

- Ch_1 _ APM 웹서버 설치
- Ch_2 _ 워드프레스 웹사이트 구축
- Ch_3 _ 클라우드 서비스 구축
- Ch_4 _ vsftpd, proftpd 구축
- Ch_5 _ NFS 서버 구축

- APM 개요

- 리눅스를 가장 많이 활용하는 분야 중에 하나가 바로 웹(Web) 서버
- 가장 안정적이고 유명한 Apache 웹 서버
- APM = Apache 웹 서버 + 프로그래밍 언어 PHP + 데이터베이스 MariaDB(이전 MySQL)
- 리눅스 환경에서 사용될 경우에 LAPM(Linux, Apache, PHP, MariaDB)이라고도 부름
- APM이라는 소프트웨어는 존재하지 않으며 이 3 가지가 서로 잘 연동되어 운영되도록 만든 환경을 APM 이라고 부르는 것

```
ubuntu@server:~$ sudo apt -y install lamp-server^
[sudo] ubuntu 암호 :
패키지 목록을 읽는 중입니다 ... 완료
아주 작은 공간을 만드는 중입니다
```

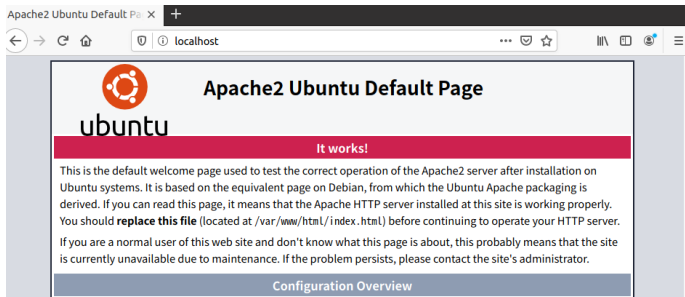
➔ APM 서버 패키지 설치 (lamp-server^)

```
ubuntu@server:~$ sudo systemctl restart apache2
ubuntu@server:~$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV s
install.
Executing: /lib/systemd/systemd-sysv-install enabl
ubuntu@server:~$ sudo systemctl status apache2
Unknown operation status.
ubuntu@server:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.s
   Active: active (running) since Mon 2022-03-28
   Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 12588 (apache2)
```

```
ubuntu@server:~$ sudo systemctl restart mysql
ubuntu@server:~$ sudo systemctl enable mysql
Synchronizing state of mysql.service with SysV
stall.
Executing: /lib/systemd/systemd-sysv-install e
ubuntu@server:~$ sudo systemctl status mysql
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql
   Active: active (running) since Mon 2022-0
   Main PID: 12795 (mysqld)
   Status: "Server is operational"
```

➔ 설치된 apache2, mysql 을 설정 (restart, enable) 후 확인 (status)

➤ 리눅스 버전에 따라 mysql, mariadb 중 하나가 설치됨

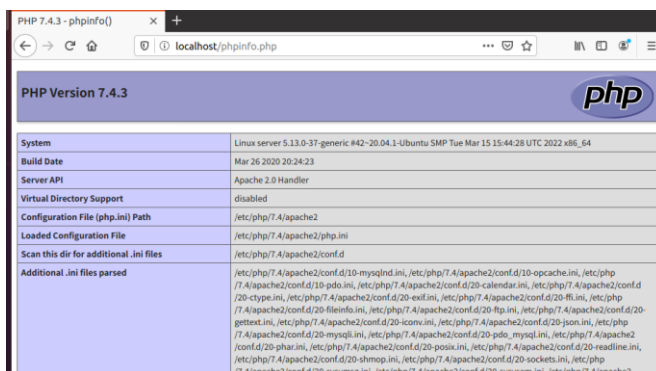


➔ 웹페이지 'localhost' 접속으로 웹 서버 정상 작동 확인

```
1
2 <?php phpinfo(); ?>
3
```

➔ /var/www/html 디렉터리에 phpinfo.php 파일 생성 후 수정

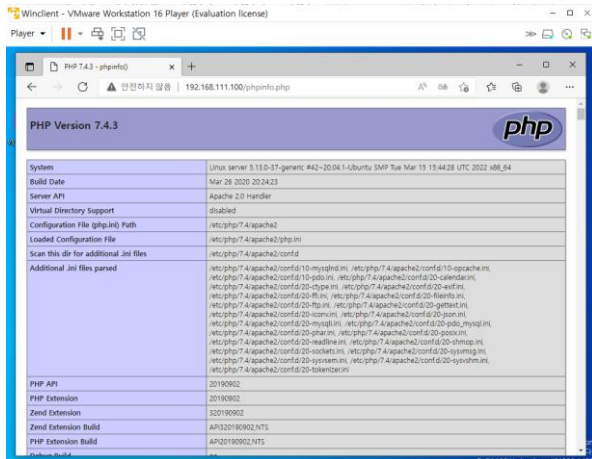
➤ <?php phpinfo(); ?> // 웹서버의 php 정보를 보여주는 코드



➔ localhost/phpinfo.php 에 접속하여 php 작동 확인

```
root@server:/var/www/html# ufw allow 80
규칙이 추가되었습니다
규칙이 추가되었습니다 (v6)
root@server:/var/www/html#
```

➔ 80 번 포트 방화벽 허용



➔ 외부(Winclient) 에서도 접속 확인 가능

● 웹서버 워드프레스 웹사이트 구축

➤ 데이터베이스 생성

- ✓ mysql 진입
- ✓ CREATE DATABASE wp_db; // wp_db 데이터베이스 생성
- ✓ CREATE USER wp_user@localhost IDENTIFIED BY '1234' // 사용자생성
- ✓ GRANT ALL ON wp_db.* TO wp_user@localhost; // 사용자 권한 설정

```
ubuntu@server:~$ sudo wget https://ko.wordpress.org/wordpress-5.2.4-ko_KR.tar.gz
--2022-03-28 10:53:01-- https://ko.wordpress.org/wordpress-5.2.4-ko_KR.tar.gz
ko.wordpress.org (ko.wordpress.org) 해석 중 ... 198.143.164.252
```

➤ 워드프레스 설치 // 우분투 제공 x 별도의 오픈 소스로 제공

```
ubuntu@server:~$ ls -l word*
-rw-r--r-- 1 root root 11822176 11월  5  2019 wordpress-5.2.4-ko_KR.tar.gz
ubuntu@server:~$ tar xzf word*
ubuntu@server:~$ ls
wordpress  wordpress-5.2.4-ko_KR.tar.gz
wordpress  공개 문서 비디오 음악
wordpress-5.2.4-ko_KR.tar.gz 다운로드 바탕화면 사진 웹플릿
ubuntu@server:~$
```

➔ 설치 확인 후 압축 해제

- ✓ 이후 wordpress 디렉토리를 /var/www/html 로 이동(mv)

```
root@server:/var/www/html# chmod 707 wordpress/
root@server:/var/www/html# chown -R www-data:www-data wordpress
```

➔ 외부권한 설정과 소유자 변경 (웹 서버에 들어오는 사람으로)

```
21 // -- MySQL settings - You can get
22 /** The name of the database for W
23 define( 'DB_NAME', 'wp_db' );
24
25 /** MySQL database username */
26 define( 'DB_USER', 'wp_user' );
27
28 /** MySQL database password */
29 define( 'DB_PASSWORD', '1234' );
30
31 /** MySQL hostname */
32 define( 'DB_HOST', 'localhost' );
```

➔ wp-config.php 파일 수정

- ✓ /var/www/html/wordpress 디렉터리에 wp-config-sample.php 파일을 wp-config.php 이름으로 복사 (cp)
- ✓ 이후 복사된 wp-config.php 파일 내용 수정 (23, 26, 29 행)

```
10
11 ServerAdmin webmaster@localhost
12 DocumentRoot /var/www/html/wordpress
13
14 # Available loglevels: trace8, ..., t
```

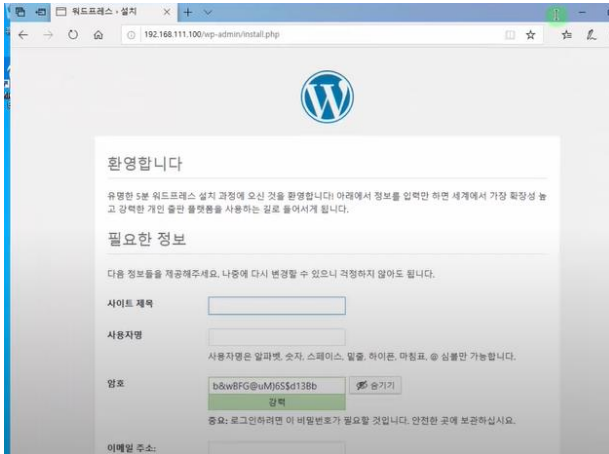
➔ /etc/apache2/sites-enabled/000-default.conf 파일 수정 (초기 홈페이지 변경)

```
175
176 #<Directory /srv/>
177 # Options Indexes FollowSymLinks
178 # AllowOverride None
179 # Require all granted
180 #</Directory>
181
182 <Directory /var/www/html/wordpress>
183     Option Indexes FollowSymLinks
184     AllowOverride All
185     Require all granted
186 </Directory>
187
188
189
190
191 # AccessFileName: The name of the file to look for
192 # for additional configuration directives. See als
193 # directive.
194 #
```

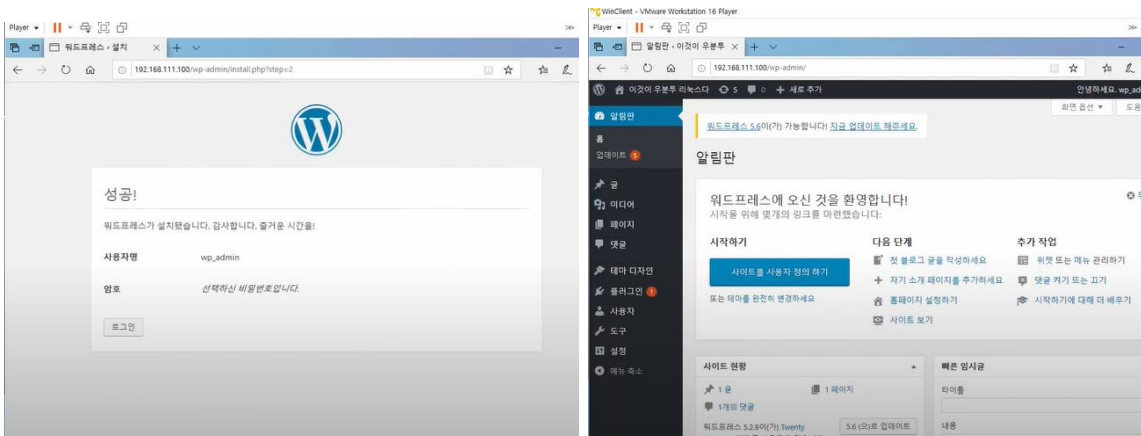
➔ /etc/apache2/apache2.conf 파일 수정

- ✓ 180 행 아래 내용 추가

- ✓ 워드프로세스 폴더에 접근하기위한 권한을 제공
- ✓ 이후 서비스 재시작(systemctl restart apache2)



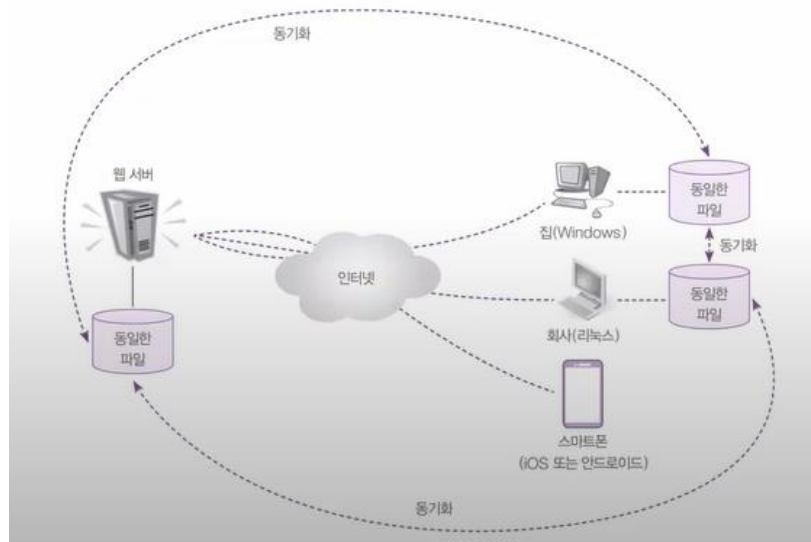
➔ 이후 접속하여 정상 작동 확인



➔ 웹서버 워드프레스 웹사이트 구축 완료

● 클라우드 서비스 구축

- 네이버의 N 드라이브, Microsoft의 OneDrive, Google의 Google 드라이브 등의 서비스를 말함
- 클라우드 서비스 개념도(한 명의 사용자만 표현됨)



- server 초기화 이후 apt apache2, mariadb-server, mariadb-client 설치
- apache2 와 mariadb restart 및 enable
- ufw allow(방화벽 포트) 80, 3306, 443 추가

```
ubuntu@Server:~$ sudo mysql_secure_installation
```

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

➔ mariadb 초기화

```
MariaDB [(none)]> CREATE DATABASE cloud_db;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> GRANT ALL ON cloud_db.* TO cloud_user@localhost IDENTIFIED BY '1234';
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> █
```

➔ mariadb DATABASE 와 사용자 새로 생성

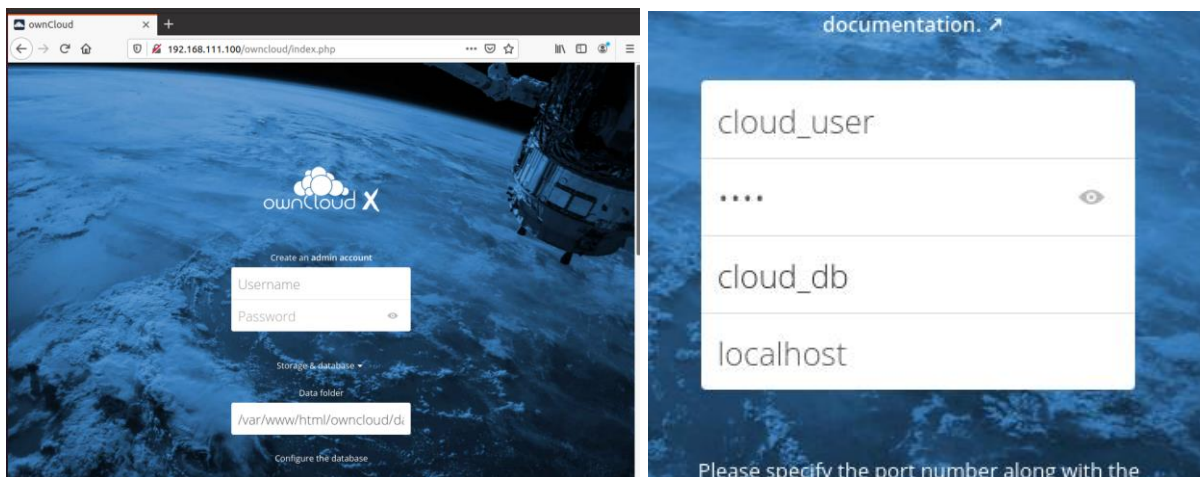
- (mkdir) php72 디렉터리 생성 후 압축 해제
- wget 명령어로 <http://dw.hanbit.co.kr/ubuntu/20.04/php72.tar.gz> 파일 다운로드
- dpkg -i *.deb 명령어로 deb 파일 한번에 설치

```
root@Server:~/바탕화면/php72# cd /var/www/html/
root@Server:/var/www/html#
root@Server:/var/www/html# wget https://download.owncloud.org/community/owncloud-10.4.1.zip
--2022-03-28 14:55:07-- https://download.owncloud.org/community/owncloud-10.4.1.zip
```

➔ /var/www/html 디렉터리에서 파일 다운로드 후 압축 해제

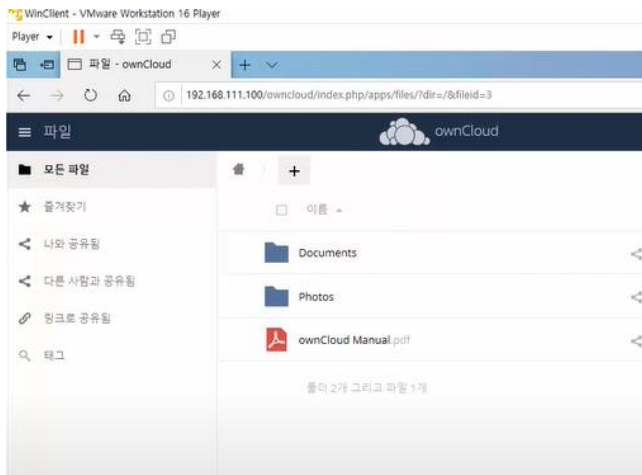
```
index.html owncloud owncloud-10.4.1.zip
root@Server:/var/www/html#
root@Server:/var/www/html# mkdir owncloud/data
root@Server:/var/www/html#
root@Server:/var/www/html# chown -R www-data:www-data owncloud
root@Server:/var/www/html#
root@Server:/var/www/html# chmod -R 755 owncloud
root@Server:/var/www/html#
root@Server:/var/www/html#
```

➔ 생성된 owncloud 디렉터리에 새 data 디렉터리 생성 후 권한 설정



➔ 시스템 재시작 후(적용) 192.168.111.100/owncloud 접속으로 정상 가동 확인

✓ 생성했던 db 명과 사용자명, 패스워드를 입력



➔ 생성 완료

● FTP 개요

- FPT(File Transfer Protocol)는 파일을 전송하기 위한 서비스
- 웹에서 FTP의 고유 기능인 파일 전송을 편리하게 할 수 있게 되어서 예전보다 인기가 많이 떨어짐
- 파일 전송 자체를 위해서는 성능이 뛰어나
- vsftpd는 우분투에서 제공해 줌
 - ✓ vsftpd(Very Secure FTPD)는 우분투에서 기본적으로 제공되며, 리눅스와 유닉스 환경에서 보안성과 성능이 우수한 FTP 서버로 인정받고 있음
 - ✓ proftpd는 주로 대형 사이트에서 오랫동안 인기가 많았던 ftp 서버
- apt install vsftpd 명령어로 패키지 설치


```

25 anonymous_enable=YES
26 #
27 # Uncomment this to allow local users to log in.
28 local_enable=YES
29 #
30 # Uncomment this to enable any form of FTP write comman
31 write_enable=YES
32 #
33 # Default umask for local users is 077. You may wish to
34 # if your users expect that (022 is used by most other
35 #local_umask=022
36 #
37 # Uncomment this to allow the anonymous FTP user to upl
38 # has an effect if the above global write enable is act
39 # obviously need to create a directory writable by the
40 anon_upload_enable=YES
41 #
42 # Uncomment this if you want the anonymous FTP user to
43 # new directories.
44 anon_mkdir_write_enable=YES
45 #

```

➔ /etc/vsftpd.conf 파일 수정

✓ 25,40,44 행 익명 사용자 관련 권한 허용

```

root@Server:/srv/ftp# chmod 777 pub/
root@Server:/srv/ftp#
root@Server:/srv/ftp# cp /boot/vmlinuz-* pub/file1
root@Server:/srv/ftp# ls
pub
root@Server:/srv/ftp#


```

➔ 외부 사용자가 사용하는 디렉터리(/srv/ftp)에 pub 디렉터리 생성 후 배포할 파일 생성(cp 로 아무파일 가져옴)

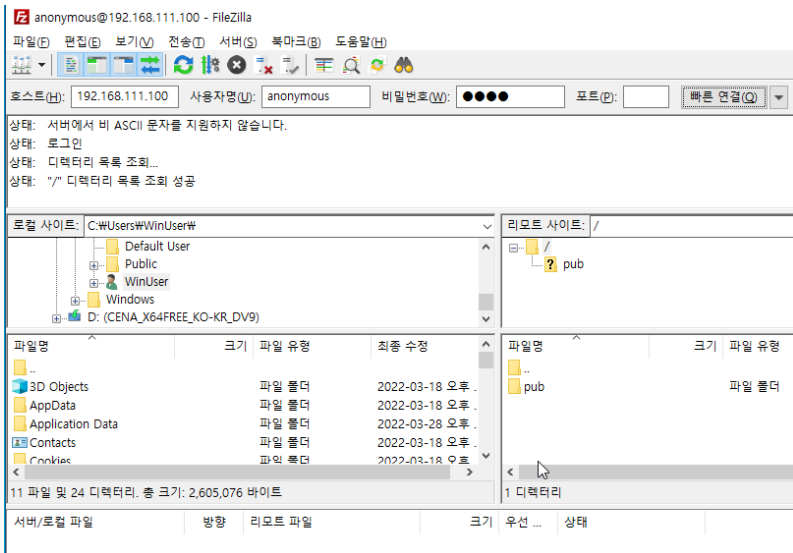
✓ systemctl restart, enable 로 시스템 재시작 후 상태확인(status)

✓ 이후 방화벽 비활성화 (ufw disable)

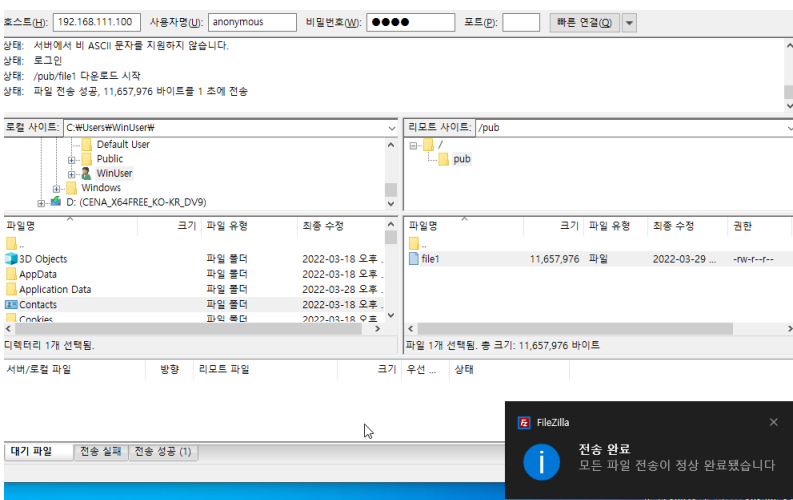
▼ 오늘 (1)

 FileZilla_3.48.1_win32_sponsored-setup 2022-03-28 오후

➔ Winclient 에서 vsftpd 에 접속할 프로그램 설치



➔ filezilla server 에 anonymous 사용자로 연결 성공



➔ 업로드 및 다운로드 가능

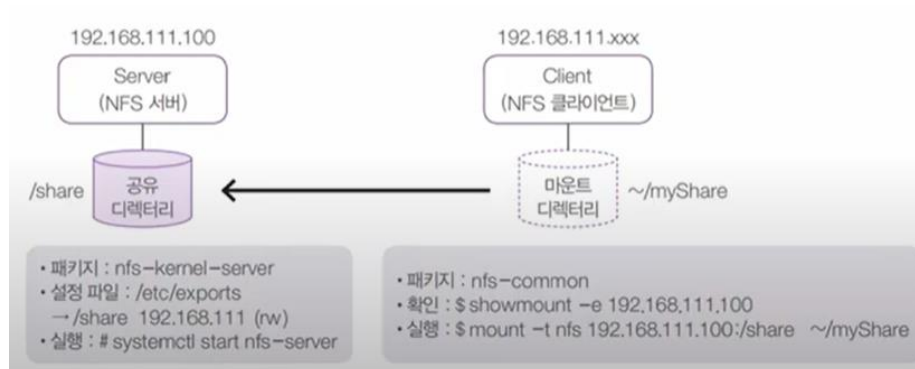
```
ubuntu@server-b:~$ sudo cp /boot/vmlinuz-* file1
ubuntu@server-b:~$ ls
file1
ubuntu@server-b:~$ lftp 192.168.111.100
lftp 192.168.111.100:~>
```

➔ server(b)에서 연결 확인

- ✓ apt install lftp 패키지 설치
- ✓ cp /boot/vmlinuz-* file1 전송테스트 파일 만들기
- ✓ lftp [연결할주소] server 에 연결

● NFS 서버 구현

- Linux(Unix) 컴퓨터끼리 저장 공간을 공유할 수 있도록 해 주는 시스템이 NFS(Network File System)
- NFS 서버 구현의 개요도



```
ubuntu@Server:~$ sudo apt install nfs-kernel-server -y
[sudo] ubuntu 알 호 :
패 키 지 목 록 을 읽 는 중 이 니 다 ... 완 료
```

➔ apt install nfs-kernel-server 패키지 설치

```
# /srv/nfs4      gss/krb5i(rw,sync,f
# /srv/nfs4/homes gss/krb5i(rw,sync,n
#
/share          192.168.111.*(rw,sync)█
```

➔ /etc/exports 파일 수정 // 하단 내용 추가

```
ubuntu@Server:~$ sudo mkdir /share
ubuntu@Server:~$
ubuntu@Server:~$ sudo chmod 707 /share/
ubuntu@Server:~$
ubuntu@Server:~$ sudo systemctl restart nfs-server
ubuntu@Server:~$ sudo systemctl enable nfs-server
ubuntu@Server:~$ █
```

➔ NFS 서버 구축 완료

- ✓ mkdir // 공유할 디렉터리 생성
- ✓ chmod // 공유 디렉터리 권한 설정
- ✓ systemctl restart, enable // 시스템 재시작

```
ubuntu@Server:~$ sudo exportfs -v
/share          192.168.111.*(rw,wdelay,root_squash,no_subtree_check,sec=sys,rw,secure,root_squash,no_all_squash)
ubuntu@Server:~$ █
```

➔ exportfs -v 명령어로 share 디렉터리가 허용되어있는 모습 확인 가능

✓ 이후 방화벽 끄기 (ufw disable 명령어)

```
ubuntu@client:~$ sudo apt -y install nfs-common
[sudo] ubuntu의 암호:
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다
```

➔ 초기화된 client 에서 nfs-common 패키지 설치 (apt install)

```
ubuntu@client:~$ showmount -e 192.168.111.100
Export list for 192.168.111.100:
/share 192.168.111.*
ubuntu@client:~$
```

➔ server 의 share 와 연결 확인

```
~$
~$ cp /boot/vmlinuz-* /share/chanwoo

ubuntu@client:~$ sudo mount -t nfs 192.168.111.100:/share myShare
[sudo] ubuntu의 암호:
ubuntu@client:~$
ubuntu@client:~$ ls myShare/
ubuntu@client:~$
ubuntu@client:~$ ls myShare/
chanwoo
ubuntu@client:~$
```

➔ server 에서 생성한 chanwoo 파일을 client 에서 공유폴더 mount 하여 확인 가능한 모습

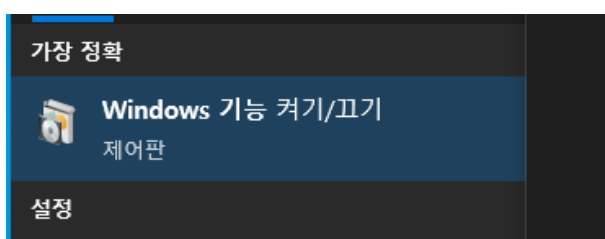
```
ubuntu@client:~$ cd myShare/
ubuntu@client:~/myShare$ touch hongik
ubuntu@client:~/myShare$
```

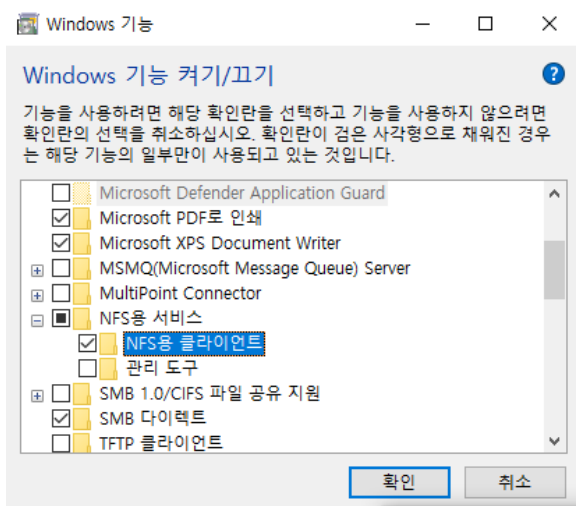
```
ubuntu@server:~$
ubuntu@server:~$ ls /share/
chanwoo hongik
ubuntu@server:~$
```

➔ 반대의 경우도 확인 가능

```
192.168.111.100:/share /home/ubuntu/myShare nfs default 0 0
```

➔ 재 접속시에도 기능 유지를 위해 /etc/fstab 파일 수정



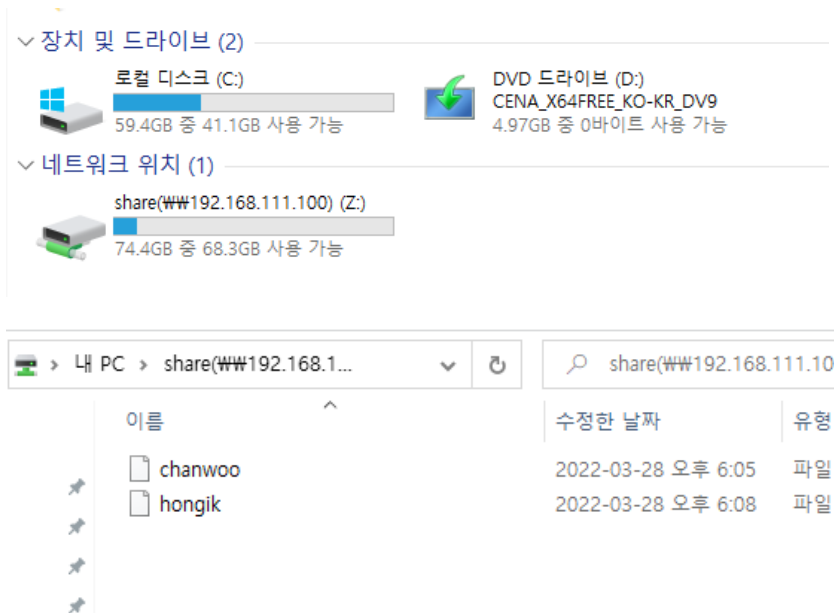


➔ winclient 에 nfs 용 클라이언트 설치

```
C:\Users\winUser>mount 192.168.111.100:/share *
```

Z: 현재 192.168.111.100:/share에 성공적으로 연결되어 있습니다.
명령을 완료했습니다.
C:\Users\winUser>

➔ 윈도우 cmd 로 연결



➔ share 디렉터리와 연결 확인