

[2019 컴퓨터네트워크]

다중 클라이언트 연결 웹 서버 구조와 성능 테스트2

데이터 네트워크 연구실 (633호)
김수현

출석

TCP Socket Programming을 이용한 출석

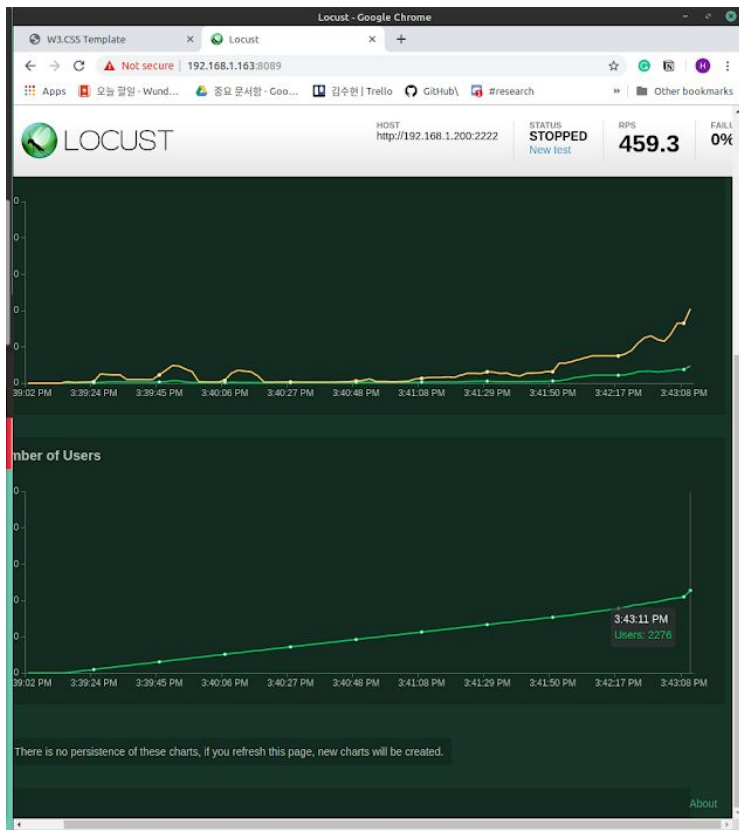
- IP : 168.188.129.156
- Port : 5555
- Code: 6357
- input OK 뜨면 출석 완료된것

과제

다중 클라이언트 지원 웹 서버와 성능테스트

1. HTTP Server 구현
 - i. Select를 이용한 Multiplexing
2. Locust를 이용한 Server Stress Test
 - i. 웹서버 구현 방법에 따른 성능비교
 1. 수용가능한 최대 RPS
 2. 다중 클라이언트 환경에서 평균 응답 시간
 - ii. Fail이 발생 했을 경우까지의 chart 이미지 캡처

Select 이용 Server Test 결과



LOCUST

HOST: http://192.168.1.200:2222

STATUS: STOPPED

RPS: 459.3

Statistics | Charts | Failures | Exceptions | Download Data

Type	Name	# Requests	# Fails	Median (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)
GET	/	73141	1	320	553	2.14338302 6123047	42274.8372 55477905	14939
Total		73141	1	320	553	2.14338302 6123047	42274.8372 55477905	14939

웹 서버 구현 방법에 따른 성능 차이

Hatch Rate 10, First Fail	# User	Request Per Second	Average Response time
Single Process	1,571	314.4	249
Multiprocess	2,438	435.6	716
Multithread	2,079	355.1	926
Select	2,618	487.2	807

# User, Hatch Rate 2000 10000 Requests	# failures	Request Per Second	Average Response time
Single Process	278	435.28	684
Multiprocess	0	520.15	431
Multithread	0	478.99	575
Select	0	518.03	377

1 - 0. select

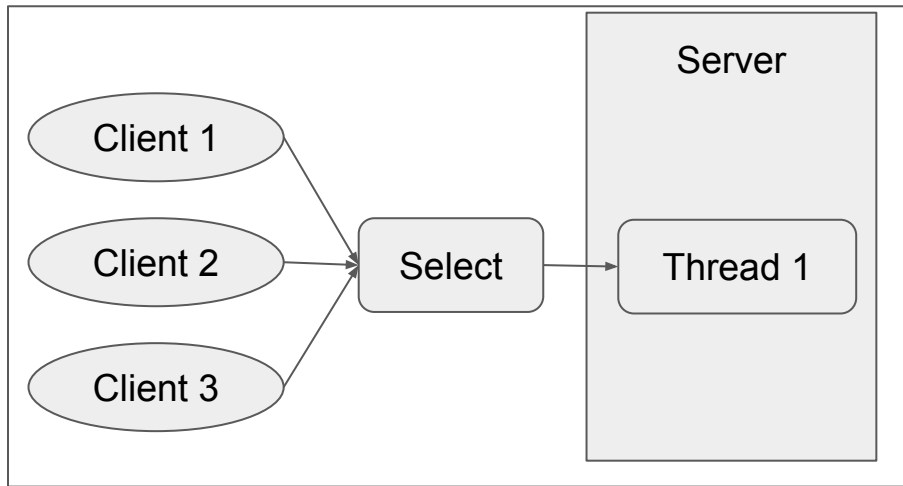
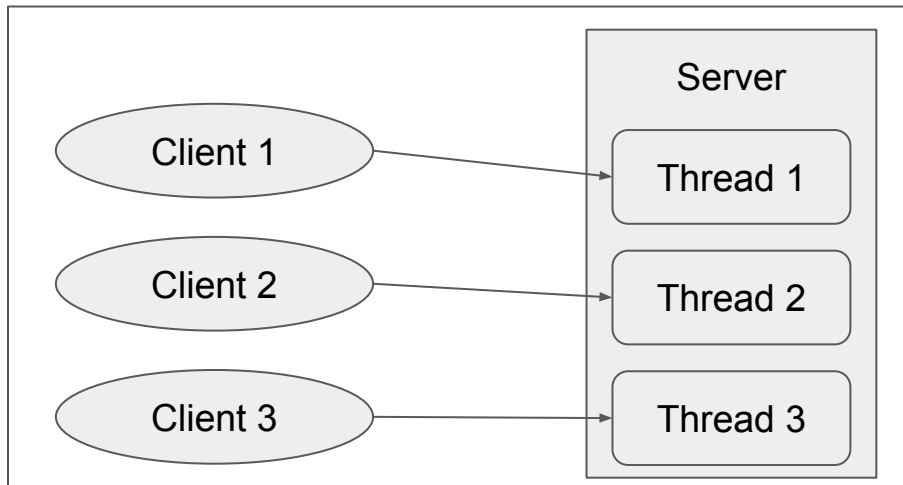
- `select ()`: 하나 이상의 file descriptor가 I/O에 대해서 'ready'상태가 될 때까지 기다리면서 여러 파일 디스크립터를 모니터
- <http://man7.org/linux/man-pages/man2/select.2.html>

```
readable = [server_socket]
writable = []
error = []

while True:
    try:
        r_list, w_list, e_list = select.select(
            readable, writable, error
        )
```

1 - 2. Multiplexing

- `select()`가 여러 socket descriptor를 관리
- thread, process와 달리 context switching이 없음
- 그림 출처 : <https://medium.com/coderscorner/tale-of-client-server-and-socket-a6ef54a74763>



Review

채점결과

- 2주차 - 5주차 채점해서 review 오늘 내로 올릴 예정입니다.
 - <https://docs.google.com/spreadsheets/d/1v9OxfK8VcHWU4-wdimDCS65gan0MBC-3nby28JTZxEc/edit?usp=sharing>
-
- 정확한 점수확인을 원하시는 분들은 메일보내주세요.

6주차

- Server stress test
 - Sing process와 multithread, multiprocess 성능 비교
 - join()의 영향!
- Locust server localhost에서 실행한 학생
 - 감점
 - 실제 NIC에서 나가지 않기 때문에(loopback) 객관적 비교가 어려움

과제제출 유의사항

과제제출

- 제출기한 : 2019.10.31 23:59시 까지
 - 사이버 캠퍼스
- 제출파일
 - select에 대한 코드(.py), 시각화 파일
 - locust chart 캡처파일(보고서에 같이 넣을것)
 - 보고서: CN_과제번호_학번_이름.pdf
 - .doc, .hwp 0점
- 딜레이 1일당 10%감점
 - 10점 만점 기준, 1일 딜레이인 경우 9점
- 과제 카피 적발시 보여준사람 본사람 0점처리
 - 카피 적발기준 : 과제 유사도 80%이상

유의사항

- 보고서:

- 과제목표(도출해야할 결과)
- 코드설명과 과제해결방법(실행결과사진도 보고서에 첨부, 코드에대한 캡처도 필요)
- 과제후기(느낀점 및 조교에게 하고싶은말, 선택사항)
- Doc, HWP 인경우 채점안함

- 질문

- 메일주소 : shkim950921@cs-cnu.org
- [네트워크] 학번_이름 양식 지킬것