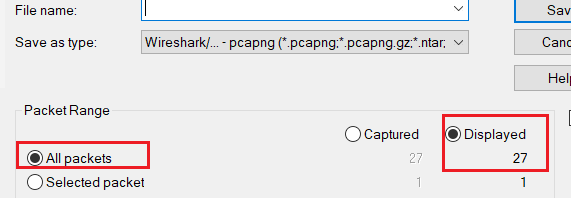
College of Computer Studies

AY 2020-2021

### NSCOM01 Project Demo

**General Instructions:**

* Download FileA.jpg and FileB.bin from the Animospace🡪 Files🡪 Project folder.
* Perform the following tests on your TFTP server and capture screenshots or packets as directed.
  + For screenshots, use compressed formats such as JPG or PNG. Reduce image resolution or crop images as long as important items still visible to shorten upload time. Use the instruction number in the demo procedure as the file name of the screenshot (e.g. 3.4.jpg or 4.1.2.png)
  + For wireshark, restart the capture for every operation you are asked to capture then stop the capture immediately afterwards. Set the filter to show only TFTP packets then save using the menu File🡪Export Specified Packets… Choose to save only displayed packets using the option provided.



* Place all images/captures under the following folder structure then compress into a ZIP archive for submission:

NSCOM01\_<surname>

File Upload

File Download

Error Handling

Bonus Features

**Demo Procedure:**

1. **Main program** – place under “NSCOM01\_<surname>” folder
   1. **Screenshot:** How to run your program
   2. **Screenshot:** Program landing screen or starting user prompt.

* Include your desktop and current computer time in the screenshot.

1. **File upload procedure**. Place all items under the “File Upload” folder :
   1. **Screenshot:** Target upload directory on the server is empty
   2. **Screenshot/s:** Inputting TFTP server address, file selection on client, inputting filename to be saved as on server.

* Use the file “FileA.jpg” provided in this demo pack for upload.
* Use the default block size: 512 bytes
* Set the file to be saved as “TestUploadA.jpg” on the server
  1. **Wireshark**: Capture and save all TFTP packets showing upload of file TestUploadA.jpg to server
  2. **Screenshot:** Client informs user of successful upload
  3. **Screenshot**: Directory showing that uploaded file is present on the server. Size and date modified of file must be visible
  4. **Screenshot**: Uploaded file can be opened on the client. Size and Date modified of file must be visible
  5. **Wireshark**: Upload “FileB.bin” to the server and capture and save all TFTP packets

1. **File download procedure**: Place all items under the “File Download” folder :
   1. **Screenshot:** Target download directory on the client is empty
   2. **Screenshot/s:** Inputting TFTP server address and name of file to be downloaded.

* Use the file previously uploaded (TestUploadA.jpg)
* Use the default block size: 512 bytes
  1. **Wireshark**: Capture and save all TFTP packets showing download of TestUploadA.jpg from server
  2. **Screenshot:** Client informs user of successful download
  3. **Screenshot**: Directory showing that downloaded file is present on the client.
  4. **Screenshot**: Downloaded file can be opened on the client. Size and Date modified of file must be visible
  5. **Wireshark**: Download “FileB.bin” to the server and capture and save all TFTP packets

1. **Error Handling:** Place all items under the “Error Handling” folder :
   1. Server not found error:
      1. Temporarily turn off/ close the TFTP server program
      2. **Screenshot :** netstat -asd output shows that UDP 69 is not open on the server
      3. **Screenshot/s:** Inputting TFTP server address and starting a download (provide any file name if needed by your program)
      4. **Screenshot:** Client indicates failure to contact server
   2. File not found error
      1. Rerun the TFTP server
      2. **Screenshot:** Target download directory on the server is empty
      3. **Screenshot/s:** Inputting TFTP server address and name of file to be downloaded then starting the download

* Use the file name “TestDownload.jpg”
  + 1. **Wireshark**: Capture and save all TFTP packets showing download attempt of file from server
    2. **Screenshot:** Client indicates a file not found error

1. **Bonus Items:** Place all items under the “Bonus Features” folder:
   1. **Screenshot/s:** Changing the transfer block size to another value other than 512 bytes
   2. **Screenshot/s:** File selection on client, inputting filename to be saved as on server “BonusUpload.jpg” and starting an upload
   3. **Wireshark**: Capture and save all packets showing download of file from server using the new block size
   4. **Screenshot/s:** Inputting filename “BonusUpload.jpg” to be downloaded from the server and starting download
   5. **Wireshark**: Capture and save all packets showing upload of file from server using the new block size