

[네트워크프로그래밍]

지정 주제 프로젝트 보고서

소속 : IT대학 - 소프트웨어학부

학번 : 20192851, 20192829, 20192847

이름 : 윤성준, 박민수, 양조은

목 차

1. 프로토콜 플로우 및 패킷 구조

- 1.1. Main server와 Working server간의 프로토콜 플로우
- 1.2. Main server와 Working server간의 패킷 페이로드 정의
- 1.3. 멀티프로세스(또는 스레드) 사용 시 역할에 대한 설명

2. 프로그램 설명

3. PoW nonce (8개)

- 3.1. 챌린지 학번 단독 실행 (난이도 7)
- 3.2. 챌린지 이름 단독 실행 (난이도 7)
- 3.3. 챌린지 학번 분산 실행 (난이도 7)
- 3.4. 챌린지 이름 분산 실행 (난이도 7)
- 3.5. 챌린지 학번 단독 실행 (난이도 8)
- 3.6. 챌린지 이름 단독 실행 (난이도 8)
- 3.7. 챌린지 학번 분산 실행 (난이도 8)
- 3.8. 챌린지 이름 분산 실행 (난이도 8)

4. 단독 실행 vs. 분산 실행

- 4.1. 챌린지 학번 또는 챌린지 이름의 난이도 7일 때의 실행시간 차이 (비율)
- 4.2. 챌린지 학번 또는 챌린지 이름의 난이도 8일 때의 실행시간 차이 (비율)

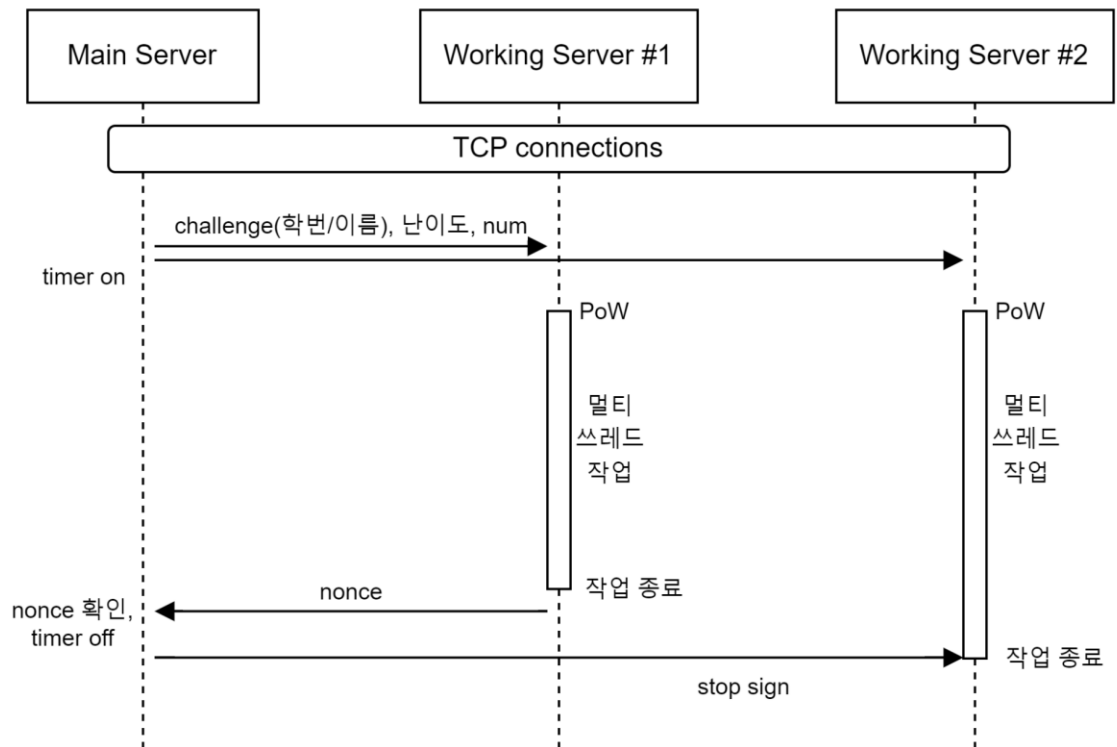
5. Port Forwarding

6. 프로젝트 회고

1. 프로토콜 플로우 및 패킷 구조

1.1. Main server와 Working server간의 프로토콜 플로우

(num은 각 Working Server의 시작 탐색 nonce를 의미)



먼저 Main Server 구동시키고 해당 IP와 Port를 통하여 두 Working Server가 접속하면 TCP 연결이 형성된다. Main Server에서 challenge와 난이도를 설정하고, 이 값들을 각 Working Server의 탐색 시작 nonce 값인 num과 함께 전송한다. 그럼 두 Working Server는 각각 주어진 num부터 시작해서 nonce값 탐색, 즉 PoW를 진행하게 된다. 이 과정에서 해시 연산을 통해 특정 조건을 만족하는 nonce 값을 찾았다면, 그 값을 Main Server에게 전송한다. 이 때, Main Server는 두 Working Server 중 어느 하나로부터 값을 먼저 받게 되면, 그 값을 채택하고 이를 나머지 서버에게 알린다. (stop sign)

1.2. Main server와 Working server간의 패킷 페이로드 정의

- Challenge:
 - 학번: 20192829||20192851||20192847
 - 이름: 박민수||윤성준||양조은
- 난이도: 7 or 8

- 탐색 시작 번호(num)
 - Working Server #1: 0 ~ 4,611,686,018,427,387,902
 - Working Server #2: 4,611,686,018,427,387,903 ~

난이도 7인 경우: Hash(Challenge || Nonce) = 0000000xxxxxxxxxxx...
 난이도 8인 경우: Hash(Challenge || Nonce) = 00000000xxxxxxxxxxx...

1.3. 멀티 스레드 사용

본 프로젝트에서 Working Server는 5개의 PoW 연산 스레드와 1개의 Stop 리스너 스레드를 만들어 병렬처리를 한다.

1.3.1 PoW 연산 스레드 함수

```
#define NUM_THREADS 5          // 작업을 수행할 스레드의 수
#define RANGE_PER_THREAD 20000 // 한 번에 각 스레드가 담당할 nonce 값 범위
#define MAX_NONCE LONG_MAX     // 최대 nonce 값

void *pow_worker(void *threadid)
{
    long tid = (long)threadid;

    unsigned char text[64];
    unsigned char hash[SHA256_DIGEST_LENGTH];

    for (long nonce = start_nonce + tid * RANGE_PER_THREAD; nonce <
        MAX_NONCE; nonce += NUM_THREADS * RANGE_PER_THREAD)
    {
        for (long my_nonce = nonce; my_nonce < nonce + RANGE_PER_THREAD;
            my_nonce++)
        {
            pthread_mutex_lock(&found_mutex);

            if (stop_received || found_nonce != -1)
```

```

    {
        pthread_mutex_unlock(&found_mutex);
        pthread_exit(NULL);
    }

    pthread_mutex_unlock(&found_mutex);
    sprintf(text, "%s%d", challenge, my_nonce);
    compute_SHA256(hash, text, strlen(text));
    if (is_valid(hash, difficulty))
    {
        print_hash(hash);
        pthread_mutex_lock(&found_mutex);
        if (found_nonce == -1)
        { // double check inside the lock
            found_nonce = my_nonce;
            found_thread_id = tid; // Store the thread ID
        }
        pthread_mutex_unlock(&found_mutex);
        pthread_exit(NULL);
    }
}

pthread_exit(NULL);
}

```

pow_worker함수는 NUM_THREADS x RANGE_PER_THREAD의 총 합계를 NUM_THREADS 수의 스레드가 RANGE_PER_THREAD 단위로 분할하여 연산을 수행하도록 구성된다. 현재 설정에 따르면, 10만 개의 숫자를 5개의 스레드가 각각 2만 개씩 나누어서 처리하게 된다. 스레드 수가 5개를 초과하면 오히려 연산 시간이 증가하는 결과가 나타나, 총 스레드 수는 5개로 결정했다. 그러나 RANGE_PER_THREAD의 최적값은 추가적인 연구를 통해 결정될 수 있을 것으로 보인다. MAX_NONCE는 다양한 실험을 통해, 주어진 프로젝트 범위 내에서 Nonce의 범위가 대체적으로 int 타입보다는 더 넓어 long 타입을 사용하는 것이 적합하다고 판단된다.

각 스레드는 총 10만 개의 숫자 중 2만 개씩을 검사한다. 이 과정에서 전역변수인 stop_received가 false이고 found_nonce가 -1이 아닌 경우, 다른 워킹 서버 또는 스레드에서 이미 nonce를 찾았다고 판단하여 연산을 중단한다. 그렇지 않은 경우, compute_SHA256 함수를 사용하여 해시 값을 계산하고, 해당 해시 값이 설정된 난이도 요구사항을 만족하는지 확인한다. 만약 만족하는 경우, 해당 해시 값을 화면에 출력하고, found_nonce를 현재의 my_nonce 값으로 변경하며, nonce를 찾은 스레드의 ID를 found_thread_id에 저장한다. 이러한 모든 작업은 각 스레드에서 동시에 실행되므로, 뮤텁스를 사용하여 스레드 간의 상호 배제를 보장한다.

1.3.2 리스너 스레드

```
void *message_listener(void *arg)
{
    int sock = *((int *)arg);
    char message[BUF_SIZE];
    while (1)
    {
        if (read(sock, message, BUF_SIZE - 1) > 0)
        {
            if (strcmp(message, "stop") == 0)
            {
                pthread_mutex_lock(&found_mutex);
                stop_received = true;
                printf("Stopped computation due to stop message\n");
                pthread_mutex_unlock(&found_mutex);
                break;
            }
        }
    }
    pthread_exit(NULL);
}
```

리스너 쓰레드는 message_listener 함수를 통해 main server에서 stop 메시지를 보내는지
항시 확인한다. stop 메시지가 온다면 stop_received를 true로 바꿈으로서 연산 쓰레드에게
stop 메시지를 알린다. 그리고 화면에 Stopped computation due to stop message을 출력
한다.

2. 프로그램 설명

2.1. 함수 설명

2.1.1. Main Server

main 함수: main 함수는 총 5가지 단계로 동작한다.

1. 서버 소켓 설정 및 working server 연결한다.
2. 사용자에게 Challenge와 Difficulty 입력 받는다.
3. Working server에 Challenge와 Difficulty, Start_nonce를 전송한다.
4. Select() 함수를 통해 수신 데이터를 감지하여 PoW결과와 소요시간을 출력한다.
5. 다른 클라이언트에게 중지 신호를 전송한다.

2.1.2. Working Server

- **compute_SHA256** 함수: 이 함수는 주어진 데이터의 OpenSSL의 SHA-256 해시를 계산한다.
- **check_4bits** 함수: 이 함수는 주어진 바이트에서 첫 4비트가 모두 0인지 확인한다.
- **is_valid** 함수: 이 함수는 주어진 해시가 PoW의 난이도 요구사항을 충족하는지 확인한다.
- **print_hash** 함수: 이 함수는 주어진 해시를 표준 출력에 출력한다.
- **pow_worker** 함수: 이 함수는 각 스레드에서 실행되며 PoW를 수행한다. 주어진 nonce 범위에서 유효한 해시를 찾는 작업을 수행하며, 이 작업은 멀티스레딩 환경에서 병렬로 진행한다.
- **message_listener** 함수: 이 함수는 메인 서버에서 stop 메시지를 수신하는 데 사용한다. stop 메시지를 수신하면, stop_received 플래그를 설정하여 모든 pow_worker 스레드가 작업을 중지하도록 한다.
- **main 함수:**
 1. 소켓을 생성하고 메인 서버에 연결한다.
 2. 메인 서버로부터 PoW에 필요한 정보를 수신한다.
 3. PoW를 수행하는 작업자 스레드 5개와 stop 메시지를 듣는 리스너 스레드 1개를 생성한다.
 4. 스레드 join()을 통해 작업자 스레드가 종료될길 기다린다.
 5. 모든 작업자 스레드가 종료되면, 리스너 스레드를 취소한다.
 6. PoW가 성공하면 성공 메시지와 nonce 값을 메인 서버에 전송한다. stop 메시지를 수신하거나 탐색에 실패하면 실패 메시지를 출력한다.

3. PoW nonce (8개)

3.1. 챌린지 학번 단독 실행 (난이도 7)

3.1.1. INPUT

- Challenge : 20192829||20192851||20192847
- 난이도 : 7

3.1.2. OUTPUT

- nonce: 740062070
- time: 239.779680초 = 3.996328분

```
ssu20192851@ubuntu: ~/Desktop/zzProject/pow1
File Edit View Search Terminal Help
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ gcc main_server.c -o main_server
main_server.c: In function 'main':
main_server.c:23:48: warning: excess elements in array initializer
    long start_nonce_values[MAX_CLIENTS] = {0, LONG_MAX / 2}; // <-- Add start_
nonce values

main_server.c:23:48: note: (near initialization for 'start_nonce_values')
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./main_server^C
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ gcc main_server.c -o main_server
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./main_server
Connected client 0
Enter Challenge: 20192829||20192851||20192847
Enter Difficulty: 7
Message from client 0: Success! Nonce: 740062070

Time for PoW: 239.779680 seconds
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

```
ssu20192851@ubuntu: ~/Desktop/zzProject/pow1
File Edit View Search Terminal Help
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./working_server2
Connected to main server
challenge: 20192829||20192851||20192847, difficulty: 7, start_nonce: 0
0000000e9d81ea9a3e983758063b061aa51081d6eb8c494ec17a1a2912cb4b41
Thread #3 found a valid nonce: 740062070
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

3.2. 챌린지 이름 단독 실행 (난이도 7)

3.2.1. INPUT

- challenge: 박민수||윤성준||양조은
- time: 18.197316초

3.2.2. OUTPUT

- nonce: 78249635
- time: 18.197316초

```
ssu20192851@ubuntu: ~/Desktop/zzProject/pow1
File Edit View Search Terminal Help
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./main_server
Connected client 0
Enter Challenge: 박민수||윤성준||양조은
Enter Difficulty: 7
Message from client 0: Success! Nonce: 78249635

Time for PoW: 18.197316 seconds
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

```
ssu20192851@ubuntu: ~/Desktop/zzProject/pow1
File Edit View Search Terminal Help
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./working_server2
Connected to main server
challenge: 박민수||윤성준||양조은, difficulty: 7, start_nonce: 0
00000007bb4a393cfa3029f0d209c0195fe6c035561c6721608cf5ac3558c54d
Thread #2 found a valid nonce: 78249635
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

3.3.챌린지 학번 분산 실행 (난이도 7)

3.3.1. INPUT

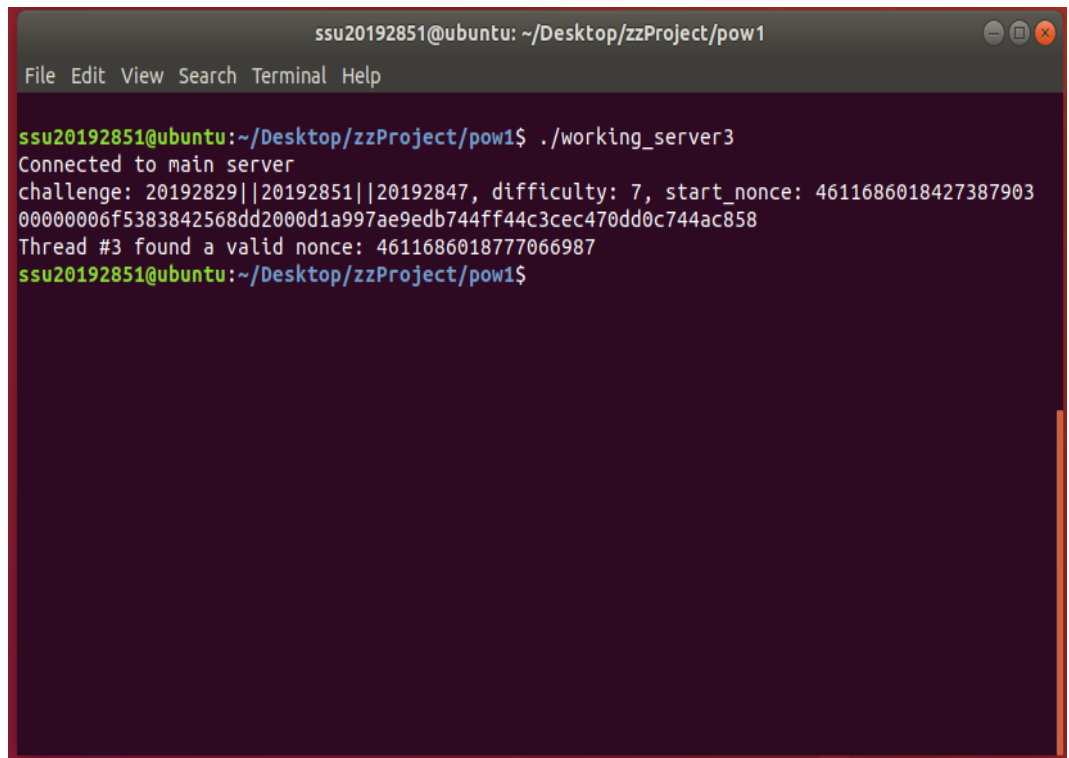
- Challenge : 20192829||20192851||20192847
- 업무분담 - Working Server 1 : 0 ~ 4,611,686,018,427,387,902
- 업무분담 - Working Server 2 : 4,611,686,018,427,387,903 ~
- 난이도 : 7

3.3.2. OUTPUT

- nonce: 4611686018777066987
- time: 92.630825초 = 1.54384708333분

```
ssu20192829@ubuntu: ~  
File Edit View Search Terminal Help  
ssu20192829@ubuntu:~$ ./main_server  
Connected client 0  
Connected client 1  
Enter Challenge: 20192829||20192851||20192847  
Enter Difficulty: 7  
Message from client 1: Success! Nonce: 4611686018777066987  
  
Time for PoW: 92.630825 seconds  
ssu20192829@ubuntu:~$
```

```
ssu20192847@ubuntu: ~/Network/pow  
File Edit View Search Terminal Help  
ssu20192847@ubuntu:~/Network/pow$ ./working_server  
Connected to main server  
challenge: 20192829||20192851||20192847, difficulty: 7, start_nonce: 0  
Stopped computation due to stop message  
Fail....  
ssu20192847@ubuntu:~/Network/pow$
```

A terminal window titled 'ssu20192851@ubuntu: ~/Desktop/zzProject/pow1' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal output shows the execution of './working_server3', which connects to a main server and provides a challenge. Thread #3 successfully finds a valid nonce.

```
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./working_server3
Connected to main server
challenge: 20192829||20192851||20192847, difficulty: 7, start_nonce: 4611686018427387903
00000006f5383842568dd2000d1a997ae9edb744ff44c3cec470dd0c744ac858
Thread #3 found a valid nonce: 4611686018777066987
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

3.4. 챌린지 이름 분산 실행 (난이도 7)

3.4.1. INPUT

- Challenge : 박민수||윤성준||양조은
- 업무분담 - Working Server 1 : 0 ~ 4,611,686,018,427,387,902
- 업무분담 - Working Server 2 : 4,611,686,018,427,387,903 ~
- 난이도 : 8

3.4.2. OUTPUT

- nonce: 46758511
- time: 12.598665초

```
ssu20192829@ubuntu: ~  
File Edit View Search Terminal Help  
ssu20192829@ubuntu:~$ ./main_server  
Connected client 0  
Connected client 1  
Enter Challenge: 20192829||20192851||20192847  
Enter Difficulty: 7  
Message from client 1: Success! Nonce: 4611686018777066987  
  
Time for PoW: 92.630825 seconds  
ssu20192829@ubuntu:~$ ./main_server  
Connected client 0  
Connected client 1  
Enter Challenge: 양조은||박민수||윤성준  
Enter Difficulty: 7  
Message from client 0: Success! Nonce: 46758511  
  
Time for PoW: 12.598665 seconds  
ssu20192829@ubuntu:~$
```

```
ssu20192847@ubuntu: ~/Network/pow  
File Edit View Search Terminal Help  
ssu20192847@ubuntu:~/Network/pow$ ./working_server  
Connected to main server  
challenge: 양조은||박민수||윤성준, difficulty: 7, start_nonce: 0  
000000096e4abca87a1a464095f04ad2756d5719c65a97391e606a0fc68712b1  
Thread #2 found a valid nonce: 46758511  
ssu20192847@ubuntu:~/Network/pow$
```

```
ssu20192851@ubuntu: ~/Desktop/zzProject/pow1
File Edit View Search Terminal Help
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./working_server3
Connected to main server
challenge: 양조은||박민수||윤성준, difficulty: 7, start_nonce: 4611686018427387903
Stopped computation due to stop message
Fail....
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

3.5. 챌린지 학번 단독 실행 (난이도 8)

3.5.1. INPUT

- Challenge : 20192829||20192851||20192847
- 난이도 : 8

3.5.2. OUTPUT

- nonce: 1632111841
- time: 522.887123초 = 8.714785383333분

```
ssu20192851@ubuntu: ~/Desktop/zzProject/pow1
File Edit View Search Terminal Help
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ gcc main_server.c -o main_server
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./main_server
Connected client 0
Enter Challenge: 20192829||20192851||20192847
Enter Difficulty: 8
Message from client 0: Success! Nonce: 1632111841

Time for PoW: 522.887123 seconds
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

```
ssu20192851@ubuntu: ~/Desktop/zzProject/pow1
File Edit View Search Terminal Help
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./working_server2
Connected to main server
challenge: 20192829||20192851||20192847, difficulty: 8, start_nonce: 0
000000000a8ad40cde16b79c522a92b7ebd38cbf55fd9d085006bf26444c4c1e
Thread #0 found a valid nonce: 1632111841
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

3.6. 챌린지 이름 단독 실행 (난이도 8)

3.6.1. INPUT

- Challenge : 박민수||윤성준||양조은
- 난이도 : 8

3.6.2. OUTPUT

- nonce:
- time: 3242.578536초


```
ssu20192851@ubuntu: ~/Desktop/zzProject/pow1
File Edit View Search Terminal Help
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./main_server
Connected client 0
Enter Challenge: 양조은||박민수||윤성준
Enter Difficulty: 8
Message from client 0: Success! Nonce: 10139854110

Time for PoW: 3242.578536 seconds
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

```
ssu20192851@ubuntu: ~/Desktop/zzProject/pow1
File Edit View Search Terminal Help
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./working_server2
Connected to main server
challenge: 양조은||박민수||윤성준, difficulty: 8, start_nonce: 0
000000003bbc2db40b2227470c8b605424234d30847147c0c083b088ac35ba96
Thread #2 found a valid nonce: 10139854110
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

3.7. 챌린지 학번 분산 실행 (난이도 8)

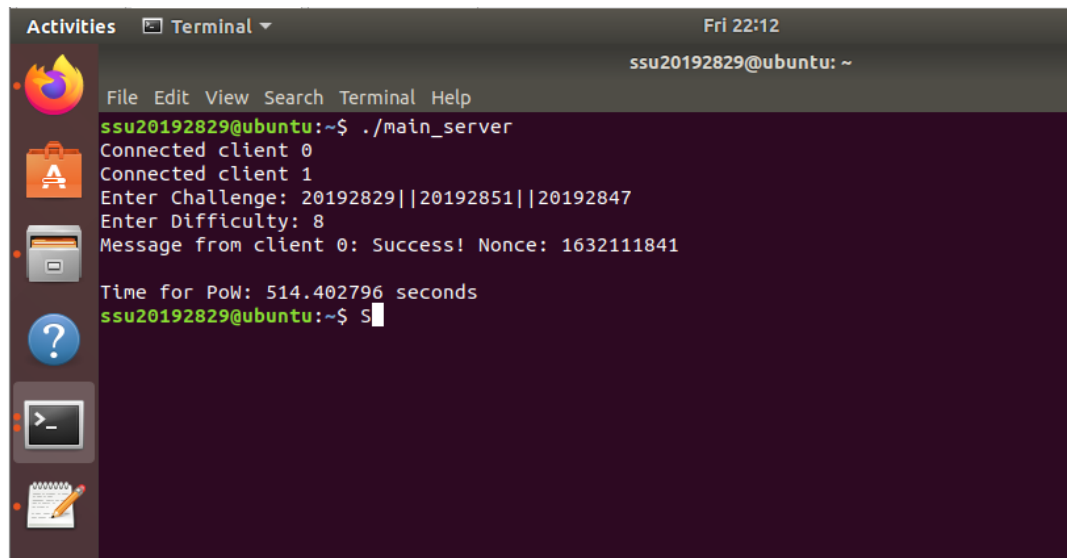
3.7.1. INPUT

- Challenge : 20192829||20192851||20192847
- 업무분담 - Working Server 1 : 0
- 업무분담 - Working Server 2 : 4,611,686,018,427,387,903

- 난이도 : 8

3.7.2. OUTPUT

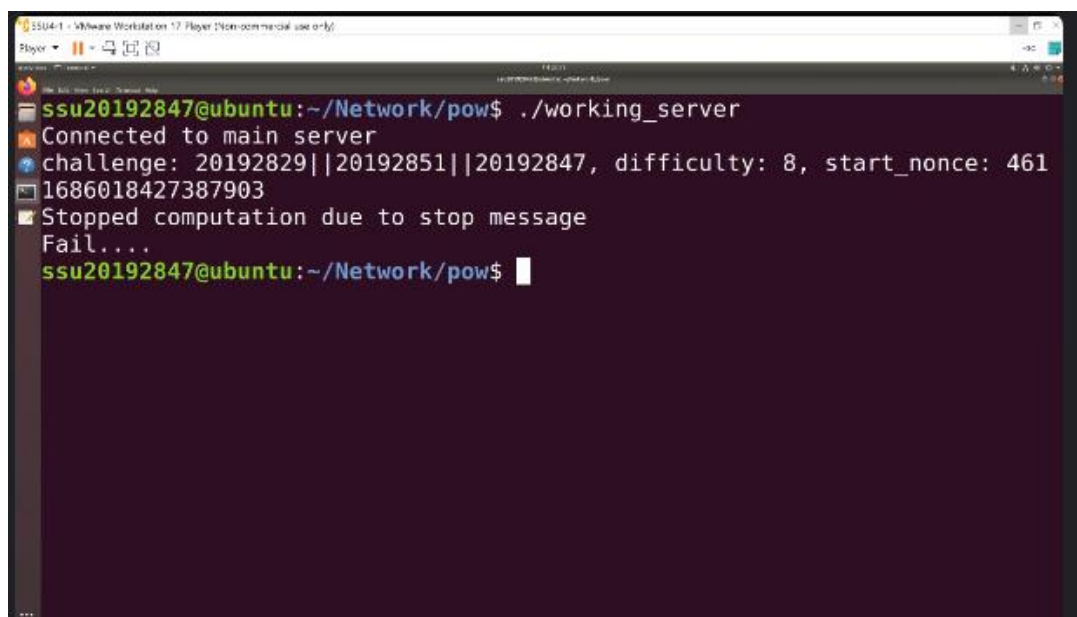
- nonce: 1632111841
- time: 514.402796초



A terminal window titled 'Terminal' with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Fri 22:12, ssu20192829@ubuntu: ~). The terminal shows the following output:

```
ssu20192829@ubuntu:~$ ./main_server
Connected client 0
Connected client 1
Enter Challenge: 20192829||20192851||20192847
Enter Difficulty: 8
Message from client 0: Success! Nonce: 1632111841

Time for PoW: 514.402796 seconds
ssu20192829@ubuntu:~$ S
```



A terminal window titled 'SSU4-1 - VMware Workstation 17 Player (Non-commercial use only)' with a menu bar (File, Edit, View, Window, Help) and a status bar (Fri 22:12, ssu20192829@ubuntu: ~). The terminal shows the following output:

```
ssu20192847@ubuntu:~/Network/pow$ ./working_server
Connected to main server
challenge: 20192829||20192851||20192847, difficulty: 8, start_nonce: 461
1686018427387903
Stopped computation due to stop message
Fail....
ssu20192847@ubuntu:~/Network/pow$
```

```
ssu20192847@ubuntu:~/Network/pow$ ./working_server
Connected to main server
challenge: 20192829||20192851||20192847, difficulty: 8, start_nonce: 461
1686018427387903
Stopped computation due to stop message
Fail....
ssu20192847@ubuntu:~/Network/pow$
```

```
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./working_server3
Connected to main server
challenge: 20192829||20192851||20192847, difficulty: 8, start_nonce: 0
000000000a8ad40cde16b79c522a92b7ebd38cbf55fd9d085006bf26444c4c1e
Thread #0 found a valid nonce: 1632111841
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

3.8. 챌린지 이름 분산 실행 (난이도 8)

3.8.1. INPUT

- Challenge : 양조은||박민수||윤성준
- 업무분담 - Working Server 1 : 0 ~ 4,611,686,018,427,387,902
- 업무분담 - Working Server 2 : 4,611,686,018,427,387,903 ~
- 난이도 : 8

3.8.2. OUTPUT

- nonce: 4611686023899175448
- time: 1882.744854초 = 31.3790809분

```
ssu20192829@ubuntu: ~  
File Edit View Search Terminal Help  
ssu20192829@ubuntu:~$ ./main_server  
Connected client 0  
Connected client 1  
Enter Challenge: 양조은||박민수||윤성준  
Enter Difficulty: 8  
Message from client 1: Success! Nonce: 4611686023899175448  
  
Time for PoW: 1882.744854 seconds  
ssu20192829@ubuntu:~$
```

```
ssu20192851@ubuntu: ~/Desktop/zzProject/pow1  
File Edit View Search Terminal Help  
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$ ./working_server3  
Connected to main server  
challenge: 양조은||박민수||윤성준, difficulty: 8, start_nonce: 4611686018427387903  
000000005cd6209950566b869055e51bde30590dd6e1ff5fa6826ec8b588098c  
Thread #4 found a valid nonce: 4611686023899175448  
ssu20192851@ubuntu:~/Desktop/zzProject/pow1$
```

4. 단독 실행 vs. 분산 실행

아래 4.1, 4.2의 표는 각 챌린지, 난이도에 따른 단독 실행과 분산 실행의 실행시간을 보여준다. "단독 실행"은 하나의 working server만 사용하고 "분산 실행"은 여러 working server를 사용하는 시나리오를 나타낸다.

4.1. 난이도 7일 때의 실행시간 차이

난이도 7	단독 실행(초)	분산 실행(초)	실행 시간 비율 (분산/단독)
학번	239.78	92.63	0.39
이름	18.20	12.60	0.69

학번 챌린지에서는 단독 실행 시 239.78초가 걸리는 반면, 분산 실행에서는 92.63초로 상당히 단축된 시간이 소요되었다. 실행 시간 비율은 0.39로, 분산 실행이 단독 실행에 비해 약 61% 단축하였다.

반면에 이름 챌린지에서는 단독 실행이 18.20초, 분산 실행이 12.60초로 이 경우에서도 분산 실행이 더 효율적이고 단독 실행에 비해 약 31% 단축하였다.

4.2. 난이도 8일 때의 실행시간 차이

난이도 8	단독 실행(초)	분산 실행(초)	실행 시간 비율 (분산/단독)
학번	522.89	514.40	0.98
이름	3242.58	1882.74	0.58

학번 챌린지에서는 단독 실행 시 522.89초, 분산 실행 시 514.40초로 거의 비슷한 시간이 소

요되었다. 실행 시간 비율은 0.98로 거의 동일하다.

이름 챌린지에서는 단독 실행 시간이 3242.58초, 분산 실행 시간이 1882.74초로 분산 실행이 단독 실행에 비해 시간을 많이 줄였다. 실행 시간 비율은 0.58로, 분산 실행이 단독 실행에 비해 약 42%의 시간을 절약하였다.

결론적으로 비슷한 환경이라는 가정 하에서 분산 실행이 일반적으로 더 빠르다. 하지만 하드웨어 성능의 차이 등으로 무조건 분산 실행이 더 빠른 결과가 나오지는 않았다. 예를 들어 난이도 8 과 같은 경우 똑같은 nonce값을 같은 working server가 찾기 때문에 분산실행과 단독실행의 실행 시간 비율이 1에 가깝다.

5. Port Forwarding

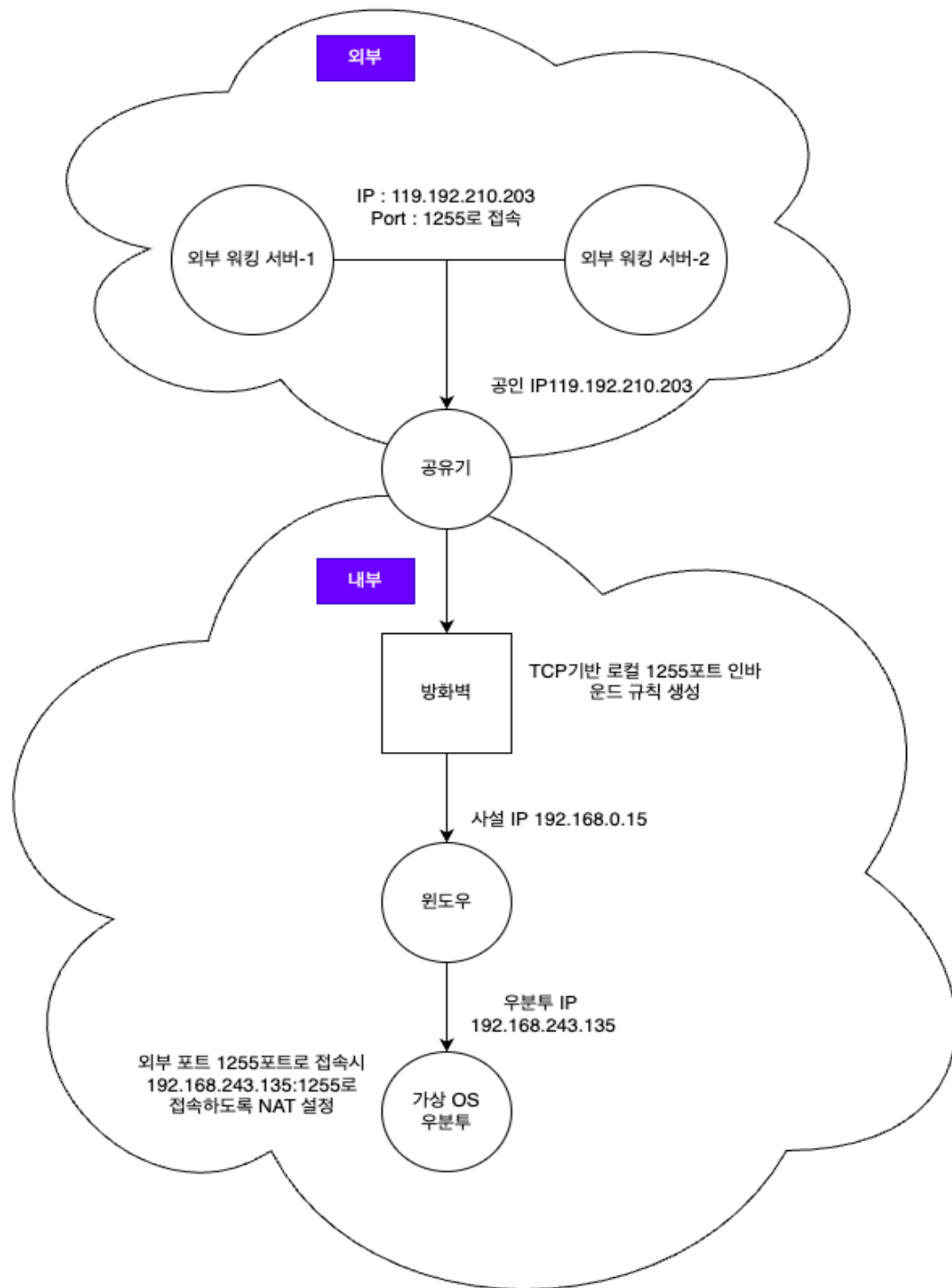


그림5.1. Port Forwarding Diagram

해당 그림5.1을 보면 외부 네트워크 영역에 있는 워킹 서버 두 대가 공인 IP 119.192.210.203과 1255 포트 번호로 접속하게 된다. 그러면, 그림5.2 공유기 포트 포워드 설정에서 ‘NAT/라우터 관리’ -> ‘포트 포워드 설정’ 카테고리에서 외부 포트 1255으로 들어오면 그림 5.3 윈도우 내부

IP주소로 AP로부터 할당받은 IP, 192.168.0.15와 1255 Port로 포트 포워딩이 되도록 설정한다. 그리고 그 포트가 윈도우 내부까지 들어올 수 있도록, 그림5.4 방화벽 인바운드 포트 규칙으로 TCP 기반으로 특정 로컬 포트인 1255포트를 허용하도록 한다. 그림5.5 가상 OS – Network Adapter 에서 우분투 OS 가상 머신을 돌리고 있는 프로그램 ‘VMware’ 에 해당 가상 머신의 Network Adapter의 Network connection을 Custom 및 NAT을 지원하는 VMnet8(NAT)으로 설정한다. 그다음, 그림5.6. VMware NAT Settings를 보면 Host Port인 1255포트로 접근한다면, 현재 실행 중인 Virtual Machine IP address로 192.168.243.135, Port 1255로 NAT이 되도록 설정하였다. 이에 따른 결과로 그림5.7 VMware Virtual Network Editor가 설정된 것을 볼 수 있고, 최종적으로 외부 Working_server 두 대가 1255포트로 접근 시 내부에 실행 중인 가상 머신 우분투 OS의 IP, 192.168.243.135 및 1255 Port로 포트 포워딩 되어서 Main_Server 소켓에 붙도록 설정하였다.

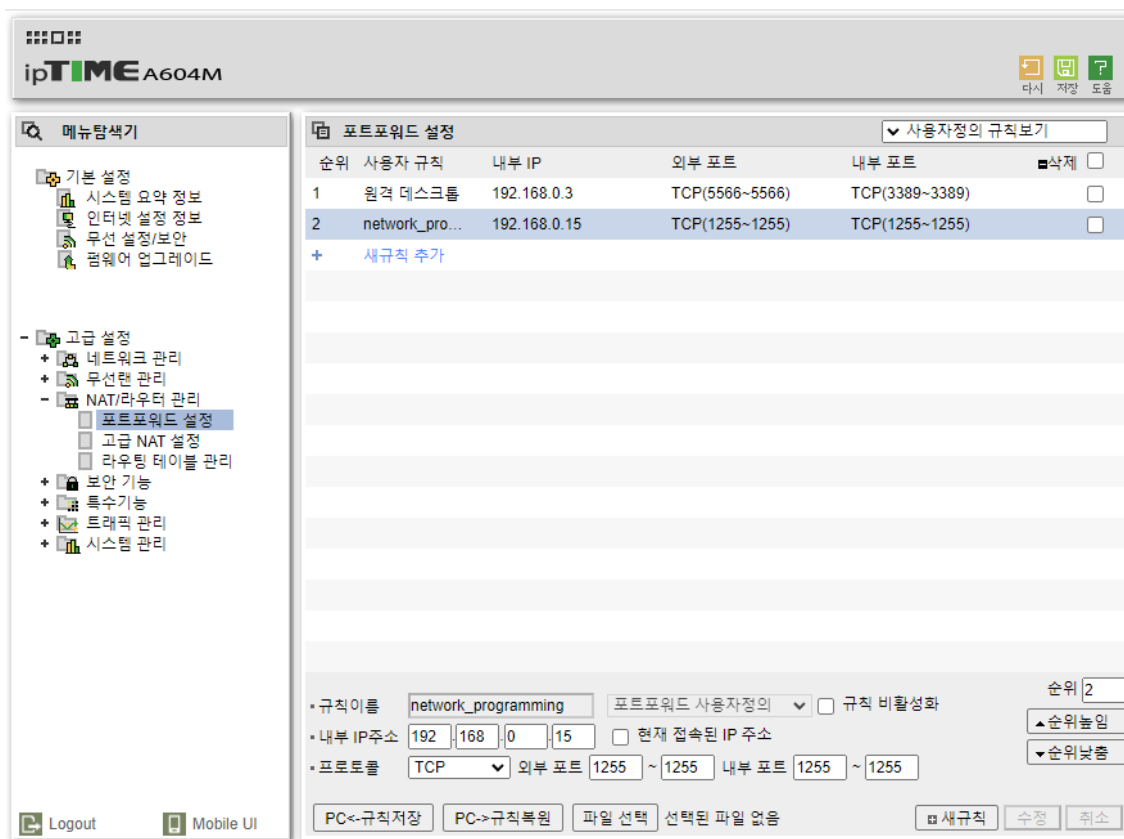


그림5.2. 공유기 포트포워드 설정

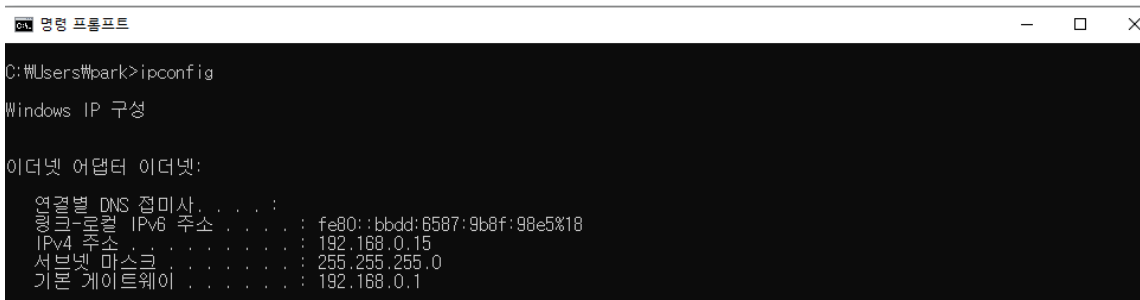


그림5.3. 윈도우 내부 IP주소

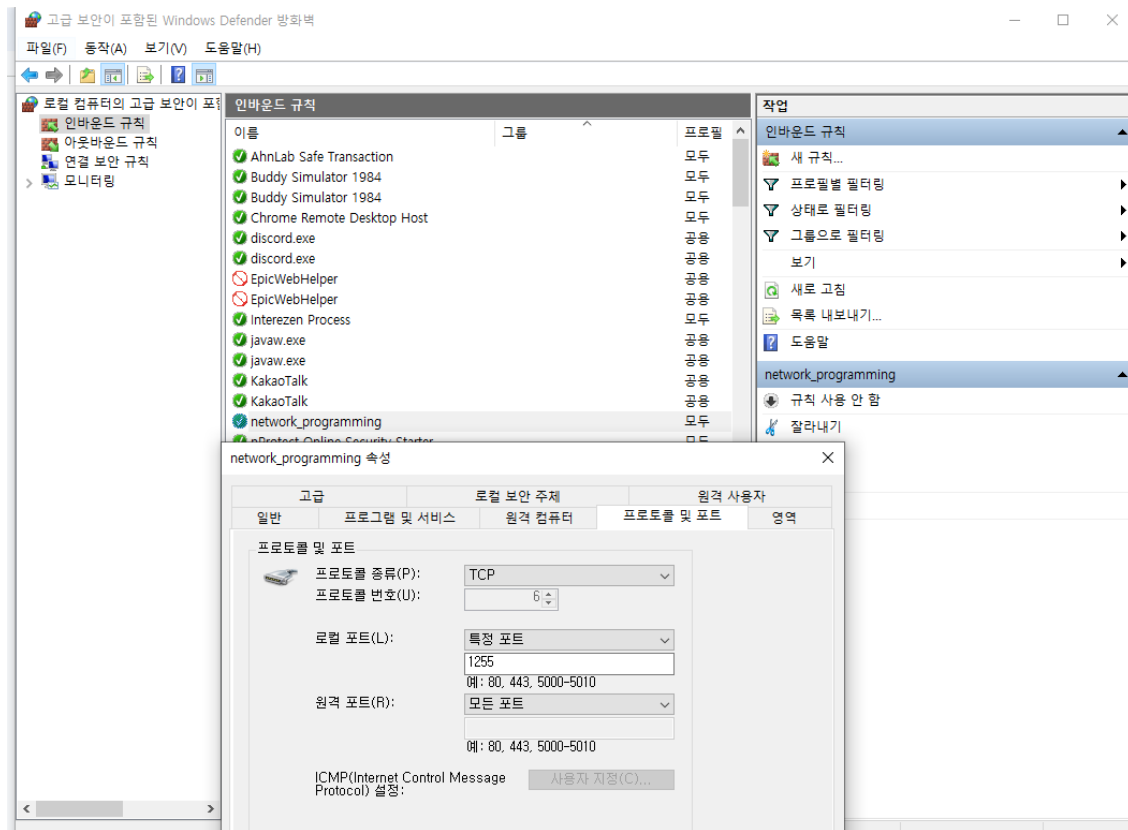


그림5.4. 방화벽 인바운드 포트 규칙

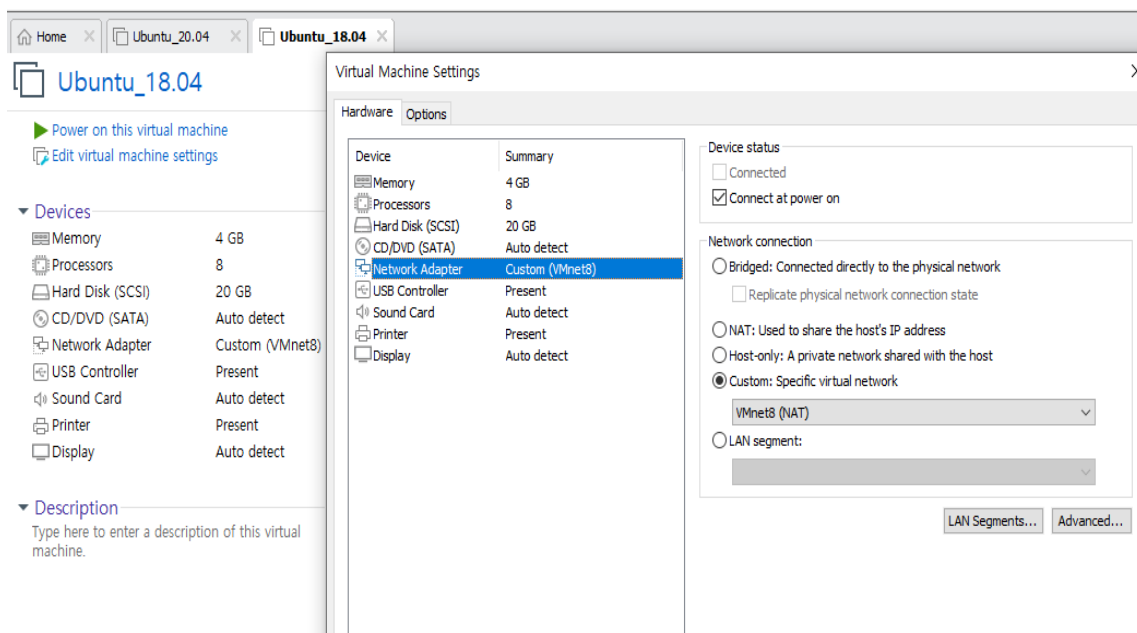


그림5.5. 가상 OS – Network Adapter

NAT Settings ✕

Network: vmnet8
Subnet IP: 192.168.243.0
Subnet mask: 255.255.255.0
Gateway IP:

Port Forwarding

Host Port	Type	Virtual Machine IP Address	Description
1255	TCP	192.168.243.135:1255	network_programming

Add... Remove Properties

Advanced

☒ Allow active FTP
☒ Allow any Organizationally Unique Identifier
UDP timeout (in seconds):
Config port:
☐ Enable IPv6
IPv6 prefix:

DNS Settings... NetBIOS Settings...

OK Cancel Help

그림5.6. VMware NAT Settings

Virtual Network Editor ✕

Name	Type	External Connection	Host Connection	DHCP	Subnet Address
VMnet1	Host-only	-	Connected	Enabled	192.168.12.0
VMnet8	NAT	NAT	Connected	Enabled	192.168.243.0

Add Network... Remove Network Rename Network...

그림5.7. VMware Virtual Network Editor

6. 프로젝트 회고

프로젝트를 작성하며, 어려웠던 점으로 초기 프로그램 작성 시 MAX_nonce 값을 정해서 프로그램을 수행했다. 그러나 생각한 것보다 난이도가 올라갈수록 우리가 설정한 MAX_nonce 값의 범위를 훨씬 초과했다. 그래서 MAX_nonce 값을 제거하여 각 Working_Server는 무한대를 향해 nonce 값을 찾도록 수행하는 코드를 바꿨다. 또한, 업무 분담 명세 내용을 진행하며 Main_Server에서 Working_Server마다 시작 탐색 nonce 값을 보내줄 때 1번 Working_Server는 0부터 시작하기 때문에 문제가 없지만, 2번 Working_Server에 보내 줄 시작 탐색 nonce 값을 정하는 과정에서 최적의 시작 탐색 nonce 값을 정하는 것과 각 Working_Server마다 5개의 Threads가 한 번에 Thread가 담당할 nonce 값 범위를 몇으로 정해야지 훨씬 더 빠르게 찾을 수 있을지 고민하는 과정에서 많은 시간을 소비했었다. 그리고 프로그램 작성을 끝내 프로그램 실행 중에 정답과 'print_hash()' 함수를 통해 해시값과 정답이 뒤섞여서 나왔었다. 이 문제를 해결하려 노력하던 중, 'print_hash()' 함수의 수행 속도가 너무 느리다고 생각해, 기존 'print_hash()' 함수는 'if (is_valid(hash, difficulty))' 에서 IF 문 바깥에 위치하여 정답을 찾았을 때뿐만 아니라 항상 해당 함수를 호출하도록 했으나, IF 문 안쪽으로 바뀌서 정답을 찾았을 때만 'print_hash()' 함수를 실행되도록 변경하여 해결했다. 해당 지정 주제 프로젝트를 수행하면서 블록체인 관련 주제와 한 학기 동안 배운 소켓 프로그래밍을 연관 지어 프로젝트를 해볼 기회가 있어 좋았다.