

데이터통신

- 실습 12주차. File Transfer -

충남대학교 데이터네트워크 연구실

Copy 주의 !

- 학생간의 과제 Copy시 무통보 0점 처리 예정
- 챗 GPT 사용이 아닌, 학생간의 동일한 과제 제출시에 대하여 판단

앞으로 할 과제들

이번주 (12주차): File Transfer (10점)

다음주 (13주차): Stop And Wait ARQ (20점)

2주뒤 (14주차): Go Back N ARQ (20점)



같은 주차에 echo service와 코드가 비슷해 과제를 풀어나가는데 큰 도움이 되었다.

Our file transfer protocol using UDP:
DCFT Protocol

DCFT1: Data communication file transfer protocol version 1

- Protocol: 서버 - 클라이언트간의 약속
- UDP 기반 파일 전송 약속

파일 정보 요청

- \Rightarrow INFO FILENAME
- \Leftarrow UTF-8 Encoded Text (File size in bytes)
 - FILENAME이 없을 경우 UTF-8 Encoded '404 Not Found'

파일 다운로드 요청

- \Rightarrow DOWNLOAD FILENAME
- \Leftarrow File Binary

실습:

UDP File Transfer Client

실습1)File Transfer Client

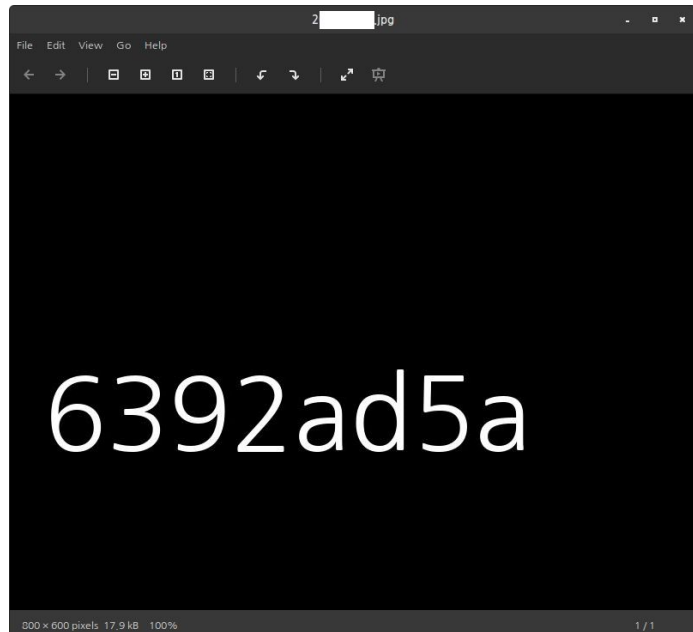
서버

- GCP에 실행해둔 공개 서버

```
Ready to send using <socket.socket fd=3, family=AddressFamily.AF_INET, type=SocketKind.SOCK_DGRAM, proto=0, laddr=('0.0.0.0', 0)>  
Filename: 2.jpg  
Request 2.jpg to (34.145.20.110, 3034)  
Receiving from ('34.145.20.110', 3034): 17933/17933  
File download success  
Filename: █
```

클라이언트

- 서버에 DCFT1 에 맞추어서 요청
 - 학번.JPG(대문자)
- 서버가 반환한 데이터 '저장'
 - jpg 이미지



실습1) Client code sample

```
while True:
    try:
        filename = input('Filename: ').strip()
        request = f'INFO {filename}'
        sock.sendto(request.encode('utf-8'), (FLAGS.address, FLAGS.port))

        response, server = sock.recvfrom(FLAGS.chunk_maxsize)
        response = response.decode('utf-8')
        if response == '404 Not Found':
            print(response)
            continue
        size = int(response)

        request = f'DOWNLOAD {filename}'
        sock.sendto(request.encode('utf-8'), (FLAGS.address, FLAGS.port))
        print(f'Request {filename} to ({FLAGS.address}, {FLAGS.port})')

        remain = size
        with open(filename, 'wb') as f:
            while remain > 0:
                ...
                f.write(chunk)
                ...
            print(f'File download success')
    except KeyboardInterrupt:
        print(f'Shutting down... {sock}')
        break
```


과제 1: File transfer server / client (Basic)

과제1) File Transfer Server / Client

- VM 에서 서버 실행
 - Docker / Cloud(GCP) 등 가능
 - 클라이언트는 Host PC에서
 - localhost (127.0.0.1) 불가능
- 요청받은 파일 1500 Bytes 씩 잘라서 전송
 - 1500보다 짧은 길이는 FLAGS.mtu
 - 요청 받은 파일이 없을 경우 404 Not Found 응답 (UTF-8)
 - 서버 시작시 전송가능한 파일 리스트 및 정보 저장해두면 좋음 (파일 이름, 경로, 크기)
- 파일 전송 전 파일 크기 전송
 - Client 에서 파일을 얼마나 다운로드 받아야하는지 미리 알아야 함 (UDP는 연결 종료 메시지가 없음)
- Client Code는 그대로 써도 무방

```
Ready to file transfer
card.png: {'size': 48416, 'path': '/home/dnlab/2024-DataCommunications/week12-FileTransfer/FileTransferBasic/files/card.png'}
thatthat.webm: {'size': 422044555, 'path': '/home/dnlab/2024-DataCommunications/week12-FileTransfer/FileTransferBasic/files/thatthat.webm'}
talesofsensuality.webm: {'size': 157583677, 'path': '/home/dnlab/2024-DataCommunications/week12-FileTransfer/FileTransferBasic/files/talesofsensuality.webm'}
1.jpg: {'size': 4865992, 'path': '/home/dnlab/2024-DataCommunications/week12-FileTransfer/FileTransferBasic/files/1.jpg'}
2.jpg: {'size': 5220922, 'path': '/home/dnlab/2024-DataCommunications/week12-FileTransfer/FileTransferBasic/files/2.jpg'}
3.jpg: {'size': 4169896, 'path': '/home/dnlab/2024-DataCommunications/week12-FileTransfer/FileTransferBasic/files/3.jpg'}
Listening on <socket.socket fd=3, family=AddressFamily.AF_INET, type=SocketKind.SOCK_DGRAM, proto=0, laddr=('0.0.0.0', 3034)>
```

```
rBasic $ python3 main_client.py --address 192.168.56.116 --port 3034
Ready to send using <socket.socket fd=3, family=AddressFamily.AF_INET, type=SocketKind.SOCK_DGRAM, proto=0, laddr=('0.0.0.0', 0)>
Filename: 1.jpg
Request 1.jpg to (192.168.56.116, 3034)
File download success
```

과제2: File transfer server / client with packet corruption

과제2) packet corruption 시뮬레이션

과제 목표

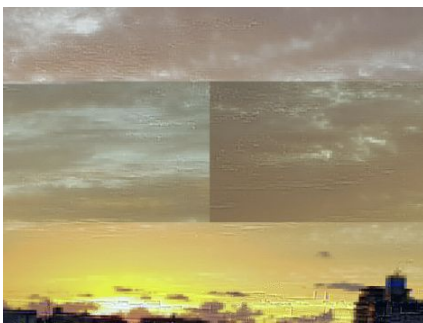
- UDP 파일 송수신간 Packet Corruption 발생 시뮬레이션
 - 이미지 파일이 손상을 보기에 좋음
 - 이미지 파일 크기에 따라서 손상 확률 조절
- tc (traffic control) utility 활용 [\[매뉴얼\]](#)

- VM에서 서버 실행
 - Docker / Cloud 등 가능
 - 클라이언트는 Host PC에서 실행 ok
 - localhost (127.0.0.1) 불가능
 - 힌트) 과제1 해결 후 진행하는게 편함!

```
sudo tc qdisc add dev enp0s3 root netem corrupt 10%
```

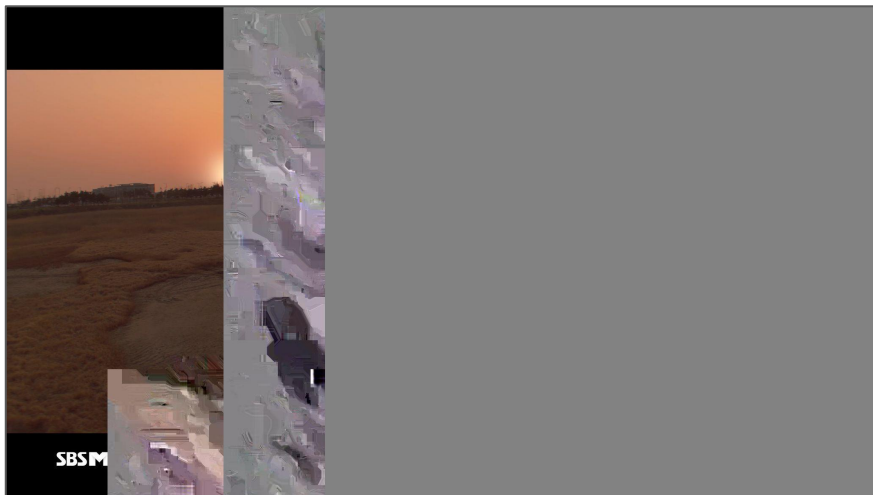
```
tc qdisc show
```

```
sudo tc qdisc delete dev enp0s3 root netem corrupt 10%
```



tc netem 활용 packet corruption 시뮬레이션 스크린샷

```
dnlab@dnlab-dc22:~/2024-DataCommunications/week12-FileTransfer/FileTransferBasic
$ sudo tc qdisc show
qdisc noqueue 0: dev lo root refcnt 2
qdisc fq_codel 0: dev enp0s3 root refcnt 2 limit 10240p flows 1024 quantum 1514
target 5.0ms interval 100.0ms memory_limit 32Mb ecn
qdisc fq_codel 0: dev enp0s8 root refcnt 2 limit 10240p flows 1024 quantum 1514
target 5.0ms interval 100.0ms memory_limit 32Mb ecn
dnlab@dnlab-dc22:~/2024-DataCommunications/week12-FileTransfer/FileTransferBasic
$ sudo tc qdisc add dev enp0s3 root netem corrupt 10%
dnlab@dnlab-dc22:~/2024-DataCommunications/week12-FileTransfer/FileTransferBasic
$ sudo tc qdisc show
qdisc noqueue 0: dev lo root refcnt 2
qdisc netem 8001: dev enp0s3 root refcnt 2 limit 1000 corrupt 10%
qdisc fq_codel 0: dev enp0s8 root refcnt 2 limit 10240p flows 1024 quantum 1514
target 5.0ms interval 100.0ms memory_limit 32Mb ecn
dnlab@dnlab-dc22:~/2024-DataCommunications/week12-FileTransfer/FileTransferBasic
$ python3 main_server.py --debug
```



실습12 과제 정리 및 채점 배점

- 보고서 제출(기본): 1점
- 파일 전송 서버, 클라이언트: 7점
 - VM: Server
 - Host: Client
 - 패킷 캡처 스크린샷으로 IP 주소가 다름을 증명할 것 (localhost 가 아니어야 인정!)
- Packet corruption 실험: 2점
 - VM: Server
 - Host: Client
 - 패킷 corruption 으로 다운로드 완료된 파일이 손상됨을 보일 것 (wireshark)
 - 패킷 캡처: 정상 vs. 손상
 - 이미지 or 동영상 파일 손상: 파일 열어보기
- 기타 부정행위시 감점
- zip 파일 압축: DC02-학번-이름.zip
 - 보고서 (학번-이름.pdf)
 - 자신이 했음을 증명
⇒ 스크린샷 및 설명
 - 소스코드 (*.py)
 - 정상 및 손상된 이미지 또는 동영상
- e-learning 사이버캠퍼스 제출
- 설문조사: [\[클릭\]](#)
- 제출 기한: 2025. 05. 28. 23:59
- 추가 제출 1일간 (30% 감점)