

ucloud server

NFS 설정 가이드

v1.1

2013. 6

목차

1. 개 요.....	4
1.1. Linux NFS 개요.....	4
1.2. ucloud NFS 구성.....	5
2. NFS 서버 구성.....	6
2.1. NFS 서버 VM 생성.....	6
2.1.1. VM 생성.....	6
ucloud Marketplace 의 상품 신청 기능을 이용하여 신규 NFS Server VM 을 생성한다.	6
2.2. 디스크 추가.....	7
2.2.1. 디스크 생성.....	7
2.2.2. 디스크 파티션 설정.....	7
2.2.3. LVM 설정.....	9
2.2.4. 포맷 및 마운트.....	10
2.2.5. fstab 수정.....	11
2.3. NFS 서버 설정.....	11
2.3.1. exports 파일 설정.....	11
2.3.2. exports 파일 예제.....	11
2.3.3. 데몬 재시작.....	11
2.4. NFS 서버 옵션.....	12
2.5. 디스크 확장.....	12
2.5.1. 디스크 추가 및 파티션.....	12
2.5.2. 디스크 사이즈 변경.....	12
3. NFS 클라이언트 설정.....	14
3.1. NFS Mount.....	14

3.2. NFS Mount 예제.....	14
3.3. Mount 옵션.....	14
3.4. 자동 마운트 설정	14

1. 개요

본 문서는 ucloud server 상에서 VM 기반의 NFS Server 를 구성하려는 사용자에게 제공되는 가이드 문서이다.

1.1. Linux NFS 개요

NFS(Network File System)란 리눅스 머신에서 이더넷 기반으로 동작하는 가장 기본적인 공유 파일 시스템 중의 하나로 서버 / 클라이언트 모델로 동작하는 스토리지 프로토콜이다.

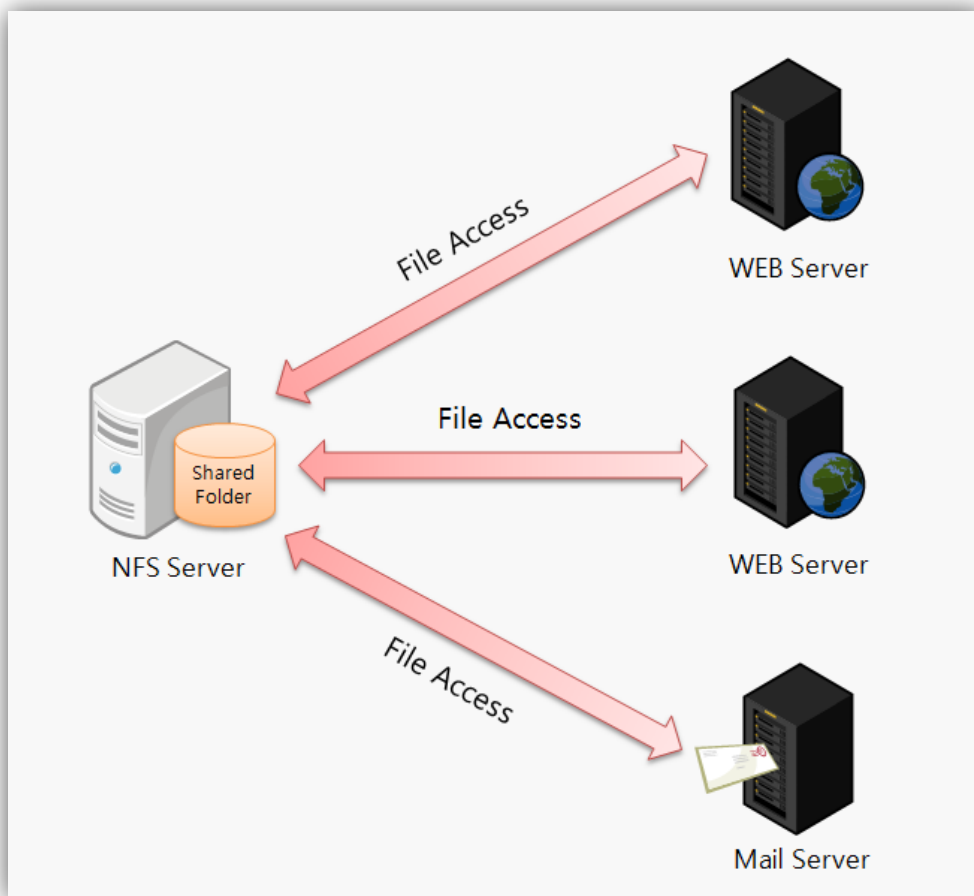


그림 1. NFS 를 이용한 공유 스토리지 구성도

각 클라이언트들이 NFS 서버의 공유 폴더(Export Path)를 마운트하게 되면 자신의 로컬 파일 시스템을 사용하는 것처럼 여러 클라이언트와 파일을 공유하여 사용할 수 있다.

1.2. ucloud NFS 구성

ucloud Marketplace 의 상품 신청 기능을 이용하여 NFS Server 를 쉽고 빠르게 구성할 수 있다.

URL: https://ucloudbiz.olleh.com/portal/ktcloudportal.epc.productintro.ucloud_server_image.html



그림 2. ucloud Marketplace 화면

생성된 NFS Server VM 은 같은 계정 내 VM 에서 접속하여 사용할 수 있다.

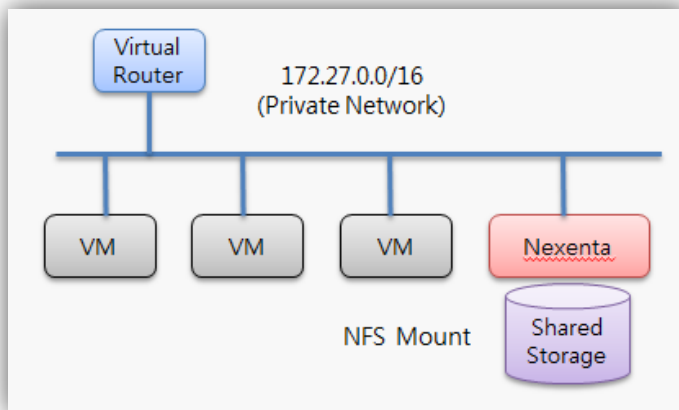


그림 3. ucloud NFS 시스템 구성도

2. NFS 서버 구성

2.1. NFS 서버 VM 생성

2.1.1. VM 생성

ucloud Marketplace의 상품 신청 기능을 이용하여 신규 NFS Server VM을 생성한다.

ucloud server :: 서버 신청

01 운영체제 선택 02 서버 선택 03 신청 내역 확인

? 용어설명

Availability Zone

분류	이미지명	종류	가격	
			월요금제	시간요금제
머신이미지	Centos + NFS	Centos 5.4 64bit	무료	무료

cloud MAS 사용여부 예 아니오

그림 4. ucloud Marketplace 에서 NFS VM 생성 화면

2.2. 디스크 추가

2.2.1. 디스크 생성

ucloud server 관리 콘솔에서 NFS Server VM 에 추가 디스크를 생성한다.



적용 서버	서버 : server01 (4784f32b-2644-47e9-8cd5-76799cc910e5)
스토리지 가격	10GB당 700원
스토리지 용량	10GB ▼
스토리지 명	<input type="text"/> 중복체크 * 영문만 입력 가능합니다. (총 25Bbytes)
월 결제 금액	700원/월 (부가세 별도)

그림 5. ucloud server 관리 콘솔에서 추가 디스크 신청 화면

2.2.2. 디스크 파티션 설정

추가한 디스크를 확인한다.

```
# fdisk -l

Disk /dev/xvda: 21.4 GB, 21474836480 bytes
255 heads, 63 sectors/track, 2610 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/xvda1    *           1           13        104391   83  Linux
/dev/xvda2             14          2610       20860402+  8e  Linux LVM

Disk /dev/xvdb: 85.8 GB, 85899345920 bytes // 추가 디스크 확인
255 heads, 63 sectors/track, 10443 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

Disk /dev/xvdb doesn't contain a valid partition table
```

디스크 파티션을 생성한다.

```
# fdisk /dev/xvdb

Command (m for help): n // 새 파티션 생성
Command action
    e extended
    p primary partition (1-4)
p // Primary 파티션으로 선택

Partition number (1-4): 1 // 파티션 번호(1) 선택
First cylinder (1-10443, default 1): // 디스크 전체를 생성하는 파티션에 할당
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-10443, default 10443):
Using default value 10443
```

파티션 타입을 설정한다.

```
Command (m for help): t // 파티션 타입 설정
Selected partition 1
Hex code (type L to list codes): 8e // LVM 타입 선택
Changed system type of partition 1 to 8e (Linux LVM)
```

생성한 파티션을 확인한다.

```
Command (m for help): p // 생성한 파티션 확인

Disk /dev/xvdb: 85.8 GB, 85899345920 bytes
255 heads, 63 sectors/track, 10443 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

    Device Boot      Start         End      Blocks   Id  System
/dev/xvdb1            1         10443     83883366   83  Linux LVM
```

설정된 파티션 정보를 저장하고 종료한다.

```
Command (m for help): w // 파티션 정보 저장
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```


2.2.3. LVM 설정

디스크를 EXT3 와 같은 일반적인 리눅스 파일 시스템으로 설정해도 상관 없지만, 디스크 추가 확장이 불가능해진다.

하지만 리눅스의 LVM(Logical Volume Manager)을 이용하게 되면 볼륨 크기를 동적으로 증가시키는 것이 가능하기 때문에 추가 디스크 설정 시 LVM 을 이용하는 것이 좋다.

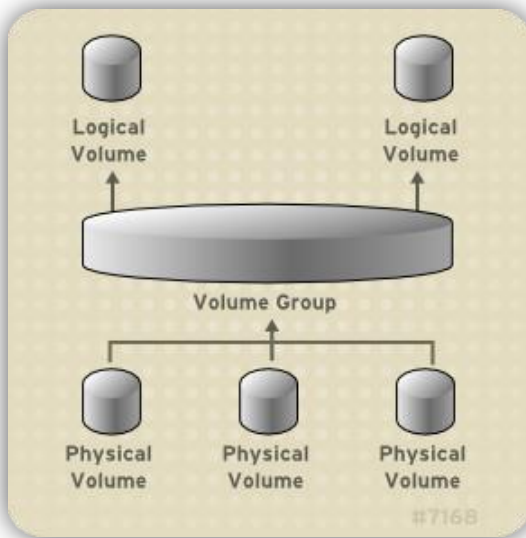


그림 6. LVM 구성도

(출처: <http://docs.redhat.com>)

PV(Physical Volume)를 생성한다.

```
# pvcreate /dev/xvdb1 // PV 생성
# pvs // PV 조회
  PV          VG          Fmt  Attr PSize  PFree
  /dev/xvda2 VolGroup00 lvm2 a-- 19.88G  0
  /dev/xvdb1          lvm2 a-- 80.00G 80.00G
```

VG(Volume Group)를 생성한다.

```
# vgcreate DataVol01 /dev/xvdb1 // VG 생성
# vgs // VG 조회
  VG          #PV #LV #SN Attr   VSize  VFree
  DataVol01   1  0  0 wz--n- 80.00G 80.00G
  VolGroup00  1  2  0 wz--n- 19.88G  0
```

LV(Logical Volume)를 생성한다.

```
# lvcreate -l 100%FREE -n NFSLV DataVol01 // LV 생성
# lvs // VG 조회
LV      VG          Attr  LSize  Origin Snap%  Move Log Copy%  Convert
NFSLV   DataVol01  -wi-a- 80.00G
```

2.2.4. 포맷 및 마운트

생성한 LV 를 EXT3 파일 시스템으로 포맷한다.

```
# mkfs.ext3 /dev/DataVol01/NFSLV // 생성한 LV 포맷
mke2fs 1.39 (29-May-2006)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
10485760 inodes, 20970496 blocks
1048524 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=4294967296
640 block groups
32768 blocks per group, 32768 fragments per group
16384 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424, 20480000

Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 31 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

포맷한 볼륨을 NFS Server VM 에 마운트 한다.

```
# mkdir /data // 추가 디스크를 마운트할 /data 디렉토리 생성
# mount /dev/DataVol01/NFSLV /data // 볼륨 마운트
```

2.2.5. fstab 수정

NFS Server VM 이 재부팅 되더라도 추가한 디스크가 자동으로 마운트 되도록 **/etc/fstab** 파일에 다음 내용을 추가한다.

```
/dev/DataVol01/NFSLV /data ext3 defaults 1 1
```

2.3. NFS 서버 설정

2.3.1. exports 파일 설정

/etc/exports 파일을 이용하여 NFS 서버 설정을 할 수 있다.

```
# vi /etc/exports
```

2.3.2. exports 파일 예제

예제 1) 1 개 이상의 웹 서버에서 읽기 전용의 이미지 파일을 공유하여 사용하기 위한 NFS 서버 구성

```
/var/www/img *(ro,all_squash)
```

예제 2) 읽기/쓰기가 빈번하게 일어나는 미디어 파일을 서로 공유하여 사용하기 위한 NFS 서버 구성

```
/data 172.27.0.0/16(rw,no_root_squash)
```

2.3.3. 데몬 재시작

위의 exports 파일 설정이 완료되면 NFS 데몬을 재시작하여 설정을 적용한다.

```
#service nfs restart
```

2.4. NFS 서버 옵션

exports 파일 설정 시 다음 옵션을 이용하여 다양한 환경을 구성할 수 있다.

옵션	설명
ro	읽기 전용으로 공유한다. (기본)
rw	읽기/쓰기 모드로 공유한다.
root_squash	클라이언트의 root 를 익명 사용자(nobody)로 매핑한다. (기본)
no_root_squash	클라이언트의 root 를 서버의 root 로 매핑한다.
all_squash	모든 사용자를 익명 사용자(nobody)로 매핑한다.
sync	클라이언트와 서버간 동기적 통신을 한다. (기본, 안전성 향상)
async	클라이언트와 서버간 비동기 통신을 한다. (속도 향상)
secure	마운트 요청 시 포트를 1024 이하로 한다. (1024 이하 포트는 root 만 설정 가능, 기본)
insecure	마운트 요청 시 1024 포트 이상도 허용한다.

2.5. 디스크 확장

만약 기존에 쓰던 디스크의 여유 공간이 부족해 디스크 확장이 필요한 경우 다음 절차를 통해 디스크 크기를 증가시킬 수 있다.

2.5.1. 디스크 추가 및 파티션

먼저 ucloud server 관리 콘솔에서 추가 디스크를 신청 후 LVM 방식으로 파티션을 선언한다. (문서 내 2.2.1 / 2.2.2 참조)

2.5.2. 디스크 사이즈 변경

새롭게 추가 된 디스크를 PV 로 생성한다.

```
# pvcreate /dev/xvdc1 // PV 생성
# pvs // PV 조회
  PV          VG          Fmt  Attr PSize  PFree
  /dev/xvda2  VolGroup00 lvm2 a-- 19.88G  0
  /dev/xvdb1  DataVol01  lvm2 a-- 80.00G  0
  /dev/xvdc1          lvm2 a-- 80.00G 80.00G
```

기존에 사용하던 VG 에 새롭게 생성한 PV 를 추가시킨다.

```
# vgextend DataVol01 /dev/xvdc1 // VG 에 PV 추가
# vgs // VG 조회
  VG          #PV #LV #SN Attr   VSize  VFree
```

```
DataVol01 2 1 0 wz--n- 159.99G 80.00G
VolGroup00 1 2 0 wz--n- 19.88G 0
```

VG 크기가 증가한 것을 확인할 수 있다.

NFS 데몬을 중지시키고, NFS 서버에 마운트된 볼륨을 언마운트 시킨다.

```
# service nfs stop // NFS 데몬 정지
# umount /data // 볼륨 언마운트
```

LV 크기를 변경한다.

```
# lvextend -L+80G /dev/DataVol01/NFSLV
```

EXT3 파일 시스템의 크기도 증가 시킨다.

```
# e2fsck -f /dev/DataVol01/NFSLV // 변경된 파일 시스템 체크
# resize2fs /dev/DataVol01/NFSLV // 파일 시스템 크기 변경
```

사이즈가 변경된 볼륨을 다시 마운트 후 NFS 데몬을 시작한다.

```
# mount /dev/DataVol01/NFSLV /data // 볼륨 마운트
# service nfs start // NFS 데몬 시작
```

3. NFS 클라이언트 설정

3.1. NFS Mount

서버에서의 설정이 끝나면 클라이언트에서는 서버에서 exports 한 NFS Path 를 마운트 할 수 있다.

```
#mount -t nfs <nfs_server_ip>:<mount_path> <mount_point> -o <options>
```

3.2. NFS Mount 예제

IP 가 172.27.0.5 인 NFS 서버의 /data 를 읽기 전용으로 마운트 하는 방법

```
# mount -t nfs 172.27.0.5:/data /mnt -o soft,retrans=10
```

3.3. Mount 옵션

클라이언트가 NFS Mount 시에 사용되는 옵션이다.

옵션	설명
rsize=n	NFS 서버로부터 읽어오는 바이트 수를 지정한다. (기본 1024)
wsize=n	NFS 서버로 쓰기를 할 때 사용하는 바이트 수를 지정한다. (기본 1024)
timeo=n	타임 아웃이 발생 후 재전송 간격
retrans=n	타임 아웃 발생 시 재전송 시도 횟수 (기본 3)
port=n	NFS 서버와 연결할 때 포트번호 지정, 기본적으로 NFS 서버와 질의 후 설정한다.
fg	첫 번째 마운트 시도 시 타임 아웃되면 바로 중단한다. (기본)
bg	첫 번째 마운트 시도 시 타임 아웃이 되더라도 백그라운드에서 계속 마운트 시도를 한다.
hard	서버가 응답할 때까지 계속 마운트를 시도하며, 응답이 없는 경우에도 umount 명령으로 마운트를 취소할 수 없다. (기본)
soft	타임 아웃 발생 시 retrans 값만큼만 재전송을 시도한다. umount 명령을 통해 마운트를 취소할 수도 있다.

3.4. 자동 마운트 설정

클라이언트가 재부팅이 되더라도 자동으로 마운트 되도록 하기 위해서는 /etc/fstab 에 다음 설정을 추가한다.

```
172.27.0.5:/data /mnt nfs defaults 0 0
```