

서버 사용법 및 스케줄러를 사용한 ORCA 실행법

전남대학교 화학과

컴퓨터화학 연구실

최종 수정: 2023년 12월 24일

1. 서버 접속

연구실 서버에 접속하기 위해서는 터미널을 켜 후 다음과 같이 입력한다(WSL을 꼭 실행할 필요는 없음).

```
ssh -p XXXX <ID>@168.YYY.YYY.YYY
```

여기에서 <ID>에는 자신의 실제 ID를 넣어야 하며, -p 다음에 오는 포트 번호와 168로 시작하는 서버 IP 주소는 보안 문제로 인하여 여기 기재하지 않고 별도로 안내된다. 예를 들어 자신의 ID가 'abc'이고, 포트 번호가 5555, IP 주소가 168.0.0.1이라면 다음과 같이 입력하여 접속할 수 있다.

```
ssh -p 5555 abc@168.0.0.1
```

괄호 <>는 제거해야 한다는 사실에 유의하자. 첫 접속 시에는 yes/no를 물어보는 질문이 나오는데 **yes**를 선택한다.

그 다음에는 비밀번호를 입력하게 되는데, 서버 관리자는 초기 비밀번호를 **1**로 설정한다. 접속한 후 **passwd**를 입력하여 비밀번호를 변경할 수 있다.

경고: 꼭 비밀번호를 변경할 것. 변경하지 않고 1로 놓아 두는 경우에는 서버 관리자가 가끔 예고 없이 파일을 전부 삭제시킬 것이다.

서버에 접속하면 맨 처음 나오는 폴더는 개인 기본 폴더인 **/home/<ID>** 이다. 개인 컴퓨터에서와 마찬가지로 **.bashrc** 파일을 수정하여 환경 설정을 할 수 있다 ('WSL 설치 및 사용 매뉴얼' 참조).

2. 서버/개인 컴퓨터 간 파일 이동

서버와 개인 컴퓨터 사이에 파일을 원격으로 주고받기 위해서는 **scp**를 사용한다. 문법은 기본적으로 cp와 비슷하지만, 서버의 주소를 입력하고 아이디와 비밀번호를 사용하여 통신 권한을 얻어야 한다는 약간의 차이가 있다.

예제 1: 서버의 개인 기본 폴더에 있는 bcd.txt를 개인 컴퓨터의 현재 위치로 복사

```
scp -P XXXX <ID>@168.YYY.YYY.YYY:/home/<ID>/bcd.txt .
```

예제 2: 개인 컴퓨터 WSL의 /home/guest 에 있는 cde.txt를 서버의 개인 기본 폴더로 복사

```
scp -P XXXX /home/guest/cde.txt <ID>@168.YYY.YYY.YYY:/home/<ID>
```

현재 폴더의 경로는 **pwd** 명령어를 입력하여 확인할 수 있다.

주의: -P 뒤에 위치하는 포트 번호와 IP 주소는 별도로 안내받은 값으로 바꾸어야 한다. 또한, 서버와 파일을 주고받을 경우 실제로 파일 및 폴더가 그 위치에 존재하고 있어야 할 뿐 아니라, 자신이 읽기/쓰기를 할 권한도 가지고 있어야 한다. 별다른 일이 없으면 개인 기본 폴더 아래에 위치한 폴더나 파일들에 대한 권한은 가지고 있다고 생각해도 좋다.

3. 서버에서 ORCA 작업 실행하기

서버로 사용하는 슈퍼컴퓨터 클러스터는 여러 대의 컴퓨터를 네트워크로 묶어 통신하게 만드는 방식으로 구축하며, 각각의 컴퓨터를 node라고 부른다. node들은 컨트롤 타워 역할을 수행하는 master(main) node와 실제 계산을 담당하는 slave node로 구분된다. 물리화학실험/컴퓨터화학 등의 강의에서 진행된 실습에서는 교수의 통제 하에 master node에서 계산을 수행하였지만, 많은 인원이 무질서하게 master node에서 작업을 하게 될 경우 master node가 자체적으로 보유한 계산 자원을 초과하게 되어 서버에 무리가 갈 수 있다.

이러한 문제 때문에, 많은 계산이 필요한 학부 인턴, 졸업 논문 작성 및 대학원 연구의 경우 slave node에 계산을 제출하며, 이러한 계산들의 경우 작업에 우선 순위를 매겨 교통 정리를 하는 스케줄러의 관리를 받게 된다. 본 연구실에서는 스케줄러로 slurm을 사용한다.

2023년 말 현재 연구실 서버에는 총 5대의 slave node가 있으며, node1~node5까지의 이름이 붙어 있다. node1과 node2는 각각 20개의 CPU core를 보유하며, node3~node5는 각각 40개의 CPU core를 보유한다. 학부생의 경우 node1과 node2에만 사용 권한이 허락되며, 대학원생의 경우 모든 node를 다 사용할 수 있다.

Slave node에 ORCA 작업을 제출하는 방법을 연습하기 위하여, 홈페이지에 있는 예제인 h2.tar를 다운받아 서버에 복사하고, 다음과 같이 입력하여 압축을 해제한다.

```
tar -xvf h2.tar
```

압축을 풀면 h2라는 이름의 폴더가 생성되며, h2.in과 h2.sh라는 파일이 들어 있다. h2.in은 수소 분자를 계산하는 ORCA input이다.

```
| B3LYP SP def2-TZVP
%PAL NPROCS 2
END

* xyz 0 1
  H      0.00000      0.00000      0.00000
  H      0.74000      0.00000      0.00000
*
```

%PAL NPROCS 다음에는 계산에 사용할 CPU core의 개수를 적는다. core를 많이 넣으면 계산 속도가 증가하며, 초심자의 경우 1~4 정도로 유지하기를 권장한다.

h2.sh는 작업 제출에 사용되는 신청서로 다음과 같이 작성된다.

```
#!/bin/sh
#SBATCH -J h2
#SBATCH -p silver
#SBATCH -N 1
#SBATCH -n 2
#SBATCH --time 48:00:00
#SBATCH -o %x.o%j
#SBATCH -e %x.e%j
ID=
JOBNAME=h2
```

이 밑에도 추가 내용이 있으나, 자기가 무슨 일을 하고 있는지에 대한 확신이 없으면 변경할 필요 없이 그대로 유지한다. 초심자의 경우 다음 부분들만 변경하면 된다.

`#SBATCH -J h2`: 서버에서 표시되는 작업의 이름이다. 혼동을 피하기 위해 ORCA input의 이름과 동일한 h2로 적는다.

`#SBATCH -n 2`: 사용할 CPU의 개수이며, ORCA input 파일에 적은 값과 같은 값인 2로 맞추었다. 두 값이 다를 경우 스케줄러에 혼란이 올 수 있으므로 거듭 확인해야 한다.

`ID=`: 자신의 ID를 (띄어쓰기 없이) 적는다.

`JOBNAME=h2`: ORCA input 파일의 이름과 맞춘다. h2.in을 계산하고자 하므로 h2로 적는다.

불필요한 띄어쓰기가 들어가거나, 신청서 내용의 순서를 임의로 바꾸면 작업 신청에서 오류가 발생할 수 있으므로 주의한다.

스케줄러에 작업을 제출하기 위해서는 `sbatch` 명령어를 사용하며, `sbatch h2.sh`와 같이 제출하고자 하는 신청서의 이름을 뒤에 적어 실행한다.

```
@qdtsv:~/h2$ sbatch h2.sh
Submitted batch job 11138
```

작업을 제출하면 작업에는 숫자로 된 번호표가 붙는다(이 경우에는 11138번). 제출된 작업들의 목록을 보기 위해서는 `squeue` 명령어를 입력한다.

```
@qdtsv:~/h2$ squeue
JOBID PARTITION NAME USER STATE TIME TIME_LIMIT NODES CPUS NODELIST(REASON)
11138 silver h2 h2 RUNNING 0:00 2-00:00:00 1 2 node1
```

STATE가 'RUNNING'일 경우 slave node에서 작업이 수행되고 있는 것이며, 'PENDING'일 경우 서버에 빈 CPU가 없어 대기 중이라는 의미이다. NODELIST 부분을 보면 제출된 작업이 node1에서 수행되고 있다는 사실도 알 수 있다.

계산 결과는 (h2.out이 아니라) 자동 생성되는 파일인 `h2.o<작업 번호>`에 저장되도록 세팅되어 있으며, 이 경우는 작업 번호가 11138이므로 생성되는 파일은 `h2.o11138`이다. 계산 결과가 끝난 후 실제로 파일을 열어 확인해 보자. 작업 번호는 매번 달라진다는 사실에 유의한다.

주의: 학부생의 경우, 한 사람이 동시에 16개가 넘는 CPU를 사용하는 것은 불가피한 경우 이외에는 자제하도록 한다.

경고: 본 매뉴얼의 내용을 읽었음에도 불구하고 불필요하게 master node에서 작업을 수행하는 것이 적발될 경우 ID가 며칠간 정지될 수 있음.