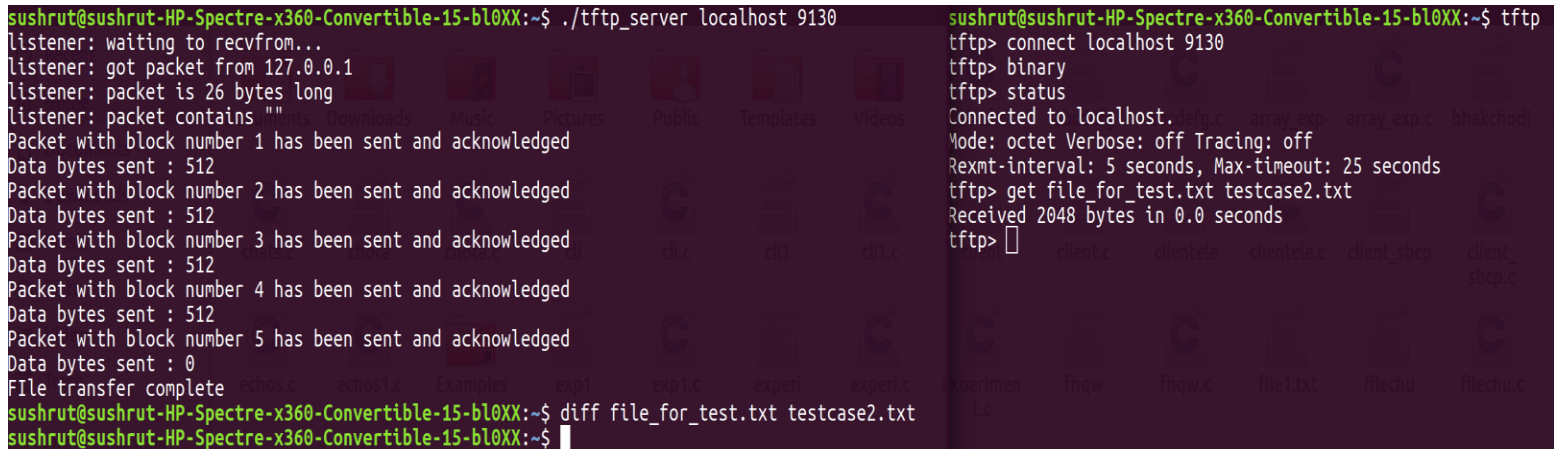


TEST CASES

TEST CASE-1

A). transfer a binary file of 2048 bytes and check that it matches the source file



```
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ ./tftp_server localhost 9130
listener: waiting to rcvfrom...
listener: got packet from 127.0.0.1
listener: packet is 26 bytes long
listener: packet contains ""
Packet with block number 1 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 2 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 3 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 4 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 5 has been sent and acknowledged
Data bytes sent : 0
File transfer complete
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ diff file_for_test.txt testcase2.txt
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$
```

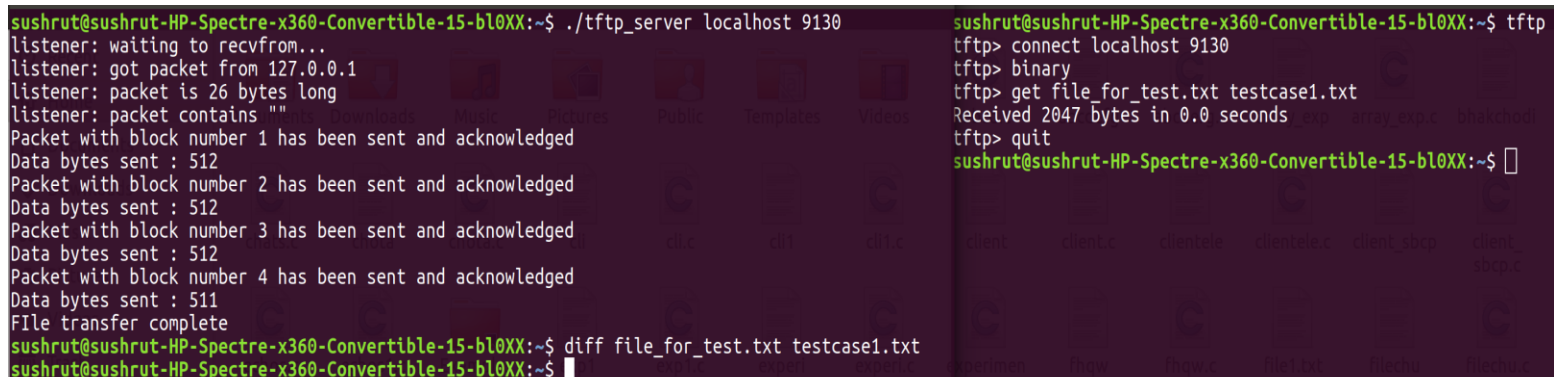
```
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ tftp
tftp> connect localhost 9130
tftp> binary
tftp> status
Connected to localhost.
Mode: octet Verbose: off Tracing: off
Rexmt-interval: 5 seconds, Max-timeout: 25 seconds
tftp> get file_for_test.txt testcase2.txt
Received 2048 bytes in 0.0 seconds
tftp>
```

For this test case we generated a binary file (file_for_test.txt) of size 2048 bytes. The server then implements reading that file using the RRQ. Afterwards, we copied the file and transferred the content onto a new file (testcase1.txt). Finally we have compared both the files for the content. The result showed that there was no difference between the files.

Also, worth noticing is that the file was sent in 4 packets (integral multiple of 512 bytes), which can be seen in the above screenshot.

TEST CASE-2

A). transfer a binary file of 2047 bytes and check that it matches the source file



```
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ ./tftp_server localhost 9130
listener: waiting to rcvfrom...
listener: got packet from 127.0.0.1
listener: packet is 26 bytes long
listener: packet contains ""
Packet with block number 1 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 2 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 3 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 4 has been sent and acknowledged
Data bytes sent : 511
File transfer complete
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ diff file_for_test.txt testcase1.txt
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$
```

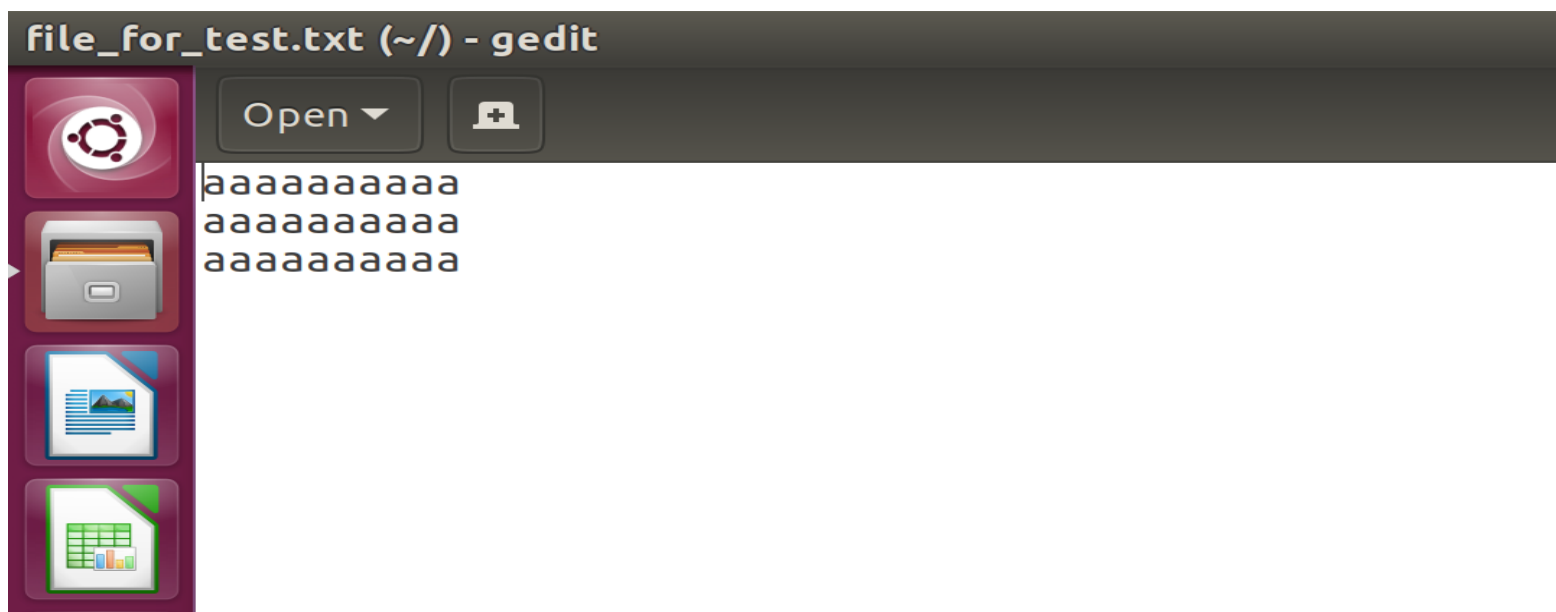
```
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ tftp
tftp> connect localhost 9130
tftp> binary
tftp> get file_for_test.txt testcase1.txt
Received 2047 bytes in 0.0 seconds
tftp> quit
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$
```

In this test case, we did everything quite similar to the first test case. The only difference is that the file size is not an integral multiple of 512 bytes (2047 bytes) this time. The bytes sent can be clearly seen in the screenshot of the command window on the left side (511 bytes).

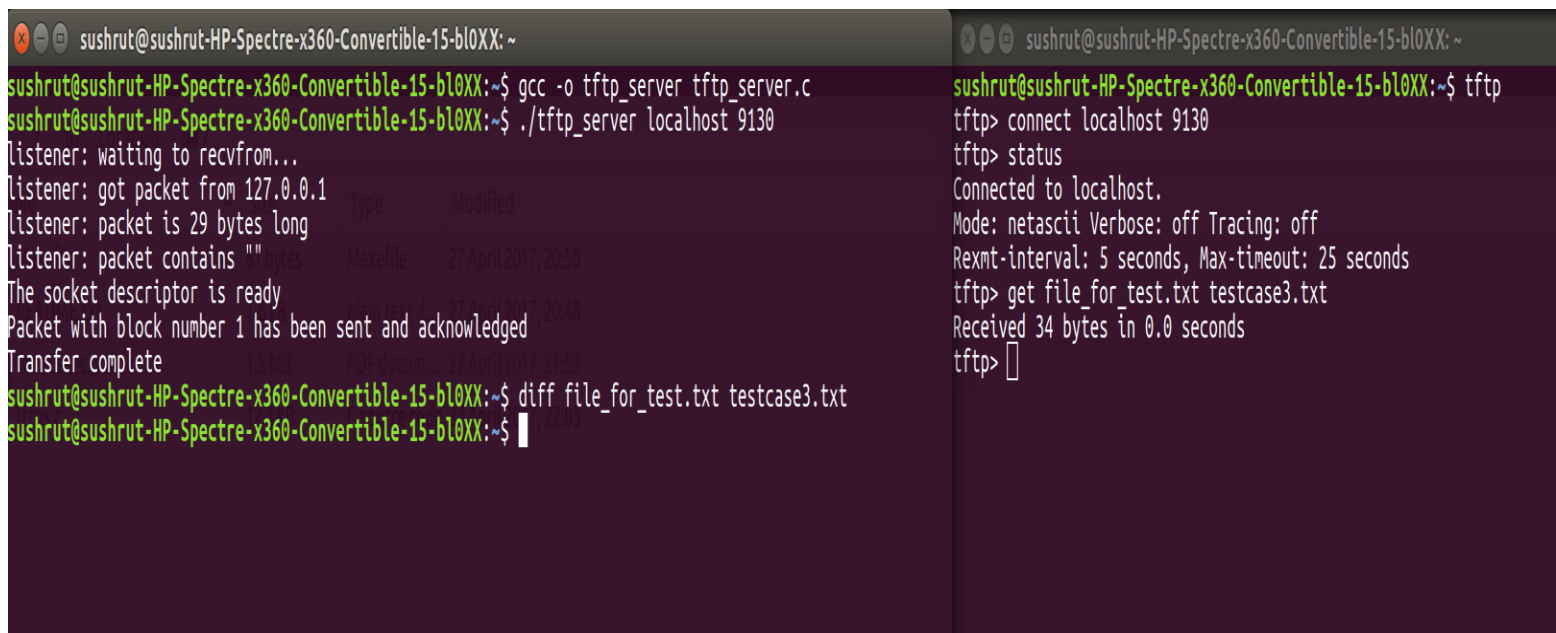
TEST CASE-3

A). transfer a netascii file that includes two CR's and check that the resulting file matches the input file

We created a file named file_for_test.txt with 2CR's and the server then implements reading that file using the RRQ and transferred the contents onto a file (testcase3.txt). For the creation of a file with 2CR's we wrote a code to introduce the required carriage returns. Below is a screenshot of the file that was read:



From the screenshots below it can be observed that the difference between the two files is turning out to be zero which proves that the read operation as executed successfully.




```
sushrut@sushrut-HP-Spectre-x360-Convertible-15-bl0XX: ~
Data bytes sent : 512
Packet with block number 65516 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65517 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65518 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65519 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65520 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65521 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65522 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65523 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65524 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65525 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65526 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65527 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65528 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65529 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65530 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65531 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65532 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65533 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65534 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 65535 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 0 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 1 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 2 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 3 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 4 has been sent and acknowledged
Data bytes sent : 512
Packet with block number 5 has been sent and acknowledged
Data bytes sent : 512

sushrut@sushrut-HP-Spectre-x360-Convertible-15-bl0XX: ~
sushrut@sushrut-HP-Spectre-x360-Convertible-15-bl0XX:~$ tftp
tftp> connect localhost 9130
tftp> binary
tftp> status
Connected to localhost.
Mode: octet Verbose: off Tracing: off
Rexmt-interval: 5 seconds, Max-timeout: 25 seconds
tftp> get file_for_test.txt testcase4b.txt
Received 34000000 bytes in 1.4 seconds
tftp>
```

For test case 4, we created a file of size 34 MB (file_for_test.txt). The rest of the operations are pretty similar to first two test cases. The only difference being the wrap-around function. This can be clearly seen in that that after the block number 65535 the program again starts with the packet number 0.

Above screenshots show scenarios for the case of NETASCII and OCTET.

TEST CASE-5

A). check that you receive an error message if you try to transfer a file that does not exist and that your server cleans up and the child process exits

```
sushrut@sushrut-HP-Spectre-x360-Convertible-15-bl0XX: ~
sushrut@sushrut-HP-Spectre-x360-Convertible-15-bl0XX:~$ ./tftp_server localhost 9130
listener: waiting to recvfrom...
listener: got packet from 127.0.0.1
listener: packet is 20 bytes long
listener: packet contains ""
[Error]! File already exists !!

sushrut@sushrut-HP-Spectre-x360-Convertible-15-bl0XX: ~/Downloads
sushrut@sushrut-HP-Spectre-x360-Convertible-15-bl0XX:~$ cd Downloads/
sushrut@sushrut-HP-Spectre-x360-Convertible-15-bl0XX:~/Downloads$ tftp
tftp> connect localhost 9130
tftp> put well.txt
Error code 6: File already exists
tftp>
```

Here we try to read a file (file_for_test.txt), which was not present in the server at that time. Which gives us an error message with error code1, which can be seen in the screenshots above.

TEST CASE-6

A). Connect to the TFTP server with three clients simultaneously and test that the transfers work correctly (you will probably need a big file to have them all running at the same time)


```
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX: ~
Packet with block number 856 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 857 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 858 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 859 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 860 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 861 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 862 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 863 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 864 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 865 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 866 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 867 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 868 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 869 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 870 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 871 has been sent and acknowledged
Transfer complete
]

sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX: ~
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ tftp
tftp> connect localhost 9130
tftp> get file_for_test.txt out1.txt
Received 34000000 bytes in 3.1 seconds
tftp> ]

sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX: ~
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ tftp
tftp> connect localhost 9130
tftp> get file_for_test.txt out2.txt
Received 34000000 bytes in 3.1 seconds
tftp> ]
```

TEST CASE-7

A). terminate the TFTP client in the middle of a transfer and see if your TFTP server recognizes after 10 timeouts that the client is no longer there (you will need a big file)

```
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX: ~
Packet with block number 11975 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11976 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11977 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11978 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11979 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11980 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11981 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11982 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11983 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11984 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11985 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11986 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11987 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11988 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11989 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11990 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11991 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11992 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11993 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11994 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11995 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11996 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11997 has been sent and acknowledged
The socket descriptor is ready
Packet with block number 11998 has been sent and acknowledged
Timeout has occurred
Timeout has occurred
Timeout has occurred
Timeout has occurred
Timeout has occurred
Timeout has occurred
Timeout has occurred
Timeout has occurred
Timeout has occurred
Timeout has occurred
The timeout counter has exceeded 10
]

sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX: ~
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ tftp
tftp> connect localhost 9130
tftp> status
Connected to localhost.
Mode: netascii Verbose: off Tracing: off
Rexmt-Interval: 5 seconds, Max-timeout: 25 seconds
tftp> get file_for_test.txt testcase7.txt
AC
tftp> ]
```

In this test case, we try to transfer a large file but abruptly kill our client process in the middle. Our server recognizes the absence of the client after 10 timeouts which can be clearly seen in the screenshot above.

TEST CASE-8

A). Bonus- test that you can send both netascii and binary files from your client to the server, and then transfer back the same files from the server to the client. Compare the original and the copied versions to make sure they are identical.

ASCII

```
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX: ~
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ gcc -o tftp_server tftp_server.c
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ ./tftp_server localhost 9130
listener: waiting to recvfrom...
listener: got packet from 127.0.0.1
listener: packet is 20 bytes long
listener: packet contains ""
received 512 bytes
It indeed sent a data packet
received 512 bytes
It indeed sent a data packet
received 181 bytes
It indeed sent a data packet
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$

sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX: ~/Downloads
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~/Downloads$ tftp
tftp> connect localhost 9130
tftp> status
Connected to localhost.
Mode: netascii Verbose: off Tracing: off
Rexnt-interval: 5 seconds, Max-timeout: 25 seconds
tftp> put well.txt
Sent 1205 bytes in 0.0 seconds
tftp> quit
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~/Downloads$
```

OCTET

```
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX: ~
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$ ./tftp_server localhost 9130
listener: waiting to recvfrom...
listener: got packet from 127.0.0.1
listener: packet is 17 bytes long
listener: packet contains ""
received 516 bytes
received 516 bytes
received 166 bytes
This is the last packet
Transfer complete
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~$

sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX: ~/Downloads
sushrut@sushrut-HP-Spectre-x360-Convertible-15-b10XX:~/Downloads$ tftp
tftp> connect localhost 9130
tftp> binary
tftp> put well.txt
Sent 1186 bytes in 0.0 seconds
tftp>
```

Here for this test case, we implemented the BONUS feature where server implements WRQ for both NETASCII & OCTET. It can be seen that the both the files have been transferred successfully in both the cases.