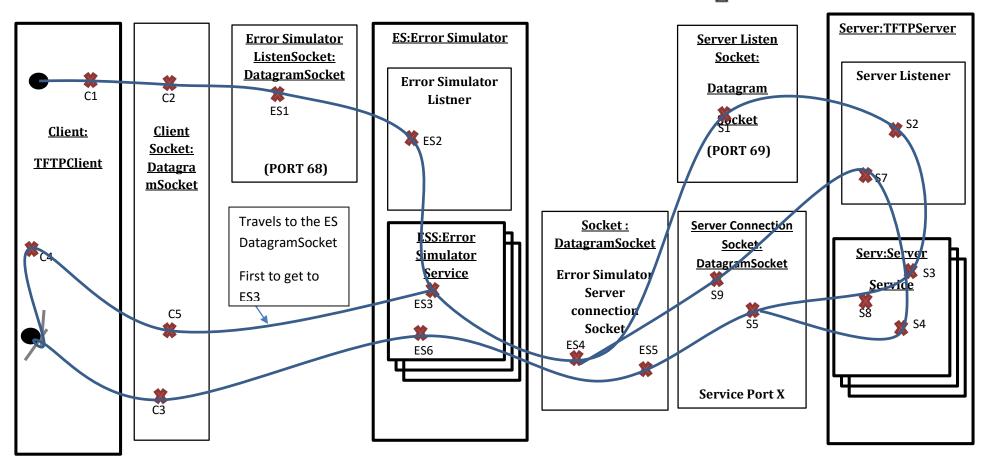
## UCM: Error Free Write request



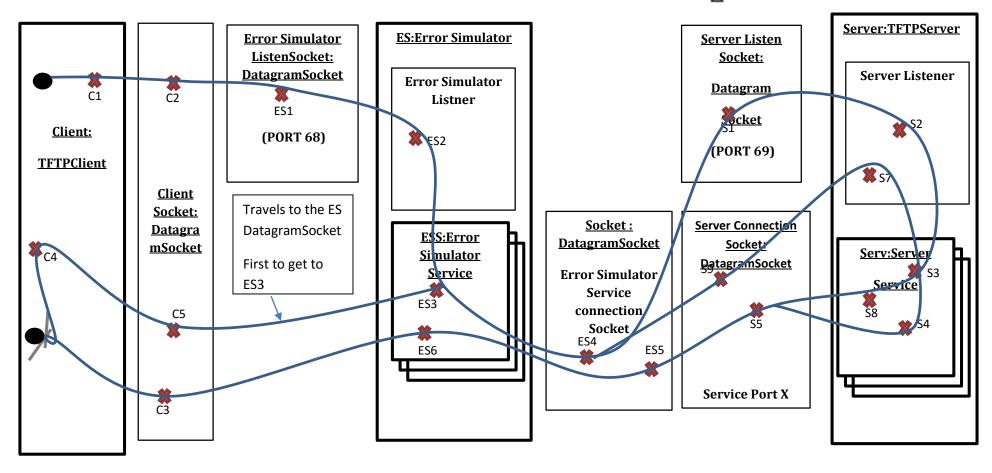
- Here, C4->C5 send preceding packets to the server
- Once ES is up, all proceeding packets for the current file transfer go toward the same port
- **C1**: Create write request packet
- **C2:** Send write request packet
- **C3:** Receive acknowledge packet
- **C4:** Prepare Data blocks
- C5: Send data block

- **ES1:** Receive write request Packet
- **ES2:** New Error service thread is created and started
- **ES3:** Rewrite packet address for server . (Port 69 on first packet, then forward to Server Service socket X)
- ES4: Send off packet
- **ES5:** Receive server response packet
- **ES6:** Rewrite address for client

- **S1:** Receive write request packet **S2:** New Server service thread is
- **S2:** New Server service thread is created and started
- **S3:** Create Acknowledge packet acknowledging that the write request packet was received
- **S4:** Check for packet formatting errors
- **S5:** Send Acknowledge packet and started

- **S6:** Receive data block
- **S7:** New Server service thread is created
- **S8:** Checks if the received data block is the correct data block
- **S9:** Send acknowledgement packet acknowledging that the data block was received

# UCM: Error Free Read request



- Here, C4->C5 send preceding packets to the server
- Once ES is up, all proceeding packets for the current file transfer go toward the same port
- **C1**: Create read request packet
- **C2:** Send read request packet
- **C3:** Receive acknowledge packet
- **C4:** Prepare Data blocks
- C5: Send data block

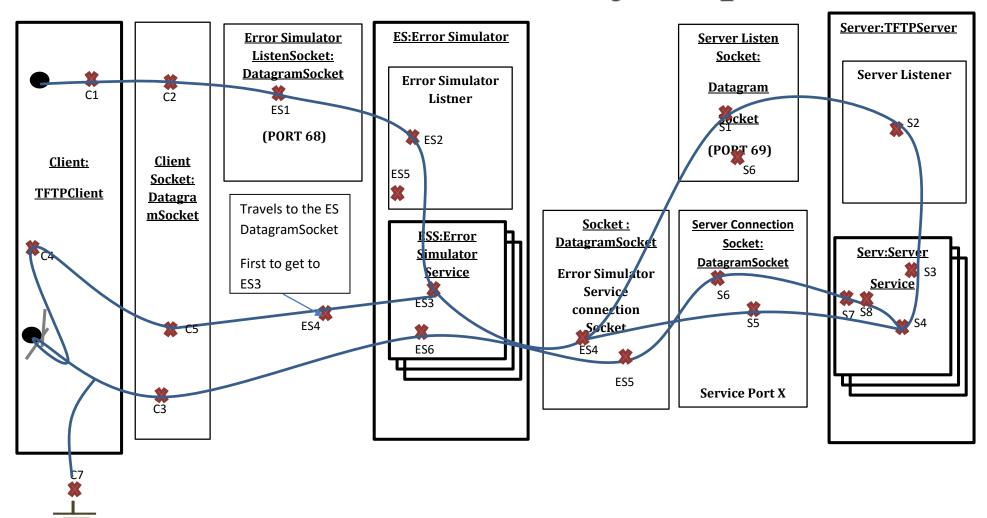
- **ES1:** Receive write request Packet
- **ES2:** New Error service thread is created and started
- **ES3:** Rewrite packet address for server . (Port 69 on first packet, then forward to Server Service socket X)
- **ES4:** Send off packet
- **ES5:** Receive server response packet
- **ES6:** Rewrite address for client

- **S1:** Receive read request packet **S2:** New Server service thread is
- created and started

  S3: Create Acknowledge packet
- acknowledging that the read request packet was received **S4:** Check for packet formatting
- **S4:** Check for packet formatting errors
- **S5:** Send Acknowledge packet and started

- S6: Receive data block
- **S7:** New Server service thread is created
- **S8:** Checks if the received data block is the correct data block **S9:** Send acknowledgement
- **S9:** Send acknowledgement packet acknowledging that the data block was received

# UCM: Error code 4-Any request



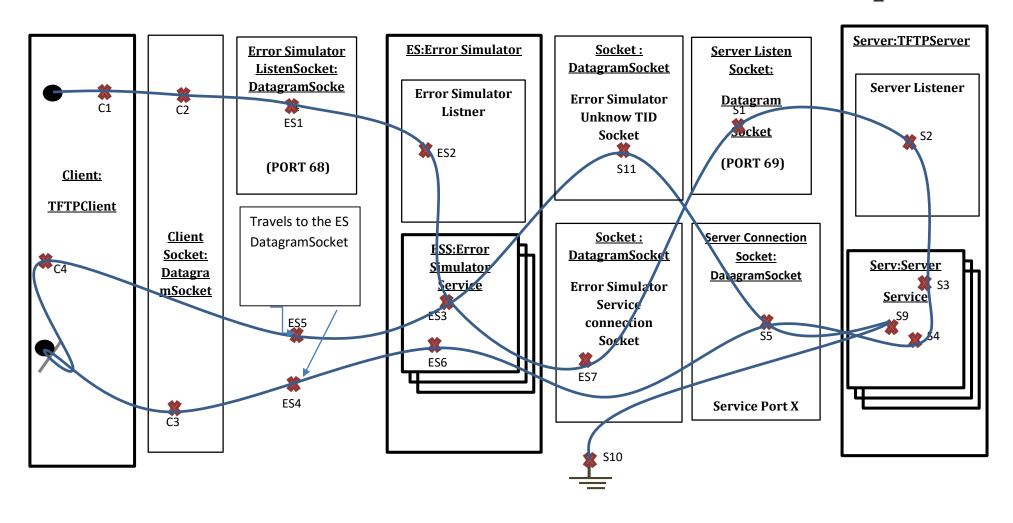
- Here EC3 doesn't do anything because it is not a data block packet
- The second path from C4 to C7repeats until an error packet is received from the server unless an error packet is received
- **C1**: Create write request packet
- **C2:** Send write request packet
- C3: Receive acknowledge packet
- **C4:** Prepare Data blocks
- C5: Send data block
- **C6:** Receive error packet
- C7: stop data block transfer and print error message

- **ES1:** Receive write request Packet
- ES2: New Error service thread is created and started
- ES3: Pass on or manipulates packets received if set to corrupt server packets
- **ES4:** Send packet out to server
- **ES5:** Receive server packet from reply
- **ES6:** Pass or manipulate packets received if set to corrupt client packets

- **S1:** Receive write request packet
- **S2:** New Server service thread is created and started
- **S3:** Create Acknowledge packet acknowledging that the write request packet was received
- **S4:** Check for packet formatting
- **S5:** Send Acknowledge packet and started

- S6: Receive data block **S7:** Send error packet
- stating that the packet received was corrupted **S8:** Checks if the received data block is the correct
- data block

### UCM: Error code 5- Server case - Write request

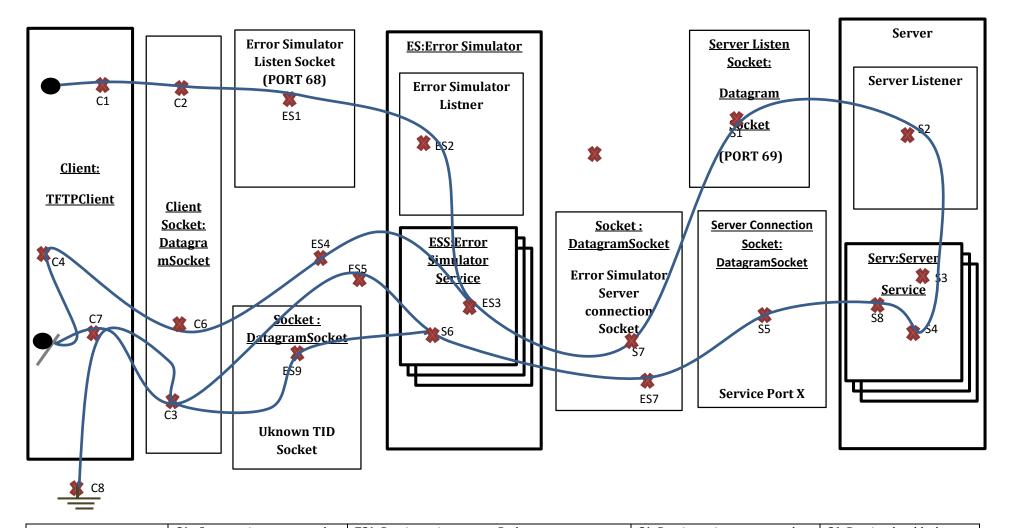


- ES5 and ES4 travel through ES Service socket
- \$10 is the end of the path for the Unknown host, but the transfer still continues for the correct TIDs
- **C1**: Create write request packet
- C2: Send write request packet
- **C3:** Receive acknowledge packet
- **C4:** Prepare Data blocks
- **ES1:** Receive write request Packet
- **ES2:** New Error service thread is created and started
- **ES3:** Forwards the packet, on every 3 packets, create a new socket Unknown TID Socket and forward thru that port
- **ES4:** Forward packet back to Client
- **ES5:** Forward packet to ES
- **ES6:** Change address to client
- **ES7:** Sends the copied packet through an error socket

- **S1:** Receive write request packet
- **S2:** New Server service thread is created and started
- S3: Create Acknowledge packet acknowledging that the write request packet was received
- **S4:** Check for packet formatting errors
- S5: Send Acknowledge packet and started

- **S6:** Receive data block
- **S7:** New Server service thread is created
- **S8:** Send acknowledgement packet acknowledging that the data block was received
- **S9:** Signals an error since the received packet is not from the expected port number
- **\$10:** Prints an illegal transfer ID error message
- **S11:** Send packet from different port

## UCM: Error code 5- Client case - Write request



C2: Send write request packet C3: Receive acknowledge packet C4: Prepare Data blocks C5: Send data block C6: Pass packet to ES C7: Signals an error since the packet received is not from the port expected C8: Send message back to TID  C2: New Error service thread is created and started started  ES2: New Error service thread is created and started S3: Create Acknowledge packet acknowledging that the write request packet was received S3: Create Acknowledge packet acknowledging that the write request packet was received S4: Check for packet formatting errors S5: Send Acknowledge packet and started S7: Recevie Packet service socket S8: Checks if the redation data block S4: Check for packet formatting errors S5: Send Acknowledge packet and started S9: Send error pa stating that the particular to packet service socket S8: Checks if the redation service socket	ock
packet C4: Prepare Data blocks C5: Send data block C6: Pass packet to ES C7: Signals an error since the packet received is not from the port expected  ES3: Manipulates and corrupts packets received ES4: Forward packet back to Client ES5: Forward packet to ES ES6: The packet received is not corrupted and passed on ES7: Accept packet from server ES8: Makes a copy of the acknowledgement packet  ES3: Create Acknowledge packet acknowledge packet and started  S8: Checks if the redata block acknowledge packet and started  S4: Check for packet formatting errors  S5: Send Acknowledge packet acknowledge p	from
C4: Prepare Data blocks C5: Send data block C6: Pass packet to ES C6: Pass packet to ES C7: Signals an error since the packet received is not from the port expected  E34: Forward packet back to Client ES5: Forward packet to ES ES6: The packet received is not corrupted and passed on ES7: Accept packet from server ES8: Makes a copy of the acknowledgement packet  E34: Forward packet back to Client request packet was received S4: Check for packet formatting errors stating that the write request packet was received S9: Send error passed on errors S5: Send Acknowledge packet and started	
C5: Send data block C6: Pass packet to ES C7: Signals an error since the packet received is not from the port expected  ES5: Forward packet to ES ES6: The packet received is not corrupted and passed on ES7: Accept packet from server ES8: Makes a copy of the acknowledgement packet  ES5: Forward packet to ES S4: Check for packet formatting errors S5: Send Acknowledge packet and started	ceived
C6: Pass packet to ES C7: Signals an error since the packet received is not corrupted and passed on ES7: Accept packet from server port expected  ES6: The packet received is not corrupted and passed on ES7: Accept packet from server ES8: Makes a copy of the acknowledgement packet and started  S4: Check for packet formatting errors  S5: Send Acknowledge packet and started	rrect
C7: Signals an error since the packet received is not from the port expected  Description:  Descript	
packet received is not from the port expected ES7: Accept packet from server ES8: Makes a copy of the acknowledgement packet and started received was corrected and started	ket
port expected ES8: Makes a copy of the acknowledgement packet and started	ket
	pted
Co. Sand massage hask to TID ESO. Sands conied packet through an arror sacket	
Co. Send message back to TID   E.S. Sends copied packet diffought all effor socket	
in unknown Host	