

<리눅스 Cent OS>

ip addr -> vi /etc/sysconfig/network-scripts/ifcfg-ens33

I키 인서트키 내용 수정할 때
BOOTPROTO 앞에 #붙이고
엔터치고 밑에

```
BOOTPROTO=static
IPADDR=192.168.100.100
NETMASK=255.255.255.0
GATEWAY=192.168.100.2
DNS1=8.8.8.8
:wq
```

rpm -qa | grep []
설치된 것 중에 [] 단어가 들어간 패키지가 설치 되어있는지 확인할 때 사용

vim /etc/selinux/config
enforcing을 disabled로 변경후 재부팅

<※ 복습하기>

root 홈 디렉토리 파일들 모두 삭제
(rm -r 및 rmdir 옵션 사용)
cloud 빈 파일생성(touch cloud)
cloud 1 2 3 4 5 6입력(vim cloud -> 1 2 3 4 5 6)
ab, cd, ef 디렉토리 생성(mkdir ab cd ef)
ab 디렉토리 안에 a안에 b안에 c안에 d안에 e안에 f라는 디렉토리를 생성
(mkdir -p ab/a/b/c/d/e/f)
cisco1 파일을 f 디렉토리 안에 복사(cp cisco1 /ab/a/b/c/d/e/f)
f디렉토리 안의 cisco1파일을 test2라는 이름의 파일로 바꾸시오(mv cisco1 test2)
welcome이라는 public파일 생성(vim public)

계정을 만들 때 public 파일도 배포하도록 설정(cp public /etc/skel)
myhome이라는 그룹을 생성(groupadd myhome)
user10 사용자 설정(useradd user10)
비번 1111설정(passwd user10 -> 1111)
user11 사용자를 생성 myhome이라는 그룹에 속하게 설정(useradd -g myhome user11)
office 그룹 생성(groupadd office)
user10의 그룹을 office로 변경(usermod -g office user10)

[]*표시 : []뒤에 어떤 거이 있더라도 나머지 모두라는 의미

■ find 파일 찾기 명령어

find 명령어

find [경로] [옵션] [조건] [action] : **기본 파일(실행파일 포함)**찾기

[옵션] -name(파일 이름), -user(소유자), -newer(전,후), -perm(허가권(퍼미션)), -size(파일 크기)

[action] -print(디폴트 옵션(기본, 생략가능)), -exec (외부명령 실행)

사용 예

```
# find /etc -name "*.conf"
```

```
# find /bin -size +10k -size -100k
```

```
# find /home -name "*.swp" -exec rm { } ₩;
```

```
find /etc -name "*ens33*"
```

-> ens33 설정파일 확인 가능

```
find /etc -size +100M
```

-> 100MB이상의 파일을 찾을 때 사용

■ grep 명령어

grep 명령어는 입력되는 파일에서 주어진 패턴 목록과 매칭되는 라인을 검색한 다음 표준 출력으로 검색된 라인을 복사해서 출력해 줌

형식: grep [옵션] pattern 파일명

<주요옵션>

-c : 해당하는 라인의 수만 표시함

```
#grep -c root /etc/passwd
```

-l : 옵션이 매칭된 파일명을 돌려 줌

```
#grep -l root /etc/passwd /etc/hosts /etc/services
```

-i : 대소문자 구분 안함

```
#grep -i Root /etc/passwd
```

-n : 행번호와 같이 출력

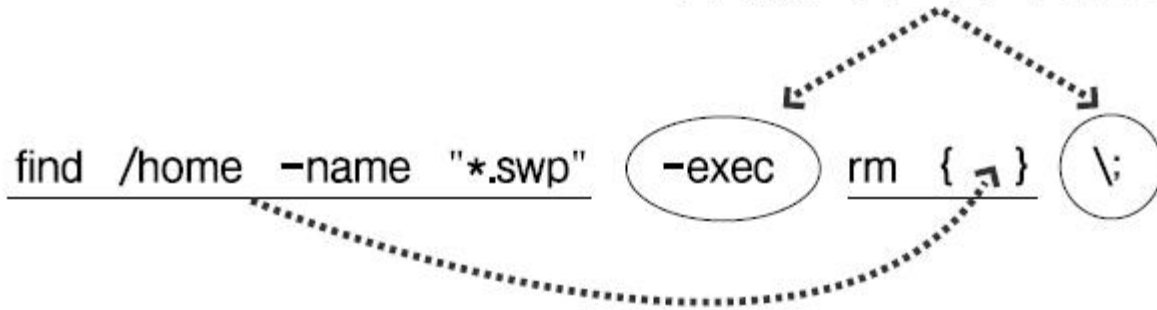
```
#grep -n root /etc/passwd
```

-v : 패턴이 포함되지 않는 라인을 출력

```
#grep -v nologin /etc/passwd
```

※ -r : 패턴의 내용을 가진 파일을 찾을 때

외부 명령어의 시작과 끝을 표시함



find의 결과인 "*.swp" 파일이 rm 명령으로 수행됨
즉, 파일이 삭제됨

■ 파이프, 필터, 리다이렉션

파이프

| (shift+W표시)

- 두개의 프로그램을 연결해 주는 연결통로의 의미

ls -l /etc | more

필터

- 필요한 것만 걸러주는 명령어

- grep tail wc sort awk sed등

ps -ef | grep [bash] [bash]글자가 들어간 것 출력

rpm -qa | grep [bind] 설치된 패키지중에서 [bind] 글자 들어간것 출력

리다이렉션

- 입출력의 방향을 바꿔 줌

표준 입력 0

표준 출력 1

표준 에러 2

- 표준 입력은 키보드 , 표준 출력은 모니터 이지만 이를 파일로 처리하고 싶을 경우

ls -l > aa.txt ls -l 결과물을 모니터가 아닌 aa.txt에 저장 또는 overwrite

ls -l >> aa.txt 기존에 파일이 있으면 기존의 내용에 append

sort < aa.txt aa.txt 파일을 읽어와서 정렬후 화면에 출력

sort < aa.txt > bb.txt aa.txt파일을 읽어와 정렬후 bb.txt 파일에 쓴다.

사용부호는 > >> < <<가 있음

1. 출력을 입력으로 : > 부호를 사용

> : 결과를 덮어쓰기

>> : 결과값 이어쓰기

2. 표준 출력인 1은 생략 가능

■ 프로그램 설치를 위한 RPM

RPM(Redhat Package Manager)

Windows의 "setup.exe"와 비슷하게 프로그램을 설치한 후에 바로 실행할 수 있는 설치 파일 확장명은 *.rpm이며, 이를 '패키지(Package)'라고 부름.

파일의 의미

```
[root@localhost linktest]# rpm -qa | grep net-tools  
net-tools-2.0-0.22.20131004git.el7.x86_64
```

패키지이름-버전-릴리즈번호.버전.아키텍처.rpm

- 패키지이름 : net-tools → 패키지(프로그램)의 이름
- 버전 : 2.0 → 대개 3자리수로 구성. 주버전, 부버전, 패치버전
- 릴리즈번호 : 0.22.20131004git → 문제점을 개선할 때마다 붙여지는 번호
- 버전 : el7 → 배포버전
- 아키텍처 : x86_64 → x86계열의 64bit CPU를 의미
i386~686 : 32bit CPU
- src : 소스
- noarch : 모든 CPU

rpm -qa <패키지 이름> → 패키지가 설치되었는지 확인

설치 : rpm -Uvh <패키지 파일(*.rpm)>

U → (대문자) 패키지가 설치/업그레이드

v → 설치과정의 확인

h → 설치진행과정을 "#"마크로 화면에 출력

패키지 파일이 현재 디렉토리에 존재해야함

삭제 : rpm -e <패키지 이름>

<패키지 파일 RPM의 단점>

패키지 파일 자체가 있어야 함

현재 내가 있는 디렉토리 있어야지 설치가 가능함

패키지 의존성이 있음 (A라는 패키지를 정상적으로 실행하려면 B라는 패키지가 있어야 함)

YUM(Yellowdog Updater Modified) 개념 = DNF(Dandified dnf)

- "rpm"명령의 패키지 의존성 문제를 완전하게 해결됨.

- 인터넷을 통하여 필요한 파일을 저장소(Repository)에서 자동으로 모두 다운로드해서 설치하는 방식 (저장소의 URL은 /etc/yum.repos.d/ 디렉터리)

* 인터넷이 되지 않으면 yum사용이 되지 않는 것이 단점

YUM 기본적인 사용법

- 기본 설치 : yum install <패키지 이름>

주로 "yum -y install <패키지 이름>"으로 사용

"-y"는 사용자의 확인을 모두 "yes"로 간주하고 설치를 진행한다는 옵션

- RPM 파일 설치 : yum localinstall rpm파일이름.rpm (인터넷으로 의존성 해결해줌)

dnf install rpm파일이름.rpm

- 업데이트 가능한 목록 보기 : yum check-update

- 업데이트 : yum update <패키지 이름>

- 삭제 : yum remove <패키지 이름>

- 정보 확인 : yum info <패키지 이름>

yum install -y wget

wget : url가지고 주소로 다운 받음

wget 주소입력 : 설치

※ 서버 : 서비스를 제공

어떤 서비스를 제공하려고 하면

1. 설치 2. 설정 3. 서비스 시작 4. 방화벽 허용

systemctl restart/status/enable : 시스템의 재시작/상태/자동으로 동작시키고 싶은 프로그램

파일질라 프로그램

파일질라 프로그램 접속후 접속해제

rpm -e로 vsftpd 삭제 → yum으로 vsftpd 설치 → vsftpd 설정 후 서비스 시작 → 파일질라로 접속

■ 파일과 디렉토리 속성

ls -l

d 디렉토리표시

- 일반파일 표시

l 링크파일 표시

r : 퍼미션

◆ 파일 유형

d : 디렉터리- : 일반적인 파일 l : 링크

◆ 파일 허가권(permission)

- "rw-." "r--" "r--" 3개씩 끊어서 읽음 (rwx r-x r-x)

- 첫 번째 "rw-" 는 소유자(User)의 접근 권한

- 두 번째 "r--" 는 그룹(Group)의 접근 권한

- 세 번째 "r-"는 그 외 사용자(Other)의 접근 권한

r : 읽기 w : 쓰기 x : 실행

- 숫자로도 표시 가능(퍼미션을 8진수로 표현) r:4 w:2 x:1

◆ 파일 소유권 : 파일이나 디렉토리의 소유자 및 소유그룹을 표시

<파일 허가권 변경>

◆ chmod

chmod [변경] [대상]

ex) rwxr-xr-x를 rw-r—r—로 변경할 때(파일이름 : a.txt)

chmod 644 a.txt

chmod -x a.txt

chmod u-x,g-x,o-x a.txt

chmod u=rw,g=r,o=r a.txt

- 파일이나 디렉토리의 퍼미션을 변경하는 명령어

- 디렉터리에 실행 권한이 있어야 그 디렉터리에 들어갈 수 있으며 파일 검색이 가능함

- 디렉터리에 쓰기 권한이 있어야 그 아래에 다른 디렉터리나 파일 생성이 가능함

ex> root계정으로 /test/test1.txt 생성 후 확인

◆ chown, chgrp

- 파일의 소유권을 바꾸는 명령어

#chown 새로운사용자명 파일 or 디렉터리 : 파일 또는 디렉터리의 소유자를 변경

#chgrp 변경할그룹(ID) 파일or 디렉터리 : 파일 또는 디렉터리의 소유그룹을 변경

#chown 사용자명.그룹명 파일or디렉터리 : 소유자와 그룹을 한번에 변경할 수 있다

- R 옵션 : 하위 디렉터리 및 파일까지 변경