스토리지(하드 디스크) 관리 LVM 실습

학습 목표

* 스토리지 관리 툴을 사용하여 LVM을 만들 수 있다.
* 기존 LVM을 확장할 수 있다.
* 시스템 재부팅시에도 LVM을 영구 마운트되도록 설정할 수 있다.

실습 준비

실습 시나리오

* 30GB 파티션 2 개가 필요하다. 이 파티션은 향후에 확장이 가능해야 한다.
* 한 개의 하드 디스크가 최대 20GB 이다.
* 세 개의 하드 디스크가 합쳐진 볼륨 그룹 이름은 vg1이다.
* LV 이름은 각각 lvm1, lvm2 이다.
* lvm1은 ext4로 포맷한다.
* lvm2는 xfs로 포맷한다.
* lvm1은 /mnt/lv1에 영구적으로 마운트되어야 한다
* lvm2는 /mnt/lv2에 영구적으로 마운트되어야 한다

단계-1 VMWare에서 하드 디스크 추가하기(VMWare 창에서 작업)

VM 🡪 Settings …

Hard Disk (SCSI) 목록 확인

Add 🡪 Hard Disk 🡪 Next 🡪 SCSI (Recommended) 🡪 Next 🡪

Create a new virtual disk 🡪 Store virtual disk as a single file 선택 🡪 Next 🡪

Disk file 🡪 Finish

본 단계를 필요한 만큼 반복한다.

단계-2 Linux-1 시스템이 작동 중이었다면 재부팅해야 한다.

~]# reboot

단계-3 하드 디스크 디바이스 파일을 확인한다.

~]# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

*표시 생략*

sdc 8:32 0 20G 0 disk

sdd 8:48 0 20G 0 disk

sde 8:64 0 20G 0 disk

sdf 8:80 0 20G 0 disk

새 논리 볼륨을 만드는 예시

단계-1 /dev/sdc, /dev/sdd, /dev/sde 하드 디스크를 각각 20GB로 단일 파티셔닝하고 파티션유형을 LVM 유형(8e)으로 변경한다.

~]# fdisk /dev/sdc

Command (m for help): n

Partition type

p primary (0 primary, 0 extended, 4 free)

e extended (container for logical partitions)

Select (default p): *Enter\_Key*

Using default response p.

Partition number (1-4, default 1): *Enter\_Key*

First sector (2048-41943039, default 2048): *Enter\_Key*

Last sector, +sectors or +size{K,M,G,T,P} (2048-41943039, default 41943039): *Enter\_Key*

Created a new partition 1 of type 'Linux' and of size 20 GiB.

Command (m for help): p

*표시 생략*

Device Boot Start End Sectors Size Id Type

/dev/sdb1 2048 41943039 41940992 20G 83 Linux

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

*표시 생략*

Device Boot Start End Sectors Size Id Type

/dev/sdc1 2048 41943039 41940992 20G 8e Linux LVM

Command (m for help): w

단계-2 파티셔닝 추가나 변경시 커널을 업데이트한다.

~]# partprobe

또는 RHEL 8.0부터 가능

~]# udevadm settle

단계-3 /dev/sdc1, /dev/sdd1, /dev/sde1 디스크를 Physical volume으로 정의한다.

[root@linux-1 ~]# pvcreate /dev/sdc1 /dev/sdd1 /dev/sde1

Physical volume "/dev/sdc1" successfully created.

Physical volume "/dev/sdd1" successfully created.

Physical volume "/dev/sde1" successfully created.

참고 : 위 명령을 다음과 같이 축약할 수 있다.

[root@linux-1 ~]# pvcreate /dev/{sdc1,sdd1,sde1}

Physical volume "/dev/sdc1" successfully created.

Physical volume "/dev/sdd1" successfully created.

Physical volume "/dev/sde1" successfully created.

단계-4 /dev/sdc1, /dev/sdd1, /dev/sde1 볼륨을 대상으로 vg1이라는 볼륨 그룹을 생성한다.

[root@linux-1 ~]# vgcreate vg1 /dev/{sdc1,sdd1,sde1}

Volume group "vg1" successfully created

단계-5 /dev/vg1의 정보를 화면에 표시하여 각종 용량들을 확인한다.

[root@linux-1 ~]# vgdisplay /dev/vg1

--- Volume group ---

*표시 생략*

VG Size <59.99 GiB

PE Size 4.00 MiB

Total PE 15357

Alloc PE / Size 0 / 0

Free PE / Size 15357 / <59.99 GiB

VG UUID Gj13zH-unQc-A0wt-XZQL-MNeb-ThL8-B7YkMg

참고 : PE란? Physical Extent

* PE는 물리볼륨(PV)의 블록(block)단위 이다.
* 물리볼륨(VG)의 PE가 볼륨그룹으로 묶여 LV로 구성된다면 PE는 LE(Logical Extent)로 변환된다.(PE=LE)
* PE의 기본 Size는 4MB(4096KB)로 VG를 생성할때 PE Size를 변경할 수 있다.
* VG가 생성된 이후로는 PE Size를 변경할 수 없기 때문에 적절하게 구성하여야 한다.
* VG에서 PE의 크기를 변경하는 옵션은 -s이다.

PE 설정 예시

[root@linux-1 ~]# vgcreate vg1 -s 8G /dev/{sdc1,sdd1,sde1}

[root@linux-1 ~]# vgcreate vg1 -s 16G /dev/{sdc1,sdd1,sde1}

단계-6 /dev/vg1을 기반으로 LVM을 lvm1과 lvm2 이름으로 각각 생성한다.

[root@linux-1 ~]# lvcreate -n lvm1 -l 50%FREE /dev/vg1

Logical volume "lvm1" created.

[root@linux-1 ~]# lvcreate -n lvm2 -l 100%FREE /dev/vg1

Logical volume "lvm2" created.

단계-7 lvm1과 lvm2를 각각 ext4와 xfs로 포맷한다.

~]# mkfs.ext4 /dev/vg1/lvm1

*표시 생략*

~]# mkfs.xfs /dev/vg1/lvm2

*표시 생략*

단계-8 시스템의 재부팅 후에도 지속(영구)적으로 마운트 되도록 설정한다.

~]# mkdir -pv /mnt/{lv1.lv2}

~]# vim /etc/fstab

*표시 생략*

/dev/vg1/lvm1 /mnt/lv1 ext4 defaults 0 0

/dev/vg1/lvm2 /mnt/lv2 xfs defaults 0 0

Esc🡪 :wq

단계-5 /etc/fstab에 설정된 정보를 근거로 마운트 한다.

~]# mount -a

기존 논리 볼륨을 확장하는 예시

시나리오

* 기존 논리 볼륨(/dev/vg1/lvm1과 /dev/vg1/lvm2)을 각각 5GB씩 확장한다.

단계-1 확장에 사용될 하드 디스크를 준비한다.(이전 단계-1을 참조하여 설정한다)

~]# fdisk /dev/sdf

*표시 생략*

Device Boot Start End Sectors Size Id Type

/dev/sdf1 2048 41943039 41940992 20G 8e Linux LVM

Command (m for help): w

단계-2 파티셔닝 추가나 변경시 커널을 업데이트한다.

~]# partprobe

또는 RHEL 8.0부터 가능

~]# udevadm settle

단계-3 /dev/sdf1 디스크를 Physical volume으로 정의한다.

[root@linux-1 ~]# pvcreate /dev/sdf1

Physical volume "/dev/sdf1" successfully created.

단계-3 기존 vg1에 /dev/sdf1의 physical volume을 확장한다.

[root@linux-1 ~]# vgextend vg1 /dev/sdf1

Volume group "vg1" successfully extended

단계-4 확장된 결과를 vgdisplay 명령으로 확인한다.

[root@linux-1 ~]# vgdisplay /dev/vg1

--- Volume group ---

표시 생략

VG Size 79.98 GiB

PE Size 4.00 MiB

Total PE 20476

Alloc PE / Size 15357 / <59.99 GiB

Free PE / Size 5119 / <20.00 GiB

단계-5 lvm1과 lvm2를 각각 5GB 확장한다.

[root@linux-1 ~]# lvextend -L +5G -r /dev/vg1/lvm1

Size of logical volume vg1/lvm1 changed from 29.99 GiB (7678 extents) to 34.99 GiB (8958 extents).

Logical volume vg1/lvm1 successfully resized.

resize2fs 1.45.4 (23-Sep-2019)

Filesystem at /dev/mapper/vg1-lvm1 is mounted on /mnt/lv1; on-line resizing required

old\_desc\_blocks = 4, new\_desc\_blocks = 5

The filesystem on /dev/mapper/vg1-lvm1 is now 9172992 (4k) blocks long.

[root@linux-1 ~]# lvextend -L +5G -r /dev/vg1/lvm2

중요 : lvextend 명령 사용시에 -r 옵션을 누락했을 경우에는 다음과 같은 명령을 실행해줘야만 확장된 용량이 적용된다.

ext3, ext4 등으로 포맷된 LVM일 경우 예시

~]# resize2fs /dev/vg1/lvm1

xfs로 포맷된 LVM일 경우 예시

~]# xfs\_growfs /dev/vg1/lvm2