

# SSD 볼륨 서비스 가이드

2015.03

## 개정이력

개정일자	Version	개정사유	개정내용
2015. 3. 12	1.0	신규 작성	

# 목 차

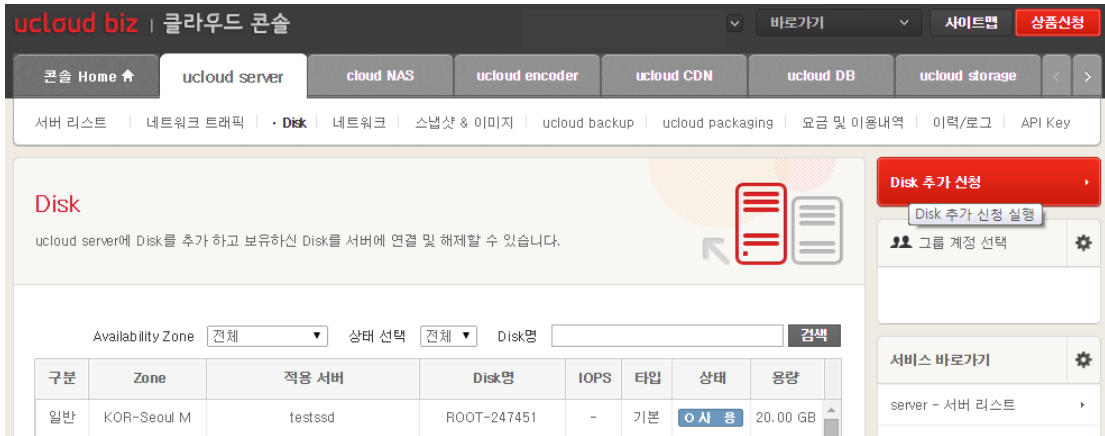
<b>1. SSD 볼륨 서비스 .....</b>	<b>4</b>
1.1 상품 신청 방법.....	4
1.2 OS 디스크 초기화.....	5
<b>2. 디스크 성능 측정.....</b>	<b>6</b>
2.1 LINUX.....	6
2.2 WINDOWS.....	6

# 1. SSD 볼륨 서비스

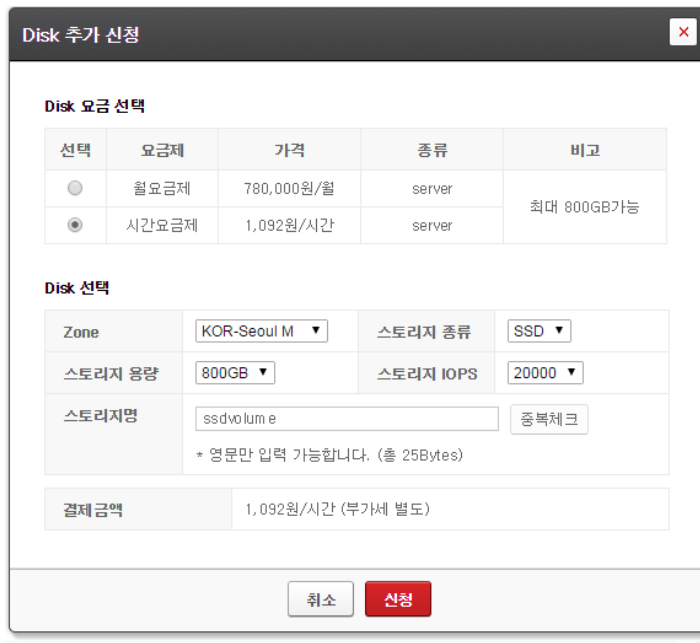
- SSD 볼륨은 300GB ~ 800GB 의 용량 및 6,000 IOPS ~ 20,000 IOPS 의 성능 제공이 가능합니다. (4KB I/O 기준, 볼륨 생성 후 용량 및 IOPS 변경 불가)
- Seoul-M Zone 만 제공 가능 합니다. (2015. 3 기준, Central A/B, HA Zone 은 추후 지원 예정)
- 4 vCore 이상의 VM 만 볼륨 연결이 가능합니다. (VM 당 최대 2 개 SSD 볼륨 연결 가능)
- VM 에서 발생 시킬 수 있는 최대 IOPS 는 20,000 입니다.
- 안정적인 성능 제공을 위해 Raid 구성은 권장하지 않습니다.  
Ex) 12,000 IOPS 를 위해 단일 볼륨 생성 (O), 6,000 IOPS 볼륨 2 개 Raid 0 로 구성 (X)

## 1.1 상품 신청 방법

- 클라우드 콘솔 > ucloud server > Disk > 오른쪽 "Disk 추가 신청"



- Disk 요금제 선택 > Zone 선택 > 스토리지 종류 "SSD" 선택
- 용량 및 IOPS 선택 > 스토리지 명 입력 후 중복 검사 > "신청"



- 생성된 볼륨 선택 > 하단 “서버 연결” 클릭

**Disk**

ucloud server에 Disk를 추가 하고 보유하신 Disk를 서버에 연결 및 해제할 수 있습니다.

Availability Zone  상태 선택  Disk명

구분	Zone	적용 서버	Disk명	IOPS	타입	상태	용량
SSD	KOR-Seoul M	--	ssdvolum e	20000	추가	×분 리	800.00 GB

- 연결 할 서버 선택 > “연결하기” 클릭

**서버 연결**

ssdvolum e스토리지에 연결할 클라우드 서버를 선택해주세요.

서버명

**\*SSD disk는 4vCore 이상 사양의 서버에서만 사용 가능합니다.\***

- 연결 후 화면

Availability Zone  상태 선택  Disk명

구분	Zone	적용 서버	Disk명	IOPS	타입	상태	용량
SSD	KOR-Seoul M	testssd	ssdvolum e	20000	추가	○사 용	800.00 GB

## 1.2 OS 디스크 초기화

- Ucloud server User Guide, p18 참조
- Download url : [https://ucloudbiz.olleh.com/manual/ucloud\\_server\\_UserManual\\_v4.1.pdf](https://ucloudbiz.olleh.com/manual/ucloud_server_UserManual_v4.1.pdf)

## 2. 디스크 성능 측정

### 2.1 Linux

- **FIO** : I/O 측정을 위한 툴 (libaio-devel 의존성을 가짐)
  - 설치 법 : Ubuntu 는 "apt-get install fio" 로 설치 가능

```
yum install libaio* gcc wget make
wget http://brick.kernel.dk/snaps/fio-2.2.6.tar.gz
gunzip fio-2.2.6.tar.gz
tar -xf fio-2.2.6.tar
cd fio-2.2.6
./configure
make
make install
```

- 사용방법 : Fio 설치 후 아래 command 실행

- ◆ Random Read

```
fio --directory=[마운트한 폴더명] --name [파일 이름] --direct=1 --rw=randread --bs=4k --size=1G --numjobs=16 --time_based --runtime=180 --group_reporting --norandommap
```

- ◆ Random Write

```
fio --directory=[마운트한 폴더명] --name [파일 이름] --direct=1 --rw=randwrite --bs=4k --size=1G --numjobs=16 --time_based --runtime=180 --group_reporting --norandommap
```

- 실행 화면

```
[root@testssd ssd]# fio --directory=/ssd/ --name fio_test_file --direct=1 --rw=randread --bs=4k --size=1G --numjobs=16 --time_based --runtime=180 --group_reporting --norandommap
fio_test_file: (g=0): rw=randread, bs=4K-4K/4K-4K/4K-4K, ioengine=sync, iodepth=1
...
fio_test_file: (g=0): rw=randread, bs=4K-4K/4K-4K/4K-4K, ioengine=sync, iodepth=1
fio-2.0.14
Starting 16 processes
|obs: 16 (f=16): [rrrrrrrrrrrrrrrrrr] [17.7% done] [80664K/0K/0K /s] [20.2K/ ] /0 iops] [eta 02m:29s]
```

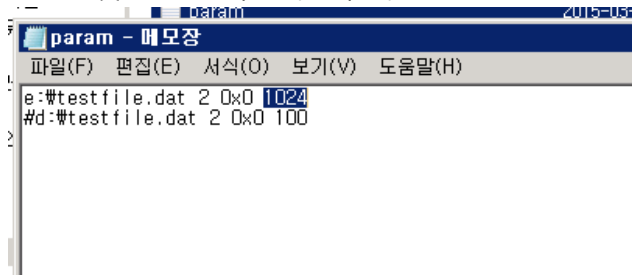
### 2.2 Windows

- SQLIO : Microsoft 에서 제공하는 I/O 측정 툴
- Download url : <http://www.microsoft.com/en-us/download/details.aspx?id=20163>
  - 설치 법 : 다운로드 파일 실행(SQLIO.msi) 후 install
  - 사용방법
    - ◆ 설치 폴더 이동



◆ param.txt 열기

◆ file 경로 및 file size 수정 (단위 M) : 100 > 1024



◆ 설치 폴더에서 아래 command 실행

◆ Random Read

```
sqlio.exe -kR -t8 -o8 -s120 -frandom -b4 -LS -Fparam.txt
```

◆ Random Write

```
sqlio.exe -kW -t8 -o8 -s120 -frandom -b4 -LS -Fparam.txt
```

## ■ 실행 화면

```

C:\Program Files (x86)\WSQLIO>sqlio.exe -kR -t8 -o8 -s120 -frandom -b4 -LS -Fparam.txt
sqlio v1.5.SG
using system counter for latency timings, 62500000 counts per second
parameter file used: param.txt
    file e:\testfile.dat with 2 threads (0-1) using mask 0x0 (0)
2 threads reading for 120 secs from file e:\testfile.dat
    using 4KB random IOs
    enabling multiple I/Os per thread with 8 outstanding
size of file e:\testfile.dat needs to be: 2147483648 bytes
current file size:      1073741824 bytes
need to expand by:     1073741824 bytes
expanding e:\testfile.dat ... done.
using specified size: 2048 MB for file: e:\testfile.dat
initialization done
CUMULATIVE DATA:
throughput metrics:
IOs/sec: 20088.64
MBs/sec:    78.47
latency metrics:
Min_Latency(ms): 0
Avg_Latency(ms): 0
Max_Latency(ms): 24
histogram:
ms: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24+
%: 91 8 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
  
```