

kt cloud

Basic Course Hands on Lab

(3) Storage

Copyright© 2023 kt cloud corp. All rights reserved.

실습 1. 가상서버의 볼륨 관리(LVM)

01

볼륨생성 및 볼륨 추가

02

파티셔닝과 LVM구성

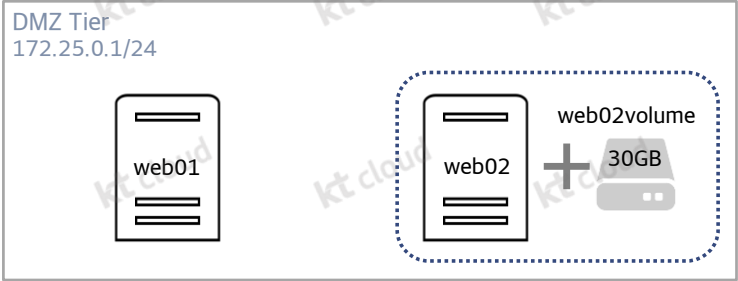
03

볼륨 사용

실습 1에서는 리눅스서버에 볼륨을 추가해 봅니다.
새 볼륨은 블록 스토리지에 해당되며, 파일시스템으로 포맷 후 사용할 수 있습니다.
추가된 볼륨이 사용가능하도록 파티션을 생성하고 포맷을 진행합니다.

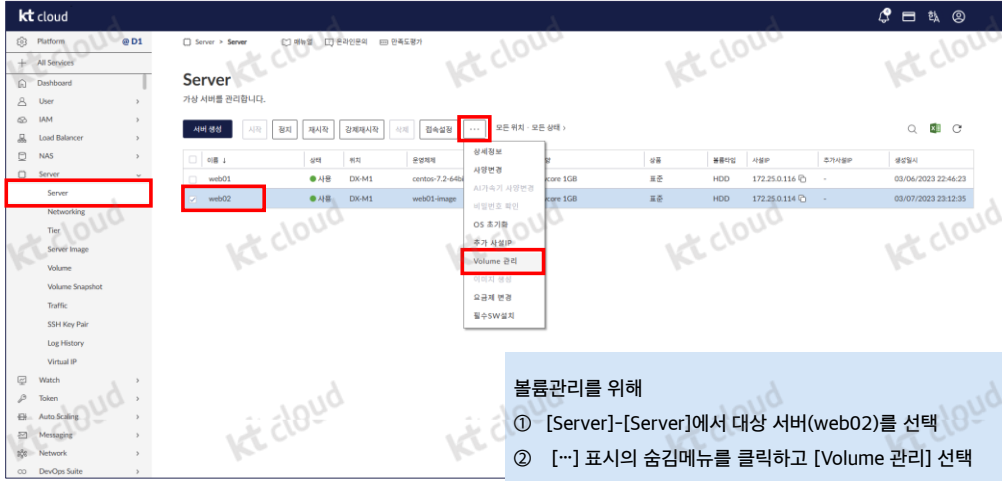
또한 볼륨 연결과 해제를 실습하겠습니다.

실습 1. 개요



Copyright© 2023 kt cloud corp. All rights reserved.

1-1. Volume관리 메뉴



kt cloud

Platform @ D1

Server > Server

가장 서버를 관리합니다.

서버 생성 | 시작 | 중지 | 재시작 | 강제재시작 | 삭제 | 연속실행 | 모든 위치 모든 상태 >

이름	상태	타입	변지	운영체제	시작방법	스케줄	용량	용량단위	사양CPU	초기사양IP	생성일시
web01	사용	DK-M1	centos-7.2-64bit	서버	자동	10GB	10GB	HDD	172.25.0.114	-	03/06/2023 22:44:23
web02	사용	DK-M1	web01-image	서버	자동	10GB	10GB	HDD	172.25.0.114	-	03/07/2023 23:12:35

서버정보
서버변경
서버삭제
OS 호환성
추가 설정
Volume 관리
이전 설정
요금제 변경
필수SW설치

볼륨관리를 위해

- ① [Server]-[Server]에서 대상 서버(web02)를 선택
- ② [...] 표시의 숨김메뉴를 클릭하고 [Volume 관리] 선택

kt cloud 에서는 원하는 용량의 데이터 볼륨을 생성하고, 이를 서버에 연결하여 저장 공간을 확장할 수 있습니다.
HDD 기반의 볼륨을 지원하며, 볼륨은 10GB 단위로 생성이 가능합니다.

1-2. 새 볼륨추가

web02 의 볼륨을 관리합니다.

연결해제

스냅샷

이름	타입	상품	용량
<input type="checkbox"/> c94a2fa0-0bb4-4077-ba7b-5c79afd0c714	Root	HDD	50 GB

+

 이 서버에 Volume 추가

- ① “+ 이 서버에 Volume 추가” 클릭
- ② 이름, 상품, 용량, 요금제 설정
- ③ [생성하기] 클릭

- web02 서버의 볼륨 추가 사양
- 볼륨이름 : web02vol1
 - 상품 : HDD
 - 용량 : 30GB
 - 월 요금제

web02 의 볼륨을 관리합니다.

연결해제

스냅샷

이름	타입	상품	용량
<input type="checkbox"/> 3928b400-96ca-41c7-8351-1d6651223604	Root	HDD	50 GB

web02vol1

DATA

HDD

30

GB

-

IOPS

월 요금제

시간 요금제

2,100원/월

+ 추가

상품은 HDD와 SSD를 선택할 수 있습니다.
용량은 10GB 단위로 설정가능 합니다.

예제는 web02 서버의 볼륨을 다음 사양으로 추가합니다.

- 볼륨이름 : web02volume
- 상품 : HDD
- 용량 : 30GB
- 월요금제

1-3. 볼륨확인

Volume

가상 서버의 볼륨을 관리합니다.

볼륨 생성

연결

연결해제

삭제

...

모든 위치 · 모두 · 추가 >

이름 ↓

위치

적용 서버

상태

타입

상품

용량

생성일시

<input type="checkbox"/>	web02vol1	DX-M1	web02	● 사용	추가	HDD	30 GB	07/23/2023 21:58:15
--------------------------	-----------	-------	-------	------	----	-----	-------	---------------------

Server

가상 서버를 관리합니다.

서버 생성

서버

정지

재시작

강제재시작

삭제

상세정보

...

모든 위치 · 모든 상태 >

이름 ↓

상태

위치

운영체제

사양

상세정보

<input type="checkbox"/>	web01	●	DX-M1	centos-7.2-64bit	1v	사양변경
<input checked="" type="checkbox"/>	web02	●	DX-M1	web01-image	1v	사양변경
<input type="checkbox"/>	web03	●	DX-M1	centos-7.2-64bit	1v	사양변경

상세정보

서버명 web02

hostname web02

서버 ID 5c327d78-70c5-4b33-864f-6a7b1623aa7e

종류 표준

운영체제 web01-image

사양 1 vCore 1 GB

Volume 80GB

상태 ● 사용

위치 DX-M1

요금제

필요요금제 test-my-key

SSH Key Pair

• [Server]-[Volume] 에서 생성된 볼륨 확인

• [Server]-[Server]-[상세정보]

[Server]-[Volume] 에서 생성된 볼륨에 대해 다음 항목을 확인합니다.

- 볼륨이름
- 위치
- 적용서버
- 상태
- 상품
- 용량

Copyright© 2023 kt cloud corp. All rights reserved.

1-4. Volume 상세보기

Volume

가상 서버의 볼륨을 관리합니다.

볼륨 생성

연결

연결해제

삭제

모든 위치 - 모두 >

이름 ↓	위치	상세 정보	상태	타입	상품	용량	IOPS	생성일시
web02volume	DX-M1	상세 정보 크기변경 스냅샷 요금제 변경	사용	추가	HDD	30 GB	-	03/23/2023 17:24:38

볼륨 이름

web02vol1

볼륨 ID

1c91ec55-8389-4806-8340-b3ac18f6faa9

타입

일반

상품

HDD

요금제

월요금제

생성일시

2023-05-22T01:07:55.000000

적용서버

web02

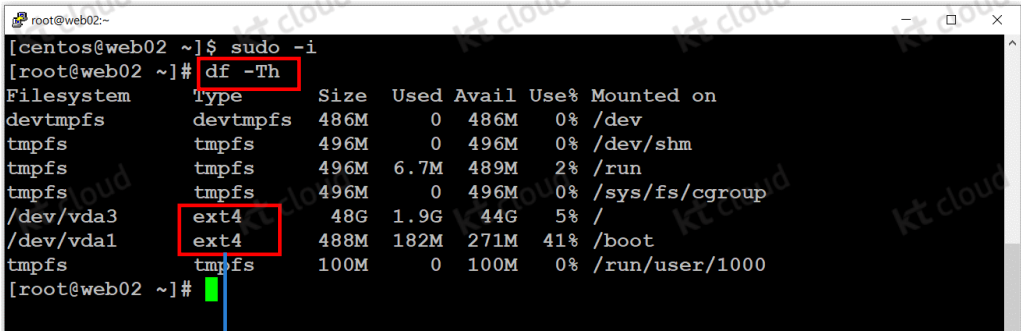
용량

30GB

[Server]-[Volume]-[...]-[상세정보]를 클릭하여 web02 서버에 30GB의 볼륨이 추가된 것을 확인합니다.

또한 [Server]-[Server]-[...]-[상세정보]를 클릭하여 web02 서버의 볼륨 총량을 확인할 수도 있습니다.

1-5. 현재 디스크 상태 확인



```
[centos@web02 ~]$ sudo -i
[root@web02 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
devtmpfs        devtmpfs  486M   0    486M   0% /dev
tmpfs           tmpfs     496M   0    496M   0% /dev/shm
tmpfs           tmpfs     496M  6.7M  489M   2% /run
tmpfs           tmpfs     496M   0    496M   0% /sys/fs/cgroup
/dev/vda3       ext4      48G   1.9G   44G   5% /
/dev/vda1       ext4      488M  182M  271M  41% /boot
tmpfs           tmpfs     100M   0    100M   0% /run/user/1000
[root@web02 ~]#
```

기존 파티션의 파일시스템이 ext4 임을 확인

리눅스 서버의 경우 추가될 volume은 파티션 과정을 거쳐야 합니다.
df -Th 리눅스 명령으로 파티션 작업 전 현재 디스크 용량상태 및 파일시스템을 확인
합니다.

1-6. 추가된 디스크 확인

```
root@web02~  
[root@web02 ~]# fdisk -l  
Disk /dev/vda: 53.7 GB, 53687091200 bytes, 104857600 sectors  
Units = sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk label type: dos  
Disk identifier: 0x000e4cff  
  
   Device Boot      Start         End      Blocks   Id  System  
   /dev/vda1  *        2048     1050623       524288   83   Linux  
   /dev/vda2             3147775     1048576       1048576   82   Linux swap / Solaris  
   /dev/vdb             104857566     50854895+       83   Linux  
  
Disk /dev/vdb: 32.2 GB, 32212254720 bytes, 62914560 sectors  
Units = sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
  
[root@web02 ~]# ls /dev/vd*  
/dev/vda /dev/vda1 /dev/vda2 /dev/vda3 /dev/vdb → 장치 이름 확인
```

디스크 및 파티션 조회 명령어 fdisk -l

하드디스크를 연결해 놓은 상태에서 fdisk 명령어를 통해 파티션을 설정한 후, 포맷으로 파일시스템을 잡아주어야 합니다. 위 예제는 fdisk -l 명령으로 조회한 파티션 리스트입니다. 새로 장착한 하드는 /dev/vdb로 약 30기가바이트의 용량입니다. 시스템에 할당되어 있지 않은 상태입니다.

1-7. 파티션 생성

```
root@web02:~# fdisk /dev/vdb
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x3c941082.

Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-62914559, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-62914559, default 62914559):
Using default value 62914559
Partition 1 of type Linux and of size 30 GiB is set
```

나머지 항목은 기본값사용(엔터)

fdisk를 사용하여 디스크를 하나의 파티션으로 생성합니다.

1-8. 파티션 설정 보기

대문자

```
Command (m for help): P

Disk /dev/vdb: 32.2 GB, 32212254720 bytes, 62914560 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x3c941082

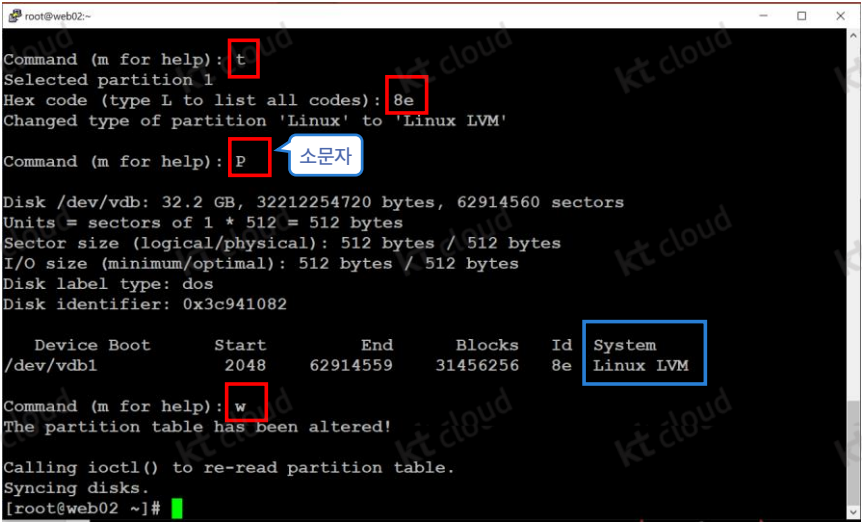
   Device Boot      Start         End      Blocks   Id  System
/dev/vdb1           2048     62914559     31456256   83   Linux
```

1-9. 시스템 ID 보기

```
root@web02:~# Command (m for help): 1 → System01 id 목록 확인
```

0 Empty	24 NEC DOS	81 Minix / old Lin	bf Solaris
1 FAT12	27 Hidden NTFS Win	82 Linux swap / So	c1 DRDOS/sec (FAT-
2 XENIX root	39 Plan 9	83 Linux	c4 DRDOS/sec (FAT-
3 XENIX usr	3c PartitionMagic	84 OS/2 hidden C:	c6 DRDOS/sec (FAT-
4 FAT16 <32M	40 Venix 80286	85 Linux extended	c7 Syrinx
5 Extended	41 PPC PreP Boot	86 NTFS volume set	da Non-FS data
6 FAT16	42 SFS	87 NTFS volume set	db CP/M / CTOS / .
7 HPFS/NTFS/exFAT	4d QNX4.x	88 Linux plaintext	de Dell Utility
8 AIX	4e QNX4.x 2nd part	8e Linux LVM	df BootIt
9 AIX bootable	4f QNX4.x 3rd part	93 Amoeba	e1 DOS access
a OS/2 Boot Manag	50 OnTrack DM	94 Amoeba BBT	e3 DOS R/O
b W95 FAT32	51 OnTrack DM6 Aux	9f BSD/OS	e4 SpeedStor
c W95 FAT32 (LBA)	52 CP/M	a0 IBM Thinkpad hi	eb BeOS fs
e W95 FAT16 (LBA)	53 OnTrack DM6 Aux	a5 FreeBSD	ee GPT
f W95 Ext'd (LBA)	54 OnTrackDM6	a6 OpenBSD	ef EFI (FAT-12/16/
10 OPUS	55 EZ-Drive	a7 NeXTSTEP	f0 Linux/PA-RISC b
11 Hidden FAT12	56 Golden Bow	a8 Darwin UFS	f1 SpeedStor
12 Compaq diagnost	5c Priam Edisk	a9 NetBSD	f4 SpeedStor
14 Hidden FAT16 <3	61 SpeedStor	ab Darwin boot	f2 DOS secondary
16 Hidden FAT16	63 GNU HURD or Sys	af HFS / HFS+	fb VMware VMFS
17 Hidden HPFS/NTF	64 Novell Netware	b7 BSDI fs	fc VMware VMKCORE
18 AST SmartSleep	65 Novell Netware	b8 BSDI swap	fd Linux raid auto

1-10. 파티션의 시스템 ID 변경



```
root@web02:~  
Command (m for help): t  
Selected partition 1  
Hex code (type L to list all codes): 8e  
Changed type of partition 'Linux' to 'Linux LVM'  
  
Command (m for help): P 소문자  
  
Disk /dev/vdb: 32.2 GB, 32212254720 bytes, 62914560 sectors  
Units = sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk label type: dos  
Disk identifier: 0x3c941082  
  
   Device Boot      Start         End      Blocks   Id  System  
  /dev/vdb1        2048     62914559     31456256   8e  Linux LVM  
  
Command (m for help): w  
The partition table has been altered!  
Calling ioctl() to re-read partition table.  
Syncing disks.  
[root@web02 ~]#
```

t 명령옵션으로 파티션의 시스템 아이디를 변경하고 저장합니다.

1-11. LVM 설치

LVVM 설치를 위해 다음 명령 실행

- `yum install -y lvm2`

```
root@web02:~  
[root@web02 ~]# yum install -y lvm2  
Loaded plugins: fastestmirror  
base | 3.6 kB | 00:00  
extras | 2.9 kB | 00:00  
mysql-connectors-community | 2.6 kB | 00:00  
mysql-tools-community | 2.6 kB | 00:00  
mysql180-community | 2.6 kB | 00:00  
updates | 2.9 kB | 00:00
```

Copyright© 2023 kt cloud corp. All rights reserved.

1-12. 파티션 확인

```
root@web02:~# fdisk -l
```

Disk /dev/vda: 53.7 GB, 53687091200 bytes, 104857600 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000e4cff

Device	Boot	Start	End	Blocks	Id	System
/dev/vda1	*	2048	1050623	524288	83	Linux
/dev/vda2		1050624	3147775	1048576	82	Linux swap / Solaris
/dev/vda3		3147776	104857566	50854895+	83	Linux

Disk /dev/vdb: 32.2 GB, 32212254720 bytes, 62914560 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x3c941082

Device	Boot	Start	End	Blocks	Id	System
/dev/vdb1		2048	62914559	31456256	8e	Linux LVM

root@web02 ~]#

파티션 확인

/dev/vdb1 로 파티션 설정된 것을 확인할 수 있습니다.

1-13. LVM 구성

```
root@web02:~# ls /dev/vd*
/dev/vda /dev/vda1 /dev/vda2 /dev/vda3 /dev/vdb /dev/vdb1
root@web02:~# pvcreate /dev/vdb1
Physical volume "/dev/vdb1" successfully created.
root@web02:~# pvs
PV          VG Fmt Attr PSize  PFree
/dev/vdb1   lvm2 --- <30.00g <30.00g
root@web02:~# vgcreate myvg /dev/vdb1
Volume group "myvg" successfully created
root@web02:~# vgs
VG      #PV #LV #SN Attr   VSize   VFree
myvg    1   0   0 wz--n- <30.00g <30.00g
root@web02:~# pvs
PV          VG      Fmt Attr   PSize  PFree
/dev/vdb1   myvg    lvm2 a--  <30.00g <30.00g
root@web02:~# lvcreate -l 100%FREE -n lv01 myvg
Logical volume "lv01" created.
root@web02:~# lvs
LV      VG      Attr   LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Conve
lv01    myvg    -wi-a----- <30.00g
```

- pvcreate 물리볼륨 생성
- vgcreate 볼륨그룹 생성
- lvcreate Logical Volume 생성

물리 볼륨 생성 및 확인 → 볼륨그룹 생성 및 확인 → Logical Volume 생성 순으로 진행해야 합니다.

1-14. LVM 확인 및 Format

```
root@web02:~# pvscan
PV /dev/vdb1   VG myvg          lvm2  [<30.00 GiB / 0   free]
Total: 1 [<30.00 GiB] / in use: 1 [<30.00 GiB] / in no VG: 0 [0   ]
root@web02:~# vgscan
Reading volume groups from cache.
Found volume group "myvg" using metadata type lvm2
root@web02:~# lvscan
ACTIVE          '/dev/myvg/lv01' [<30.00 GiB] inherit
root@web02:~#
```

LVM 생성 확인

```
root@web02:~# mkfs.ext4 /dev/myvg/lv01
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
```

Format

볼륨을 사용하기 위해 리눅스 파일시스템에 맞도록 포맷을 해야 합니다.
대표적으로 사용하는 리눅스 파일시스템으로 ext4와 xfs가 있습니다.

이 실습에서는 ext4로 포맷을 진행합니다.

1-15. 마운트와 fstab 파일 등록

```
root@web02:~  
[root@web02 ~]# mkdir /data01  
[root@web02 ~]# mount /dev/myvg/lv01 /data01  
[root@web02 ~]# vi /etc/fstab
```

UUID=3d0ca40b-e945-4c0c-88fc-113f518b21cc	/	ext4	default		
ts	1 1				
UUID=313d9281-54bb-47f7-aaca-3e21104c808b	/boot	ext4	default		
ts	1 2				
UUID=bed849d0-36db-4cdb-879c-2a394134cecb	swap	swap	default		
ts	0 0				
/dev/myvg/lv01	/data01	ext4	defaults	0	0

추가입력 후 저장

- ① i - [Enter]
- ② 입력
- ③ [Esc]
- ④ :wq

마운트 할 디렉토리 생성 후 마운트를 합니다.
서버 재 시작 시에도 자동으로 마운트 될 수 있도록 /etc/fstab 파일을 수정합니다.

1-16. 자동 마운트 테스트

```
root@web02:~# df -Th
Filesystem      Type      Size      Used      Avail     Use%      Mounted on
devtmpfs        devtmpfs  486M       0      486M      0%        /dev
tmpfs           tmpfs     496M       0      496M      0%        /dev/shm
tmpfs           tmpfs     496M      6.7M     489M      2%        /run
tmpfs           tmpfs     496M       0      496M      0%        /sys/fs/cgroup
/dev/vda3       ext4      48G        2.0G     44G        5%        /
/dev/vda1       ext4      488M      182M     271M      41%       /boot
tmpfs           tmpfs     100M       0      100M      0%        /run/user/1000
/dev/mapper/myvg-lv01 ext4      30G        45M     28G        1%        /data01
root@web02:~# umount /data01
root@web02:~# mount -a
root@web02:~# df -Th
Filesystem      Type      Size      Used      Avail     Use%      Mounted on
devtmpfs        devtmpfs  486M       0      486M      0%        /dev
tmpfs           tmpfs     496M       0      496M      0%        /dev/shm
tmpfs           tmpfs     496M      6.7M     489M      2%        /run
tmpfs           tmpfs     496M       0      496M      0%        /sys/fs/cgroup
/dev/vda3       ext4      48G        2.0G     44G        5%        /
/dev/vda1       ext4      488M      182M     271M      41%       /boot
tmpfs           tmpfs     100M       0      100M      0%        /run/user/1000
/dev/mapper/myvg-lv01 ext4      30G        45M     28G        1%        /data01
```

umount 후 mount -a 명령을 실행해서 자동 마운트 여부를 확인합니다.

1-17. 볼륨 사용(파일생성)

```
root@web02/data01
[root@web02 ~]# cd /data01
[root@web02 data01]# pwd
/data01
[root@web02 data01]# ls
lost+found
[root@web02 data01]# touch test.txt
[root@web02 data01]# ls
lost+found test.txt
[root@web02 data01]#
```

- ① /data01로 이동
- ② pwd 명령으로 현재 디렉토리 확인
- ③ test.txt 파일 생성
- ④ ls 명령으로 파일 확인

Copyright© 2023 kt cloud corp. All rights reserved.

추가된 볼륨에 파일을 생성하고 확인합니다.

1-18. MOUNT와 Umount

```
[root@web02 ~]# cd
[root@web02 ~]# umount /data
[root@web02 ~]# cd /data
[root@web02 data]# ls
[root@web02 data]# cd
[root@web02 ~]# mount /data
[root@web02 ~]# cd /data
[root@web02 data]# ls
lost+found test.txt
[root@web02 data]#
```

umount 명령을 실행하면 /data의 파일이 보이지 않음



umount 명령은 운영 체제에서 파일 시스템을 마운트 지점으로부터 연결을 해제하며 컴퓨터로부터 분리되어 더 이상 접근을 하지 못하게 합니다.
mount와 umount 명령어는 변경 사항을 적용하기 위해서는 루트 사용자 권한이 필요합니다.

콘솔에서 Volume [연결] 또는 [연결해제]를 수행하여 attach 및 detach 할 수 있습니다.
이는 마치 외장하드를 연결하고 분리하는 작업과 유사합니다.

실습 2. 볼륨스냅샷 생성과 활용

01

볼륨스냅샷 생성

02

볼륨스냅샷으로 새 볼륨 생성

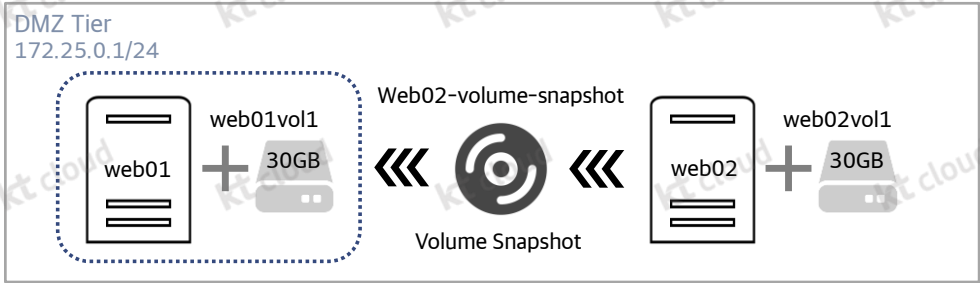
03

서버와 새 볼륨 연결

실습 2에서는 볼륨스냅샷을 생성하고 활용합니다.

- 볼륨스냅샷 생성
- 볼륨스냅샷으로 새 볼륨 생성
- 새 볼륨과 서버의 연결 및 사용

실습 2. 개요



Copyright© 2023 kt cloud corp. All rights reserved.

2-1. 볼륨스냅샷 생성(1)



‘스냅샷’은 현재 VM disk volume들의 상태를 저장한 것입니다.

[Server]-[Volume]-[...]'의 [스냅샷]을 클릭하여 볼륨스냅샷을 생성할 수 있습니다.
예제에서는 web02volume에 대한 web02-volume-snapshot을 생성합니다.

생성된 스냅샷은 [Server]-[Volume Snapshot]에서 확인할 수 있습니다.

2-1. 볼륨스냅샷 생성(2)

web02의 볼륨을 관리합니다.

영역별 | 스냅샷

이름	타입	상호	용량
58ff9c46-6887-4f8a-bed1-5849ff7528f3	Root	HDD	50 GB
web01vol1	Data	HDD	30 GB

볼륨 스냅샷 관리

- [Server]-[Server]-[...]-[볼륨관리]
- Root 볼륨 관리 기능 포함

볼륨 스냅샷 생성 및 연결 관리는 [Server]-[Server]-[...]-[볼륨관리] 메뉴에서도 수행할 수 있습니다.
특히 [Server] 메뉴의 볼륨관리에는 Root 볼륨의 스냅샷 생성이 포함되어 있습니다.

2-2. Volume Snapshot으로 새 볼륨 생성

The screenshot displays the kt cloud management console. On the left, the 'All Services' menu is visible, with 'Volume Snapshot' highlighted under the 'Volume' category. The main panel shows the 'Volume Snapshot' page with a table of existing snapshots. A red box highlights the '볼륨 생성' (Create Volume) button. A red arrow points from this button to a modal dialog titled 'web02-volume-snapshot'. The dialog contains fields for '이름' (Name) set to 'web01vol1', '용량' (Capacity) set to '30', and '서버 연결' (Server Connection) set to 'web01'. A blue callout bubble points to the '이름' field with the text 'web01volume', and another bubble points to the '서버 연결' field with the text 'web01 서버와 연결'. At the bottom of the dialog, a red box highlights the '생성하기' (Create) button. The price '2,100원/월' is also visible.

[Server]-[Volume Snapshot]에서 [볼륨생성]을 클릭하여 새 볼륨을 생성할 수 있습니다.

2-3. 추가된 볼륨 확인

+

All Services

Dashboard

User

IAM

Server

Server

Networking

Tier

Server Image

Volume

Volume Snapshot

Traffic

SSH Key Pair

Volume

가상 서버의 볼륨을 관리합니다.

볼륨 생성

연결

연결해제

삭제

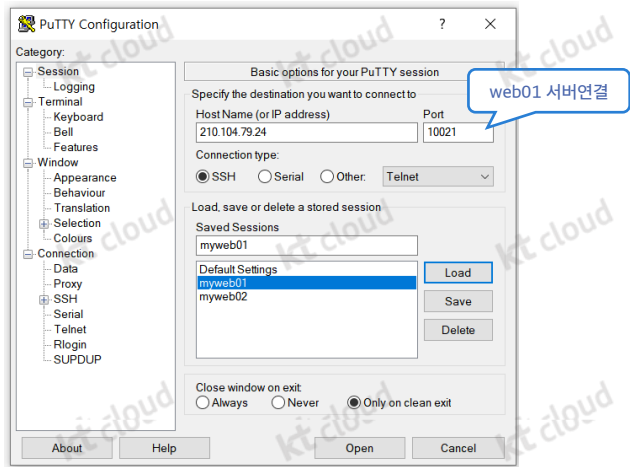
...

모든 위치 · 모두 · 추가 >

	이름 ↓	위치	적용 서버	상태	타
<input type="checkbox"/>	web01vol1	DX-M1	web01	● 사용	추
<input type="checkbox"/>	web02vol1	DX-M1	web02	● 사용	추

Copyright© 2023 kt cloud corp. All rights reserved.

2-4. 볼륨을 추가한 서버 접속



PuTTY를 실행하고 web01 서버로 연결합니다.

2-5. 볼륨 확인 및 LVM 설치

```
[root@web01 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
devtmpfs        devtmpfs  486M   0    486M   0% /dev
tmpfs           tmpfs     496M   0    496M   0% /dev/shm
tmpfs           tmpfs     496M  44M   452M   9% /run
tmpfs           tmpfs     496M   0    496M   0% /sys/fs/cgroup
/dev/vda3        ext4      48G   2.0G   44G   5% /
/dev/vda1        ext4      488M  182M  271M  41% /boot
tmpfs           tmpfs     100M   0    100M   0% /run/user/1000
```

추가한 볼륨 마운트 필요

```
root@web01:~
[root@web01 ~]# yum install -y lvm2
Loaded plugins: fastestmirror
base                                     | 3.6 kB    00:00
extras                                 | 2.9 kB    00:00
mysql-connectors-community             | 2.6 kB    00:00
mysql-tools-community                 | 2.6 kB    00:00
mysql80-community                     | 2.6 kB    00:00
```

콘솔에서 볼륨을 추가한 web01 서버에 아직 추가된 볼륨 30GB가 보이지 않습니다.

2-6. Attach 된 볼륨 확인 및 마운트

```
root@web01:~# pvscan
PV /dev/vdb1   VG myvg               lvm2 [<30.00 GiB / 0   free]
Total: 1 [<30.00 GiB] / in use: 1 [<30.00 GiB] / in no VG: 0 [0   ]
root@web01 ~# vgscan
Reading volume groups from cache.
Found volume group "myvg" using metadata type lvm2
root@web01 ~# lvscan
inactive            '/dev/myvg/lv01' [<30.00 GiB] inherit
root@web01 ~# lvchange -ay myvg/lv01
root@web01 ~# lvscan
ACTIVE              '/dev/myvg/lv01' [<30.00 GiB] inherit
root@web01 ~# mkdir /dir1
root@web01 ~# mount /dev/myvg/lv01 /dir1
root@web01 ~# cd /dir1
root@web01 dir1# ls
lost+found test.txt
root@web01 dir1#
```

마운트 및 파일 확인

물리볼륨, 볼륨그룹, 논리볼륨을 확인합니다.
또한 web01 서버에서 해당 볼륨에 대한 마운트를 진행합니다.
마운트 후 web02 서버에서 생성한 파일이 보입니다.

root@web01:~

```
[root@web01 ~]# vi /etc/fstab
[root@web01 ~]#
```

편집

추가입력 후 저장

```
UUID=bed849d0-36db-4cdb-879c-2a394134cecb swap swap default
ts 0 0
/dev/myvg/lv01 /dir1 ext4 defaults 0 0
~
~
~
:wq
```

- ① i - [Enter]
- ② 입력
- ③ [Esc]
- ④ :wq

Copyright© 2023 kt cloud corp. All rights reserved.

2-8. 볼륨 마운트 확인

```
root@web01:~# df -Th
Filesystem                Type      Size  Used Avail Use% Mounted on
devtmpfs                  devtmpfs  486M   0    486M   0% /dev
tmpfs                     tmpfs     496M   0    496M   0% /dev/shm
tmpfs                     tmpfs     496M  6.7M  489M   2% /run
tmpfs                     tmpfs     496M   0    496M   0% /sys/fs/cgroup
/dev/vda3                 ext4      48G   2.0G   44G    5% /
/dev/vda1                 ext4     488M  182M  271M   41% /boot
tmpfs                     tmpfs     100M   0    100M   0% /run/user/1000
/dev/mapper/myvg-lv01     ext4      30G   45M   28G    1% /dir1 →마운트 확인

[root@web01 ~]# umount /dir1
[root@web01 ~]# mount /dir1
[root@web01 ~]# df -Th
Filesystem                Type      Size  Used Avail Use% Mounted on
devtmpfs                  devtmpfs  486M   0    486M   0% /dev
tmpfs                     tmpfs     496M   0    496M   0% /dev/shm
tmpfs                     tmpfs     496M  6.7M  489M   2% /run
tmpfs                     tmpfs     496M   0    496M   0% /sys/fs/cgroup
/dev/vda3                 ext4      48G   2.0G   44G    5% /
/dev/vda1                 ext4     488M  182M  271M   41% /boot
tmpfs                     tmpfs     100M   0    100M   0% /run/user/1000
/dev/mapper/myvg-lv01     ext4      30G   45M   28G    1% /dir1
```

web01 서버에서는 포맷이 완료된 디스크 볼륨을 복제했으므로 마운트 후 바로 사용할 수 있습니다.

- mkdir 명령으로 마운트 지점을 생성
- mount 명령으로 해당 경로로 볼륨 마운트
- df -Th 명령으로 마운트 확인

실습 3. NAS 생성과 활용

01

NAS 생성

02

NAS 스냅샷 생성

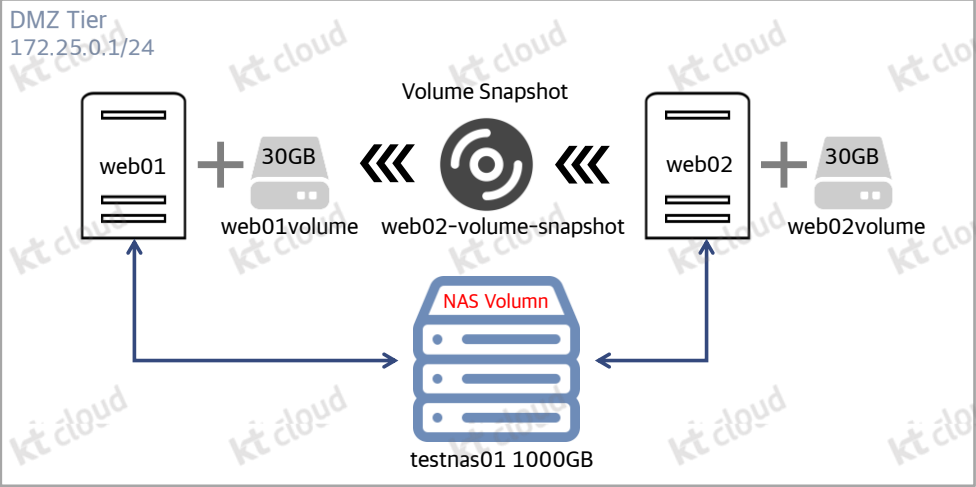
03

NAS 활용

실습 3에서는 NAS를 생성하고 활용합니다.

- NAS 생성
- NAS 스냅샷 생성
- NAS를 활용한 서버간 데이터 공유

실습 3. 개요



Copyright© 2023 kt cloud corp. All rights reserved.

3-1. NAS 네트워크 생성

Platform @ D1

All Services

Dashboard

User

IAM

Load Balancer

NAS

NAS Volume

Snapshot

CIFS Authentication

IQN

Log History

Server

Watch

Token

Auto Scaling

Messaging

Network

NAS > NAS Volume

메뉴얼

오라클문의

만족도평가

NAS Volume

볼륨 생성 후 볼륨을 사용하고자 하는 서버 IP에 대한 접근제어 설정이 필요합니다.
접근제어 설정은 상세 기능() 버튼을 클릭 후 "접근제어 설정" 메뉴를 선택하시면 됩니다.

볼륨 생성 후 볼륨을 사용하고자 하는 서버 IP에 대한 접근제어 설정이 필요합니다.
접근제어 설정은 상세 기능() 버튼을 클릭 후 "접근제어 설정" 메뉴를 선택하시면 됩니다.

이름 상태 위치

초화된 데이터가 없습니다.

NAS 네트워크를 관리합니다.

생성 삭제

Tier	NAS 네트워크	위치	CIDR	VLAN
<input type="checkbox"/> Private_Sub	● 사용 안 함	DX-M1	172.25.1.1/24	1797
<input type="checkbox"/> DMZ_Sub	● 사용	DX-M1	172.25.0.1/24	1798

NAS 네트워크 생성

① [NAS]-[NAS Volume]에서 [NAS 네트워크관리]를 클릭

② Tier를 선택하고 [생성] 클릭

VM을 생성한 Tier를 먼저 확인 후 다음과 같이 NAS 네트워크 관리에서 해당 Tier로 NAS 네트워크를 생성해야 합니다.

- [NAS]-[NAS Volume]에서 [NAS 네트워크관리]를 클릭
- Tier를 선택하고 [생성] 클릭

3-2. NAS 볼륨 생성

The screenshot shows the 'NAS Volume' creation page in the kt cloud console. On the left, the 'NAS Volume' menu item is highlighted. The main panel shows the '볼륨 생성' (Create Volume) button. The right panel contains the configuration form with the following fields and annotations:

- 이름 (Name):** testnas01
- 위치 (Location):** DX-M1
- Tier:** DMZ_Sub (Annotated with 'DMZ-Sub 선택')
- 볼륨구분 (Volume Type):** HDD
- 프로토콜 (Protocol):** NFS (Annotated with 'Linux/Unix 파일공유 프로토콜')
- 필요금액 (Required Amount):** 기본 1000GB (Annotated with '기본 1000GB')
- 요금제 (Plan):** 기본 1,000 GB + 추가 GB
- 가격 (Price):** 70,000 원/월
- 생성하기 (Create):** Button at the bottom right.

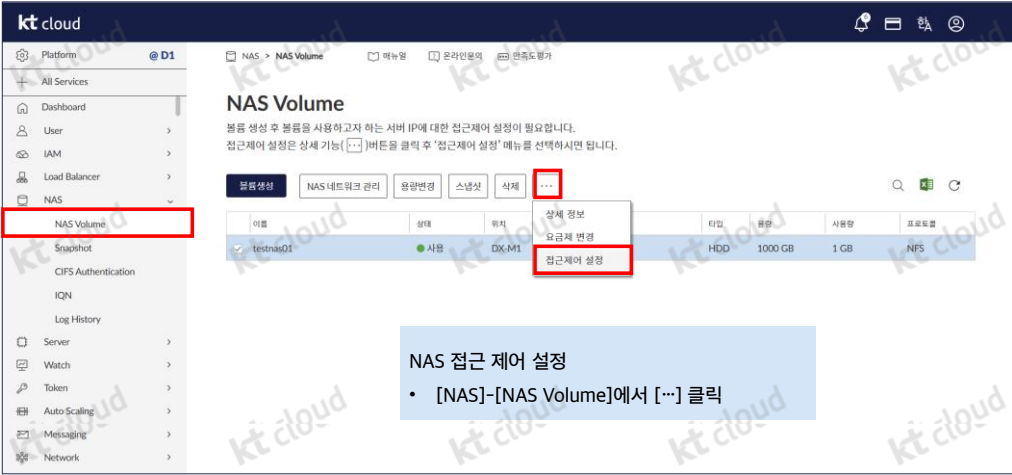
Below the main panel, a summary box titled 'NAS 볼륨 생성' lists the steps:

- ① [NAS]-[NAS Volume]에서 [볼륨생성] 클릭
- ② 이름, Tier 등의 값을 입력 또는 선택
- ③ [생성하기]

NAS Volume 생성은 [NAS]-[NAS Volume]에서 [볼륨생성] 클릭해서 할 수 있습니다. 볼륨생성 창에서 볼륨이름, Tier 선택 등을 하고 사용할 프로토콜 등을 지정한 후 [생성하기]를 클릭합니다.

NAS Volume의 기본 크기는 최소 1000GB 입니다.

3-3. 접근제어 설정(1)



NAS Volume에 대한 접근제어 설정은 [NAS]-[NAS Volume]에서 [...]로 표시된 숨긴 메뉴에 있습니다.

3-4. 접근 제어 설정(2)

testnas01 볼륨의 접근제어를 설정합니다.
* 입력하지 않을 경우 모든 접근을 거부합니다.

Read Write	접근허용할 정보를 IP나 CIDR형식으로 입력하세요.	+ 추가
Read Only		
Read Write		

NAS 볼륨 접근 제어 설정

- ① 목록에서 Read Write 접근제어 선택
- ② CIDR 형식으로 접근제어 설정
- ③ [추가]

testnas01 볼륨의 접근제어를 설정합니다.
* 입력하지 않을 경우 모든 접근을 거부합니다.

Read Write	0.0.0.0/0	+ 추가
Read Write	0.0.0.0/0	🗑

Copyright© 2023 kt cloud corp. All rights reserved.

NAS Volume의 접근제어는 Read Write 또는 Read Only로 설정할 수 있으며 CIDR 형식으로 지정합니다.

예제에서는 CIDR을 0.0.0.0/0으로 설정합니다.

3-5. NAS Volume Snapshot 생성

NAS Volume

볼륨 생성 후 볼륨을 사용하고자 하는 서버 IP에 대한 접근제어 설정이 필요합니다.
접근제어 설정은 상세 기능(...)버튼을 클릭 후 '접근제어 설정' 메뉴를 선택하시면 됩니다.

볼륨생성 NAS 네트워크 관리 용량변경 **스냅샷** 삭제 ...

이름 상태

<input checked="" type="checkbox"/>	testnas01	● 사용
-------------------------------------	-----------	------

NAS 볼륨 스냅샷 생성

- ① [NAS]-[NAS Volume]에서 [스냅샷] 클릭
- ② 스냅샷 이름 입력 후 [생성하기]

testnas01 볼륨의 스냅샷을 생성합니다.

스냅샷 이름: **testnas01snap**

스냅샷은 NAS 볼륨 안에 생성되므로 별도 요금이 없습니다.
생성한 스냅샷은 NAS > Snapshot에서 확인할 수 있습니다.

취소 **생성하기**

[NAS]-[NAS Volume] 에서 특정 NAS를 지정한 후 [스냅샷]을 클릭합니다.
스냅샷 이름을 부여한 후 생성할 수 있습니다.

스냅샷은 NAS 볼륨에 생성되므로 별도로 과금되지 않습니다.

3-6. NAS Volume 스냅샷 관리

Platform @ D1

All Services

Dashboard

User >

IAM >

Load Balancer >

NAS >

NAS Volume

Snapshot

CIFS Authentication

IQN

Log History

Nas > Snapshot

메뉴얼 온라인문외 만족도평가

Snapshot

NAS의 스냅샷을 관리합니다.

볼륨 생성

삭제

	이름	대상 Nas	위치
<input type="checkbox"/>	testnas01snap	testnas01	DX-M1

NAS 볼륨 스냅샷 관리

- [NAS]-[Snapshot]에서 [볼륨생성] 또는 [삭제하기]

생성된 스냅샷은 [NAS]-[Snapshot]에서 확인하거나 삭제할 수 있습니다.

3-7. NAS Volume Mount Path 찾기

NAS Volume Mount Path

- ① [NAS]-[NAS Volume]에서 [...] 클릭
- ② [상세정보] 선택
- ③ Mount Path 확인

이름	용량	프로토콜	Mount Path
testnas01	1000 GB	NFS	172.25.0.9:/share_5fc53a09_eadf_48ca_b2bc_904193e18943

서버가 NAS Volume에 접근하기 위하여 NAS Volume의 Mount Path 정보가 필요합니다.

NAS Volume의 Mount Path는 [NAS]-[NAS Volume]에서 [...]로 표시된 숨김메뉴의 [상세정보]에서 확인할 수 있습니다.

3-8. NAS Volume Mount Path 복사

이름	testnas01	생성 일시	03/24/2023 02:10:32
용량	1000 GB	요금제	월요금제
프로토콜	NFS		
Mount Path	172.25.0.9:/share_5fc53a09_eadf_48ca_b2bc_904193e18943		

Mount Path 정보를 메모장으로 복사

복사 완료
다.

클립보드에 복사되었습니다

*제목 없음 - Windows 메모장

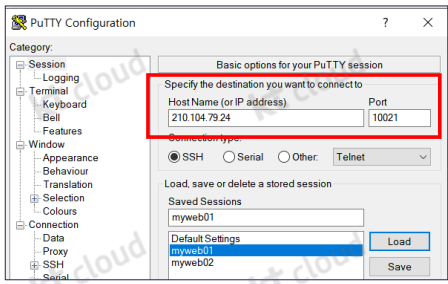
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

172.25.0.9:/share_5fc53a09_eadf_48ca_b2bc_904193e18943

Ln 1, Col 55 100% Windows (CRLF) UTF-8

Mount path는 NAS 볼륨 연결에 필요한 정보입니다.

3-9. Linux 서버에서 NAS Volume Mount



NAS Volume Mount

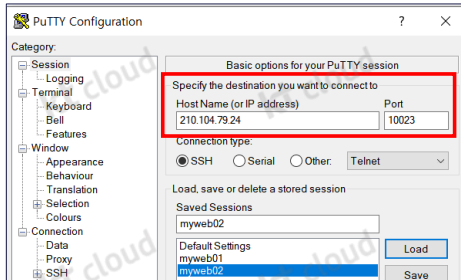
- ① PuTTY 실행 후 web01 서버 접속
- ② /n1 디렉토리 생성
- ③ NAS Volume Mount Path 정보로 /n1 디렉토리 마운트
- ④ /n1 디렉토리에 파일 생성 (t1.txt)

```
[root@web01 ~]# sudo -i
[root@web01 ~]# mkdir /n1
[root@web01 ~]# mount -t nfs 172.25.0.9:/share_5fc53a09_eadf_48ca_b2bc_904193e18943 /n1
[root@web01 ~]# cd /n1
[root@web01 n1]# ls
[root@web01 n1]# touch t1.txt
[root@web01 n1]# ls
t1.txt
```

NAS Volume Mount Path 정보

앞서 확인한 mount path 정보로 NAS를 mount 합니다.

3-10. NAS Volume Linux 서버 간 공유



NAS Volume Mount

- ① PuTTY 실행 후 web02 서버 접속
- ② /n2 디렉토리 생성
- ③ NAS Volume Mount Path 정보로 /n2 디렉토리 마운트
- ④ /n2 디렉토리에 파일 생성 (t2.txt)

```
[root@web02 ~]# sudo -i
[root@web02 ~]# mkdir /n2
[root@web02 ~]# mount -t nfs 172.25.0.9:/share_5fc53a09_eadf_48ca_b2bc_904193e18
943 /n2
[root@web02 ~]# cd /n2
[root@web02 n2]# ls
t1.txt
[root@web02 n2]# touch t2.txt
[root@web02 n2]# ls
t1.txt t2.txt
```

testnas01 NAS Volume 공유

web02 서버도 /n2 라는 이름의 디렉토리를 생성한 후 동일한 NAS Volume Mount Path 정보로 마운트를 합니다.

web02 서버는 web01 서버가 작성한 파일을 볼 수 있습니다.

web01 서버 역시 web02 서버가 작성한 파일을 볼 수 있습니다.

즉 t1.txt 파일과 t2.txt 파일을 두 서버가 공유하고 있습니다.

실습 4. 정리

01

LVM 삭제

02

볼륨 해제

03

NAS 해제

실습 3에서는 NAS를 생성하고 활용합니다.

- NAS 생성
- NAS 스냅샷 생성
- NAS를 활용한 서버간 데이터 공유

4-1. 볼륨 마운트 해제 및 LVM 관련 삭제

```
root@web02:~  
[root@web02 ~]# umount /data01  
[root@web02 ~]# lvscan  
ACTIVE '/dev/myvg/lv01' [39.99 GiB] inherit  
[root@web02 ~]# lvremove /dev/myvg/lv01  
Do you really want to remove active logical volume myvg/lv01? [y/n]: y  
Logical volume "lv01" successfully removed  
[root@web02 ~]# pvscan  
PV /dev/vdb1   VG myvg          lvm2 [ <30.00 GiB / <30.00 GiB free]  
PV /dev/vdc1   VG myvg          lvm2 [ <10.00 GiB / <10.00 GiB free]  
Total: 2 [39.99 GiB] / in use: 2 [39.99 GiB] / in no VG: 0 [0 ]  
[root@web02 ~]# vgremove myvg  
Volume group "myvg" successfully removed  
[root@web02 ~]# pvremove /dev/vdb1  
Labels on physical volume "/dev/vdb1" successfully wiped.
```

삭제는 역순으로 진행

- ① umount 로 마운트해제
- ② lvremove 논리볼륨 삭제
- ③ vgremove 볼륨그룹삭제
- ④ pvremove 물리볼륨삭제

Copyright© 2023 kt cloud corp. All rights reserved.

4-2. /etc/fstab 수정

```
root@web02~  
login as: centos  
Authenticating with public key "imported-openssh-key"  
[centos@web02 ~]$ sudo -i  
[root@web02 ~]# vi /etc/fstab
```

/dev/myvg/lv01	/data01	ext4	defaults	0	0
----------------	---------	------	----------	---	---

삭제 후 저장

"/etc/fstab" 12L, 570C

4-3. LVM 삭제 확인

```
root@web01:~  
[root@web01 ~]# fdisk -l
```

Disk /dev/vda: 53.7 GB, 53687091200 bytes, 104857600 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000e4cff

Device	Boot	Start	End	Blocks	Id	System
/dev/vda1	*	2048	1050623	524288	83	Linux
/dev/vda2		1050624	3147775	1048576	82	Linux swap / Solaris
/dev/vda3		3147776	104857566	50854895+	83	Linux

Disk /dev/vdb: 32.2 GB, 32212254720 bytes, 62914560 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0xd1f398f7

Device	Boot	Start	End	Blocks	Id	System
/dev/vdb1		2048	62914559	31456256	8e	Linux LVM

Copyright© 2023 kt cloud corp. All rights reserved.

4-4. 콘솔에서 볼륨연결 해제

Volume

가상 서버의 볼륨을 관리합니다.

볼륨 생성

연결

연결해제

삭제

...

모든 위치 · 모두 · 추가 >

이름 ↓
<input checked="" type="checkbox"/> web01vol1
<input type="checkbox"/> web02vol1
<input type="checkbox"/> web02vol2

① [Server] - [Volume]

② Volume 선택 - [연결해제]

✕

볼륨 서버 연결을 해제합니다.

취소

해제하기

Copyright© 2023 kt cloud corp. All rights reserved.

4-5. 볼륨 연결 해제 확인

```
root@web01:~# fdisk -l
```

Disk /dev/vda: 53.7 GB, 53687091200 bytes, 104857600 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000e4cff

Device	Boot	Start	End	Blocks	Id	System
/dev/vda1	*	2048	1050623	524288	83	Linux
/dev/vda2		1050624	3147775	1048576	82	Linux swap / Solaris
/dev/vda3		3147776	104857566	50854895+	83	Linux

```
root@web01 ~]#
```

4-6. 볼륨제거

Volume

가상 서버의 볼륨을 관리합니다.

볼륨 생성

연결

연결해제

삭제

...

모든 위치 · 모두 · 추가 >

🔍

🗖

🔄

이름 ↓	위치	적용 서버	상태	타입	상품	용량	생성일시
<input checked="" type="checkbox"/> web01vol1	DX-M1	--					
<input type="checkbox"/> web02vol1	DX-M1	web02					
<input type="checkbox"/> web02vol2	DX-M1	web02					

① [Server] - [Volume]

② Volume 선택 - [삭제]

web01vol1 볼륨을 삭제합니다.

! 볼륨 내 데이터가 모두 삭제됩니다.

취소

삭제하기

연결해제된 볼륨만 삭제할 수 있습니다.

4-7. NAS 마운트 및 연결 해제

```
root@web01:~  
[root@web01 ~]# umount /n1  
[root@web01 ~]#
```

NAS > NAS Volume

볼륨 생성 후 볼륨을 사용하여 접근제어 설정은 상세 기능

볼륨 생성 NAS 네트워크

이름

☒ testnas01

이름 testnas01

Mount Path 172.25.0.38:/share_b83...

프로토콜 NFS

취소 삭제하기

추가실습 1. 가상서버의 볼륨 확장(LVM)

01

볼륨생성 및 볼륨 추가

02

파티션 생성 및 포맷

03

볼륨 연결 및 해제

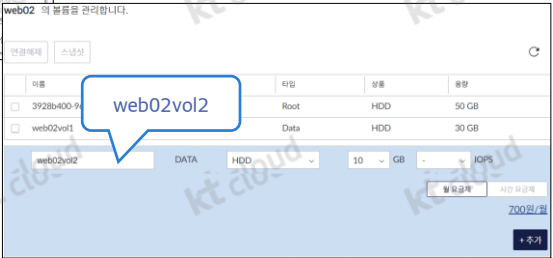
실습 1에서는 리눅스서버에 볼륨을 추가해 봅니다.
새 볼륨은 블록 스토리지에 해당되며, 파일시스템으로 포맷 후 사용할 수 있습니다.
추가된 볼륨이 사용가능하도록 파티션을 생성하고 포맷을 진행합니다.

또한 볼륨 연결과 해제를 실습하겠습니다.

1-1. 서버에 새 볼륨 추가



- ① [Server] - [Server] - web02 서버 생성
- ② 숨김메뉴[...] - [Volume관리]
- ③ 새 볼륨 명 : web02vol2
- ④ 사이즈 10 GB



1-2. 볼륨 확인

Volume

가상 서버의 볼륨을 관리합니다.

볼륨 생성

연결

연결해제

삭제

...

모든 위치 · 모두 · 추가 >

🔍

📄

🔄

	이름 ↓	위치	적용 서버	상태	타입	상품	용량	생성일시
<input type="checkbox"/>	web01vol1	DX-M1	web01	● 사용	추가	HDD	30 GB	07/23/2023 22:37:01
<input type="checkbox"/>	web02vol1	DX-M1	web02	● 사용	추가	HDD	30 GB	07/23/2023 21:58:15
<input type="checkbox"/>	web02vol2	DX-M1	web02	● 사용	추가	HDD	10 GB	07/23/2023 23:22:34

Copyright© 2023 kt cloud corp. All rights reserved.

1-3. 리눅스에서 새 볼륨 확인

root@web02~

```
[root@web02 ~]# fdisk -l
```

```
Disk /dev/vda: 53.7 GB, 53687091200 bytes, 104857600 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000e4cff
```

...

```
Disk /dev/mapper/myvg-lv01: 32.2 GB, 32208060416 bytes, 62906368 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disk /dev/vdc: 10.7 GB, 10737418240 bytes, 20971520 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

Copyright© 2023 kt cloud corp. All rights reserved.

1-4. 파티션 생성 및 시스템 ID 변경(1)

```
root@web02 ~]# fdisk /dev/vdc
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x89dd95fc.

Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-20971519, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-20971519, default 20971519):
Using default value 20971519
Partition 1 of type Linux and of size 10 GiB is set
```

Copyright© 2023 kt cloud corp. All rights reserved.

1-5. 파티션 생성 및 시스템 ID 변경(2)

```
Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
[root@web02 ~]#
```

Copyright© 2023 kt cloud corp. All rights reserved.

1-6. 기존 볼륨 확장

```
root@web02 ~# pvcreate /dev/vdc1
Physical volume "/dev/vdc1" successfully created.
root@web02 ~# pvs
PV          VG      Fmt  Attr  PSize    PFree
/dev/vdb1   myvg   lvm2  a--   <30.00g    0
/dev/vdc1   lvm2   ---   <10.00g <10.00g

root@web02 ~# vgextend myvg /dev/vdc1
Volume group "myvg" successfully extended
root@web02 ~# vgs
VG      #PV #LV #SN Attr   VSize    VFree
myvg    2   1   0 wz--n- 39.99g <10.00g

root@web02 ~# lvextend -l +100%FREE /dev/myvg/lv01
Size of logical volume myvg/lv01 changed from <30.00 GiB (7679 extents) to 39.99 GiB (10238 extents).
Logical volume myvg/lv01 successfully resized.
root@web02 ~# lvs
LV      VG      Attr      LSize    Pool Origin Data%  Meta%  Move Log Cpy%Sync Conver
lv01    myvg   -wi-ao---- 39.99g
```

Copyright© 2023 kt cloud corp. All rights reserved.

1-7. 논리볼륨 RESIZE

```
root@web02 ~# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
devtmpfs        devtmpfs  486M   0    486M   0% /dev
tmpfs           tmpfs     496M   0    496M   0% /dev/shm
tmpfs           tmpfs     496M   6.7M  489M   2% /run
tmpfs           tmpfs     496M   0    496M   0% /sys/fs/cgroup
/dev/vda3       ext4      48G    2.0G   44G    5% /
/dev/vda1       ext4      488M   182M   271M   41% /boot
/dev/mapper/myvg-lv01 ext4      30G    45M    28G    1% /data01
tmpfs           tmpfs     100M   0    100M   0% /run/user/1000
root@web02 ~# resize2fs -p /dev/myvg/lv01
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/myvg/lv01 is mounted on /data01; on-line resizing required
old_desc_blocks = 4, new_desc_blocks = 5
The filesystem on /dev/myvg/lv01 is now 10483712 blocks long.
```

Copyright© 2023 kt cloud corp. All rights reserved.

1-8. 볼륨 확장 확인

```
[root@web02 ~]# df -Th
Filesystem                Type      Size  Used Avail Use% Mounted on
devtmpfs                  devtmpfs  486M    0  486M   0% /dev
tmpfs                     tmpfs     496M    0  496M   0% /dev/shm
tmpfs                     tmpfs     496M  6.7M  489M   2% /run
tmpfs                     tmpfs     496M    0  496M   0% /sys/fs/cgroup
/dev/vda3                 ext4      48G   2.0G   44G   5% /
/dev/vda1                 ext4     488M  182M  271M  41% /boot
/dev/mapper/myvg-lv01     ext4      40G   48M   38G   1% /data01
tmpfs                     tmpfs     100M    0  100M   0% /run/user/1000
[root@web02 ~]#
```

Copyright© 2023 kt cloud corp. All rights reserved.

추가실습 2. 가상서버의 볼륨 관리(Non LVM)

01

볼륨생성 및 볼륨 추가

02

파티션 생성 및 포맷

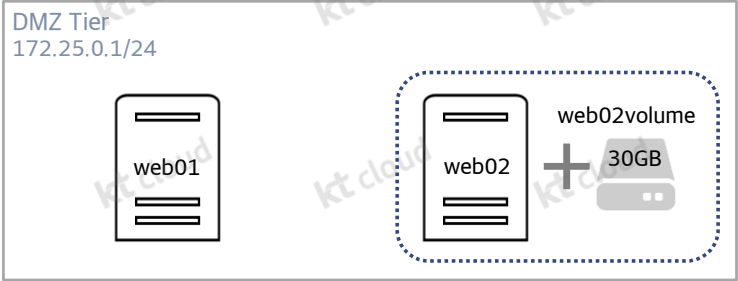
03

볼륨 연결 및 해제

실습 1에서는 리눅스서버에 볼륨을 추가해 보았습니다.
새 볼륨은 블록 스토리지에 해당되며, 파일시스템으로 포맷 후 사용할 수 있습니다.
추가된 볼륨이 사용가능하도록 파티션을 생성하고 포맷을 진행합니다.

또한 볼륨 연결과 해제를 실습하겠습니다.

실습 2. 개요



Copyright© 2023 kt cloud corp. All rights reserved.

2-1. Volume관리 메뉴

The screenshot shows the kt cloud console interface. On the left sidebar, the 'Server' menu is highlighted. In the main content area, the 'Server' page is displayed. A red box highlights the 'More' icon (three dots) in the top bar of the server list. A dropdown menu is open from this icon, and the 'Volume 관리' (Volume Management) option is highlighted with a red box. Below the screenshot, a blue box contains the following instructions:

볼륨관리를 위해

- ① [Server]-[Server]에서 대상 서버(web02)를 선택
- ② [...] 표시의 숨김메뉴를 클릭하고 [Volume 관리] 선택

kt cloud 에서는 원하는 용량의 데이터 볼륨을 생성하고, 이를 서버에 연결하여 저장 공간을 확장할 수 있습니다.
HDD 기반의 볼륨을 지원하며, 볼륨은 10GB 단위로 생성이 가능합니다.

2-2. 새 볼륨추가

web02 의 볼륨을 관리합니다.

연결해제 스냅샷

이름	타입	상품	용량
<input type="checkbox"/> 58f9c46-6887-4f8a-bed1-5849f7528f3	Root	HDD	50 GB

① “+ 이 서버에 Volume 추가” 클릭

② 이름, 상품, 용량, 요금제 설정

③ [생성하기] 클릭

+ 이 서버에 Volume 추가

web02 서버의 볼륨 추가 사양

- 볼륨이름 : web02volume
- 상품 : HDD
- 용량 : 30GB
- 월 요금제

web02 의 볼륨을 관리합니다.

연결해제 스냅샷

이름	타입	상품	용량
<input type="checkbox"/> 58f9c46-6887-4f8a-bed1-5849f7528f3	Root	HDD	50 GB

DATA HDD 10 GB IOPS

700원/월

상품은 HDD와 SSD를 선택할 수 있습니다.
용량은 10GB 단위로 설정가능 합니다.

예제는 web02 서버의 볼륨을 다음 사양으로 추가합니다.

- 볼륨이름 : web02volume
- 상품 : HDD
- 용량 : 30GB
- 월요금제

2-3. 볼륨확인

kt cloud

Platform @ D1

Server > Volume

Volume

가상 서버의 볼륨을 관리합니다.

볼륨 생성 연결 연결예제 삭제 ... 모든 위치 · 모두 · 추가 >

이름	위치	적용 서버	상태	타입	상품	용량
<input type="checkbox"/> web02volume	DX-M1	web02	사용	추가	HDD	30 GB

[Server]-[Volume] 에서 생성된 볼륨 확인

[Server]-[Volume] 에서 생성된 볼륨에 대해 다음 항목을 확인합니다.

- 볼륨이름
- 위치
- 적용서버
- 상태
- 상품
- 용량

2-4. Volume 상세보기

Volume

가상 서버의 볼륨을 관리합니다.

볼륨 생성

연결

연결해제

삭제

...

모든 위치 - 모두 >

🔍

🌐

🔄

이름 ↓	위치	상세 정보	상태	타입	상품	용량	IOPS	생성일시
web02volume	DX-M1	상세 정보 크기변경 스냅샷 요금제 변경	사용	추가	HDD	30 GB	-	03/23/2023 17:24:38

볼륨 이름

web02volume

볼륨 ID

1c91ec55-8389-4806-8340-b3ac18f6faa9

타입

일반

상품

HDD

요금제

월요금제

생성일시

2023-05-22T01:07:55.000000

적용서버

web02

용량

30GB

[Server]-[Volume]-[...]-[상세정보]를 클릭하여 web02 서버에 30GB의 볼륨이 추가된 것을 확인합니다.

또한 [Server]-[Server]-[...]-[상세정보]를 클릭하여 web02 서버의 볼륨 총량을 확인할 수도 있습니다.

2-5. 현재 디스크 용량 상태 확인

```
root@web02:~  
[centos@web02 ~]$ sudo -i  
[root@web02 ~]# df -Th  
Filesystem      Type      Size  Used Avail Use% Mounted on  
devtmpfs        devtmpfs  486M   0    486M   0% /dev  
tmpfs            tmpfs     496M   0    496M   0% /dev/shm  
tmpfs            tmpfs     496M  6.7M  489M   2% /run  
tmpfs            tmpfs     496M   0    496M   0% /sys/fs/cgroup  
/dev/vda3        ext4       48G   1.9G   44G   5% /  
/dev/vda1        ext4      488M  182M  271M  41% /boot  
tmpfs            tmpfs     100M   0    100M   0% /run/user/1000  
[root@web02 ~]#
```

기존 파티션의 파일시스템이 ext4임을 확인

리눅스 서버의 경우 추가될 volume은 파티션 과정을 거쳐야 합니다.
df -Th 리눅스 명령으로 파티션 작업 전 현재 디스크 용량상태 및 파일시스템을 확인
합니다.

2-6. 추가 디스크 확인

```
root@web02:~  
[root@web02 ~]# fdisk -l  
Disk /dev/vda: 53.7 GB, 53687091200 bytes, 104857600 sectors  
Units = sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk label type: dos  
Disk identifier: 0x000e4cff  


| Device    | Boot | Start     | End       | Blocks    | Id | System               |
|-----------|------|-----------|-----------|-----------|----|----------------------|
| /dev/vda1 | *    | 2048      | 1050623   | 524288    | 83 | Linux                |
| /dev/vdb  |      | 3147775   | 1048576   | 1048576   | 82 | Linux swap / Solaris |
| /dev/vdb1 |      | 104857566 | 50854895+ | 50854895+ | 83 | Linux                |

  
Disk /dev/vdb: 32.2 GB, 32212254720 bytes, 62914560 sectors  
Units = sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

새 볼륨을 추가하고 리눅스에서 fdisk -l 명령어로 확인한 결과 attach 된 /dev/vdb 30GB의 disk의 추가된 볼륨을 볼 수 있습니다. 리눅스 명령어 parted -l 명령으로 파티션되지 않은 볼륨상태를 확인할 수도 있습니다.

이제 이 볼륨을 마운트하기 위해 파티션을 설정해야 합니다.

2-7. 포맷 및 파일 시스템 생성

```
root@web02 ~ # mkfs.ext4 /dev/vdb
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
1966080 inodes, 7864320 blocks
393216 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2155872256
240 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
```

Copyright© 2023 kt cloud corp. All rights reserved.

70

볼륨을 사용하기 위해 리눅스 파일시스템에 맞도록 포맷을 해야 합니다.
대표적으로 사용하는 리눅스 파일시스템으로 ext4와 xfs가 있습니다.

이 실습에서는 ext4로 포맷을 진행합니다.

2-8. 마운트 및 확인

```
root@web02:~# mkdir /data
root@web02:~# mount /dev/vdb /data
root@web02:~# df -Th
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
devtmpfs	devtmpfs	486M	0	486M	0%	/dev
tmpfs	tmpfs	496M	0	496M	0%	/dev/shm
tmpfs	tmpfs	496M	6.7M	489M	2%	/run
tmpfs	tmpfs	496M	0	496M	0%	/sys/fs/cgroup
/dev/vda3	ext4	48G	1.8G	44G	4%	/
/dev/vda1	ext4	488M	182M	271M	41%	/boot
tmpfs	tmpfs	100M	0	100M	0%	/run/user/1000
/dev/vdb	ext4	30G	45M	28G	1%	/data

```
root@web02:~#
```

Mount 된 30GB 새 볼륨

- ① 마운트할 /data 디렉토리 생성
- ② /dev/vdb를 /data 디렉토리에 마운트
- ③ df -Th 명령으로 마운트 상태 확인

마운트할 새 디렉토리를 생성하고 볼륨을 해당 디렉토리와 마운트 합니다.
df -Th 명령을 실행해서 새 디스크의 마운트 상태를 확인합니다.

실습 1-5에서 df -Th 명령을 실행한 결과와 비교해 보면 디스크 용량이 추가된 것을 확인할 수 있습니다.

2-9. /etc/fstab에 마운트 옵션 추가

```
[root@web02 ~]# vi /etc/fstab
[root@web02 ~]#
```

편집

/etc/fstab 파일 입력 내용 형식

<마운트 파티션 영역> <마운트 위치> <파일시스템 유형> <마운트 옵션> <dump 설정> <파일 체크 옵션>

(예) /dev/vdb /data ext4 defaults,nofail 0 0 → 내용 추가

```
# /etc/fstab
# Create and mount the file systems.
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=3d0ca40b-e945-4c0c-88fc-113f518b21cc / ext4 default
ts 1 1
UUID=313d9281-54bb-47f7-aaca-3e21104c808b /boot ext4 default
ts 1 2
UUID=bed849d0-36db-4cdb-879c-2a394134cecb swap swap default
ts 0 0
/dev/vdb /data ext4 defaults,nofail 0 0
```

추가

시스템 재시작 후에도 마운트를 유지하기 위해 /etc/fstab 내용을 수정해야 합니다.

```
vi /etc/fstab
/dev/vdb /data ext4 defaults,nofail 0 0
```

CentOS 7점대의 경우 볼륨을 연결하지 않고 VM 부팅 시 볼륨 마운트 오류가 있어도 부팅이 가능하기 위해서는 nofail 마운트 옵션을 추가할 수 있습니다.

2-10. 추가된 볼륨에 데이터 생성

```
root@web02/data
[root@web02 ~]# vi /etc/fstab
[root@web02 ~]# cd /data
[root@web02 data]# pwd
/data
[root@web02 data]# touch test.txt
[root@web02 data]# ls
lost+found test.txt
[root@web02 data]#
```

- ① /data로 이동
- ② pwd 명령으로 현재 디렉토리 확인
- ③ test.txt 파일 생성
- ④ ls 명령으로 파일 확인

추가된 볼륨에 파일을 생성하고 확인합니다.

2-11. MOUNT와 UMount

```
[root@web02 ~]# cd
[root@web02 ~]# umount /data
[root@web02 ~]# cd /data
[root@web02 data]# ls
[root@web02 data]# cd
[root@web02 ~]# mount /data
[root@web02 ~]# cd /data
[root@web02 data]# ls
lost+found test.txt
[root@web02 data]#
```

umount 명령을 실행하면 /data의 파일이 보이지 않음



umount 명령은 운영 체제에서 파일 시스템을 마운트 지점으로부터 연결을 해제하며 컴퓨터로부터 분리되어 더 이상 접근을 하지 못하게 합니다.
mount와 umount 명령어는 변경 사항을 적용하기 위해서는 루트 사용자 권한이 필요합니다.

콘솔에서 Volume [연결] 또는 [연결해제]를 수행하여 attach 및 detach 할 수 있습니다.
이는 마치 외장하드를 연결하고 분리하는 작업과 유사합니다.

2-12. 볼륨 연결 테스트

The screenshot shows the kt cloud console interface. On the left, there's a sidebar with navigation options like 'All Resources', 'Dashboard', 'Server', 'Network', 'Storage', etc. The main area displays a list of servers. A modal window is open, showing a warning message about stopping the server. A blue callout bubble points to the 'Stop' button with the text 'web02 서버 정지 후 재시작' (Stop and restart web02 server). Below the modal, a terminal window shows the following commands and output:

```
login as: centos
Authenticating with public key "imported-openssh-key"
[centos@web02 ~]$ cd /data
[centos@web02 data]$ ls
lost+found test.txt
[centos@web02 data]$
```

A blue callout bubble points to the terminal output with the text '서버 재시작 시 볼륨이 attach 상태임을 확인' (Check if the volume is in attach state when the server is restarted).

/etc/fstab 설정으로 서버 재부팅시 볼륨이 자동으로 연결됩니다.

