CHAPTER 06

관리자 권한 실행

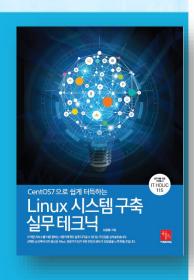
Section **01** | 관리자 계정 접속

Section **02** | 사용자 계정 관리

Section **03** | 그룹 관리

Section **04** | 소유자와 소유그룹 변경

Section **05** | 리눅스 시스템 종료





관리자 계정 접속

• 리눅스 시스템 관리자 계정 이해하기



- [1. 리눅스 시스템 관리자 계정에 대해 이해합니다. 2. 터미널 창에서 프롬프트를 변경하는 실습을 진행
 - 2, 터미널 창에서 프롬프트를 변경하는 실습을 진행합니다.
 - 3. 관리자 계정과 사용자 계정에 대한 정보를 확인합니다.

Section 01 관리자 계정 접속

- 관리자 계정은 사용자 계정생성과 권한 부여 등의 리눅스 시스템의 전반적인 관리를 수행
- 관리자 계정으로 접속하려면 먼저 터미널 창에 나타난 '\$' 프롬프트를 '#' 프롬프트로 변경한 다음 관리자인 root 계정으로 로그인

■ 프롬프트 변경 : su

- 터미널 창에 나타난 \$ 프롬프트는 일반 사용자 계정으로 접속된 상태를 의미
- 다른 계정으로 교체할 때는 substitute의 약어인 su 명령을 사용
- 일반 사용자 계정에서 root 또는 다른 사용자 계정으로 변경할 경우에는 암호를 물어보지만
- root에서 다른 사용자 계정으로 변경할 경우에는 모든 권한을 가지고 있기 때문에 암호를 묻지 않음

Section 01 관리자 계정 접속

● su 명령어 사용형식

\$ su

기능 다른 계정으로 변경

형식 su [옵션] 계정이름 (Enter-)

옵션 -c: 셸을 실행하지 않고 주어진 명령만 수행

-s: 지정된 셸로 접속

-, -1: 지정한 사용자의 환경변수를 적용하여 접속

Section 01

관리자 계정 접속

● 터미널 창에서 프롬프트 변경

| 예제 6-1 |

Step ○1 | 터미널 창에서 su 명령과 옵션 -을 함께 선언하여 su - root와 같이 명령을 수행하도록 하고 암호는 리눅스 시스템을 설치할 때 지정했던 spacezone을 입력합니다. 암호는 입력할 때 보안상의 이유로 입력한 암호가 화면에는 나타나지 않습니다.

\$ su - root 기능 root 계정으로 변경 형식 su [옵션] 계정이름 Enter니

```
root@localhost:~ - □ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[cskisa@localhost ~] $ su - root
암호:
마지막 로그인 실패: 뭘 12월 12 18:12:13 KST 2016 일시 pts/0
마지막 로그인 후 1 번의 로그인 시도가 실패하였습니다.
[root@localhost ~] # ■
```

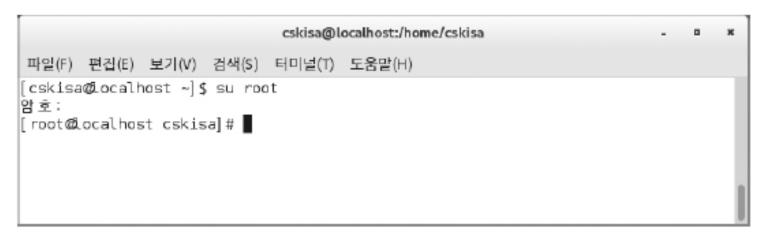
[그림 6-1] 일반 사용자 계정에서 관리자 계정인 root로 변경

관리자 계정 접속

위 그림에서는 프롬프트의 기호가 \$에서 #으로 변경된 것을 확인할 수 있습니다. 그리고 "마지막 로그인 1번의 로그인 시도가 실패하였습니다."는 메시지가 출력되었습니다. root 계정의 암호를 spacezone이 아닌 다른 암호를 입력했기 때문입니다. 독자 분들도 실습하는 과정에서 리눅스 시 스템을 설치했을 때 지정했던 root 계정의 암호를 기억하지 못하여 다른 암호를 입력하게 되면 접 속했던 실패 메시지가 [그림 6-1]과 같이 출력됩니다.

su - root 명령을 사용할 때 su 명령과 옵션 - 그리고 root 사이에는 반드시 공백이 들어가야합니다. su 명령을 사용할 때 [그림 6-2]와 같이 su root 명령을 - 옵션을 생략한 채 수행해도 됩니다. su 명령을 사용하면서 - 옵션을 사용하면 root 셸 환경변수를 가져와서 사용할 수 있지만 - 옵션을 사용하지 않으면 root의 셸 환경변수를 가져오는 것이 아니라 원래 사용자 계정의 셸 환경변수를 사용하게 됩니다. 그렇기 때문에 온전하게 관리자 계정에 설정된 환경변수를 사용하기 위해서는 반드시 su - root와 같이 옵션을 함께 사용해 주어야합니다.

Section 01 관리자 계정 접속



[그림 6-2] root의 셸 환경변수를 사용하지 않고 root로 변경

Section 01

관리자 계정 접속

• Step 02 | 터미널 창에서 프롬프트가 #으로 나타난 상태는 관리자 계정인 root로 접속되었다는 것을 의미합니다. 그렇다면 root 계정에서 일반 사용자 계정인 cskisa로 변경하고자 한다면 다음과 같이 명령을 수행하면 됩니다.

```
# su - cskisa
기능 root 계정으로 변경
형식 su [옵션] 계정이름 Enter
```



[그림 6-3] root 계정에서 사용자 계정인 cskisa로 변경

Step 03 | 다음 실습을 위해 터미널 창에서 root 계정으로 변경합니다.

```
root@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[cskisa@Localhost ~] $ su - root
암호:
마지막 로그인: 뭘 12월 12 18:28:56 KST 2016 일시 pts/0
[root@localhost ~]#
```

[그림 6-4] root의 셸 환경변수를 적용하여 root 계정으로 변경

|계정정보 확인

- 계정에 대한 일반정보는 /etc/passwd 파일에 저장되어 있으며
- ▶ 유닉스를 사용했던 초기시절에 비해 해킹에 대한 위험이 최근에는 급증함에 따라
- 사용자 계정 암호를 /etc/shadow 파일에도 별도로 저장하여 관리

Section 01

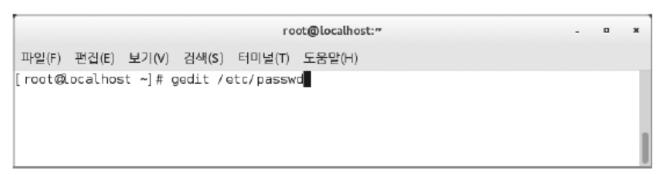
관리자 계정 접속

● 계정정보 확인

| 예제 6-2 |

• Step [] | 터미널 창의 프롬프트가 #으로 표기되어 있는지를 확인한 다음 gedit 또는 vi 에디터 를 사용하여 /etc/passwd 파일을 엽니다.

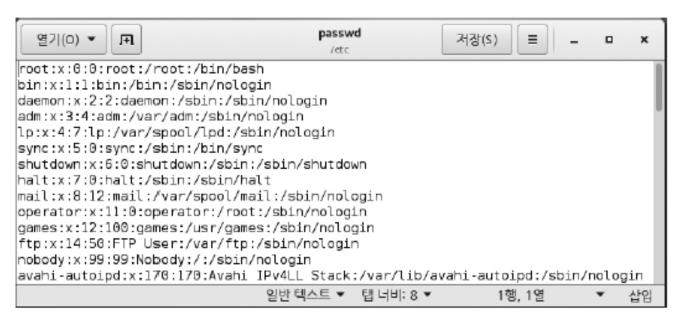
gedit /ect/passwd 기능 터미널 창에서 지에디트를 실행하여 /etc/passwd 파일 열기 형식 gedit /경로명/파일이름 Enter니



[그림 6-5] gedit 명령으로 /etc/passwd 파일 열기

Section 01) 관리자 계정 접속

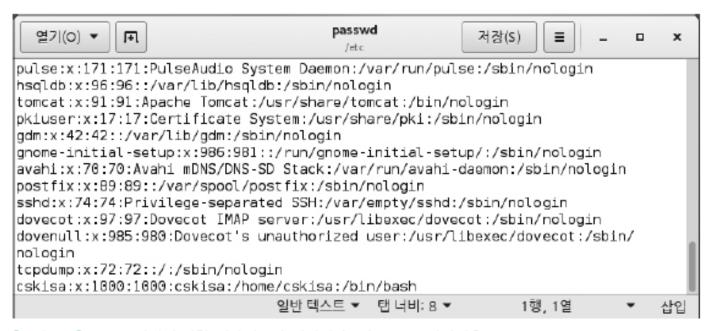
• Step 02 | gedit 창이 활성화되면서 /etc/passwd 파일에 저장되어 있는 root 계정의 일반정보가 자세히 나타나면 맨 위에 있는 첫 줄의 내용을 확인합니다.



[그림 6-6] root 계정에 대한 일반정보가 저장된 /etc/passwd 파일내용

Section 01) 관리자 계정 접속

• Step 03 | gedit 창이 활성화되면서 /etc/passwd 파일의 제일 하단에는 사용자 계정인 cskisa 에 대해 구성된 정보를 확인할 수 있습니다.



[그림 6-7] cskisa 계정에 대한 일반정보가 저장된 /etc/passwd 파일내용

Section 01

관리자 계정 접속

/etc/passwd 파일에는 root 관리자 계정에 대한 정보와 cskisa 사용자 계정에 대한 정보가 저장 되어 있습니다. 두 개의 계정 중에서 cskisa 계정에 대한 /etc/passwd 파일 구성내용을 각 기본 항목별로 자세히 살펴보도록 하겠습니다.

① 로그인 ID

로그인 ID는 사용자 계정의 이름을 의미합니다. 로그인 ID는 최대 32자까지 지정할 수 있지만 너무 긴 ID는 효율성이 떨어지기 때문에 다른 운영체제와의 연동을 위해 적당한 길이(8자 정도)로 지정하는 것을 권장합니다. 로그인 ID에는 콜론(:)과 [Enter_]를 제외한다면 어떤 문자라도 상관 없이 사용할 수 있습니다. 그렇지만 동일한 ID를 사용할 수는 없습니다.

관리자 계정 접속

② 암호저장 항목

x는 유닉스 초기시절 사용자 암호를 저장하기 위한 항목입니다. 이 항목은 이전 프로그램과의 호 환성을 위해 기본 항목을 그대로 유지하고 있으며 x로 표시합니다.

UID

리눅스 시스템이 사용자를 구분하기 위해 사용되는 사용자 ID 번호입니다. 일반사용자는 1000 번부터 할당되며 사용자에 대한 시스템 관리업무를 위해 내부적으로 사용자를 위해 예약되어 있 기 때문에 임의로 변경하지 않는 것이 좋습니다. 사용자를 구별함에 있어 로그인 ID가 다르더라도 UID가 같으면 리눅스 시스템은 동일한 사용자로 판단합니다. 따라서 시스템 관리자는 UID가 중 복되지 않도록 관리에 주의를 기울여야 합니다.

4 GID

사용자가 속해 있는 그룹의 ID를 나타냅니다. 리눅스 시스템에서 사용되는 모든 사용자는 무조건 하나 이상의 그룹에 속해 있어야 합니다. GID는 사용자를 등록할 때 정해지며 GID를 지정하지 않으면 자동으로 로그인 ID가 그룹으로 등록됩니다. 등록된 GID는 /etc/group 파일에서 확인할 수 있습니다.

관리자 계정 접속

⑤ 설명

사용자에 대한 일반적인 정보(이름, 부서, 연락처 등)가 기록되는 항목입니다. 이 항목은 유닉스에서 유래된 기원을 바탕으로 사용자 개인정보GECOS라고도 불립니다.

⑥ 홈 디렉터리

이 항목에서는 사용자 계정에 할당된 홈 디렉터리를 절대경로의 형태로 저장됩니다. 홈 디렉터리 는 사용자 계정을 생성하면서 자동으로 부여되며 홈 디렉터리 아래에는 서브 디렉터리와 파일 등 을 생성하거나 저장할 수 있습니다.

🕜 로그인 셸

사용자의 로그인 기본 셸을 지정해 줍니다. 현재 CentOS 리눅스에서는 배시 셸을 기본 셸로 적용 하여 사용자에 대한 로그인 셸을 지정합니다.

root 계정 접속

- su 명령을 사용하지 않고 리눅스 시스템을 재부팅하여 처음부터 root 계정으로 접속할 수 있음
- 리눅스 시스템을 재부팅하면 등록된 사용자 계정목록이 나타남
- 사용자 계정 맨 아래 '목록에 없습니까?'를 클릭하여 root 계정 접속

Section 01 관리자 계정 접속

- 리눅스 시스템 재부팅 또는 부팅 후 root 계정 접속
 - Step 01 | 아래 그림에서 '목록에 없습니까?'를 클릭합니다.



[그림 6-8] root 계정으로 접속하기 위한 절차

• Step 02 | 사용자 이름에 root를 입력 후 〈다음〉 버튼을 클릭합니다.



[그림 6-9] 사용자 이름에 root 입력

관리자 계정 접속

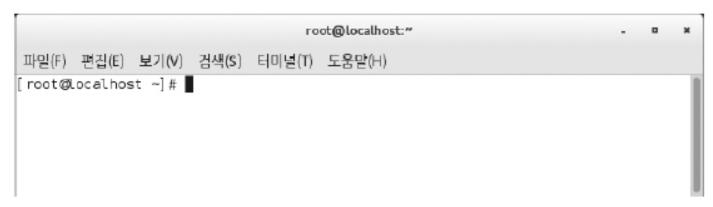
• Step 03 | 관리자 계정인 root 계정의 암호는 리눅스 시스템을 설치할 때 설정한 암호 spacezone을 입력 후 〈로그인〉 버튼을 클릭합니다.



[그림 6-10] root 계정의 암호 spacezone 입력

관리자 계정 접속

Step 04 | root 계정으로 접속된 상태에서 터미널 창을 열어보면 프롬프트가 '#으로 나타납니다. 그리고 터미널 창의 상단에 보면 root@localhost:~와 같이 root 계정으로 접속된 상태임을 알 수 있습니다.



[그림 6-11] root 계정으로 접속된 상태에서의 # 프롬프트



사용자 계정 관리

• 사용자 계정 발급과 관리방법 이해하기



- 1. 사용자 계정을 생성하는 과정에 대해서 살펴봅니다. 2. 사용자 계정정보 수정방법에 대해 살펴봅니다.

 - 3. 사용자 계정과 삭제에 대해서 실습합니다.

▋ 사용자 계정생성

- 사용자 계정은 여러 명의 사용자가 리눅스 시스템에 접속하여 사용할 때 사용자를 구별하는 역할을 수행
- ▶ 사용자 계정은 사용자가 시스템에 접근할 수 있는 유일한 방법이며
- ▶ 시스템 관리자 입장에서도 사용자의 접근권한을 통제할 수 있는 중요한 역할을 수행
- ▶ 그렇기 때문에 사용자 계정 관리는 리눅스 시스템 관리자의 중요한 업무 중 하나에 속함

● 사용자 계정 생성 명령 : adduser 보다 useradd 명령을 사용

\$ useradd

기능 사용자 계정을 새로 생성

형식 useradd [옵션] [생성할 계정이름] Enter↓]

옵션 -c: 사용자 이름 등 부가적인 설명 지정

-d: 사용자 계정의 홈 디렉터리 지정

-D: 기본설정 값을 출력하거나 설정

-e: 유효기간 설정 (YYYY-MM-DD)

-f: 사용자 계정의 유효일자 (-f-175: 앞으로 175일간 사용가능)

-g: 사용자 계정의 로그인 기본그룹 GID 지정

-G: 2차 그룹의 GID 지정

-o: 사용자 계정의 UID 중복 허용

-p: 사용자 계정의 패스워드

-s: 사용자 계정의 기본 셸 지정

-u: 사용자 계정의 UID 지정

Section 02 사용자 계정 관리

● 터미널 창에서 프롬프트가 #으로 표기되어 있는지를 확인한 다음 옵션 없이 user_001 사용자 계정 생성

| 예제 6-3 |

• Step [] | home 디렉터리에 존재하는 사용자 계정 리스트를 출력합니다.

ls /home

기능 home 디렉터리에 존재하는 사용자 계정 리스트 출력

형식 1s [/디렉터리명] [Enter↓]

Section 02 가용자 계정 관리

Step 02 | 사용자 계정 user_001 계정을 새로 생성한 다음 추가된 계정이 존재하는지를 확인합니다.

```
# useradd user_001
기능 user_001 사용자 계정을 새로 생성
형식 useradd [새로 생성할 계정이름] Enter_)
```

```
root@localhost:~ 고 보
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~] # ls /home
cskisa
[root@localhost ~] # useradd user_001
[root@localhost ~] # ls /home
cskisa user_001
[root@localhost ~] # ■
```

[그림 6-12] 새로운 사용자 계정 user_001 생성

Section 02 사용자 계정 관리

● 생성된 새 계정에 암호 지정 : passwd 명령 사용형식

passwd

기능 사용자 계정의 암호 수정

형식 passwd [옵션] 계정이름 Enter니

옵션 -1: 지정한 계정의 암호 잠금

-u : 암호의 잠금 해제

-d: 지정한 계정의 암호 삭제

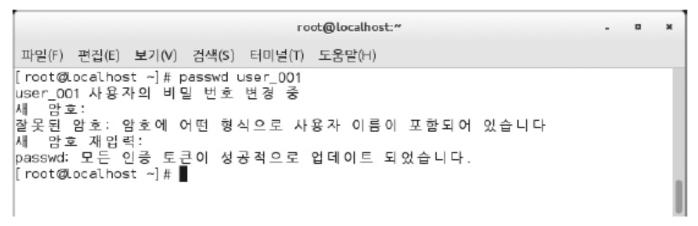
Section 02 사용자 계정 관리

● 사용자 계정 user_001 계정의 암호를 계정이름과 동일하게 설정

| 예제 6-4 |

사용자 계정인 user_001 계정의 암호를 계정이름과 동일하게 user_001로 설정합니다.

passwd user_001 기능 user_001 계정에 암호 설정 형식 passwd [사용자 계정이름] Enter-



[그림 6-13] user_001 사용자 계정에 암호 설정

Section 02 사용자 계정 관리

● user_001 계정정보 확인:/etc/passwd

| 예제 6-5 |

사용자 계정인 user_001 계정에 대한 자세한 정보를 gedit 창에서 확인합니다.

gedit /etc/passwd 기능 계정에 대한 정보확인 형식 gedit [/경로명/파일명] Enter



[그림 6-14] user_001 계정에 대한 정보 출력

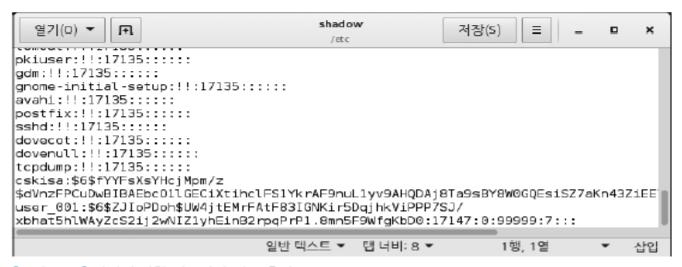
Section 02 〉사용자 계정 관리

● 암호화된 user_001 계정정보 확인:/etc/shadow

| 예제 6-6 |

리눅스 시스템을 보호하기 위해 별도로 암호가 저장되어 있는 /etc/shadow 파일에 대한 정보를 확인합니다.

gedit /etc/shadow 기능 계정의 암호화된 정보 확인 형식 gedit [/경로명/파일명] Enter-J



[그림 6-15] 계정에 대한 암호화된 정보 출력

Section 02 사용자 계정 관리

● 홈 디렉터리 지정하여 계정생성

| 예제 6-7 |

Step 01 | 새로 생성할 사용자 계정 user_002의 홈 디렉터리를 work_06으로 지정하면 별도의 디렉터리 생성과정 없이 자동으로 홈 디렉터리가 work_06으로 지정됩니다. 암호는 사용자계정이름과 동일하게 user_002로 입력합니다.

useradd -d /work_06 user_002

기능 새 계정을 생성하면서 홈 디렉터리를 work_02으로 지정

형식 useradd [옵션] [/홈 디렉터리 경로명] [계정이름] [Enter』

passwd user 002

기능 user_002 계정에 암호 설정

형식 passwd [계정이름] Enter니

Section 02 사용자 계정 관리

```
root@localhost:~ - ■ ★
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

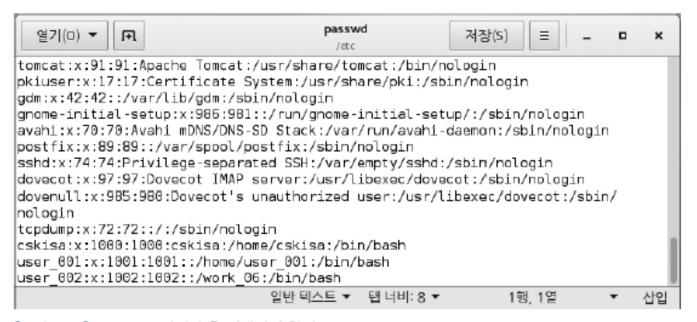
[root@localhost ~] # useradd ~d /work_06 user_002
[root@localhost ~] # passwd user_002
user_002 사용자의 비밀 번호 변경 중
새 암호:
잘못된 암호: 암호에 어떤 형식으로 사용자 이름이 포함되어 있습니다
새 암호 재입력:
passwd: 모든 인증 토큰이 성공적으로 업데이트 되었습니다.
[root@localhost ~] # ■
```

[그림 6-16] 홈 디렉터리를 지정하여 user_002 계정 생성

Section 02 〉 사용자 계정 관리

 Step 02 | gedit 창에서 /etc/passwd 파일을 열어 새로 생성한 user_002 계정에 지정된 홈 디렉터리가 work_06으로 적용되었는지 정보를 확인합니다.

gedit /etc/passwd 기능 계정에 대한 정보확인 형식 gedit [/경로명/파일명] Enter니



[그림 6-17] user_002 계정의 홈 디렉터리 확인

Section 02 사용자 계정 관리

● 리눅스 파일내용 출력: tail 명령

\$ tail

기능 파일내용을 마지막 라인부터 10번째 라인까지만 내용출력

형식 tail [옵션] [생성할 계정이름] [Enter-]

옵션 -f: 파일의 마지막 10라인을 실시간으로 계속해서 내용출력

-F: 파일 변동 시 다시 명령을 실행할 필요 없이 실시간으로 내용출력

-n: 지정한 n라인만큼 내용출력

-n +n : 마지막 라인이 아니라 첫 번째 라인부터 시작해 n번째 라인이후 출력

--byte=n : 지정한 n바이트 만큼의 내용을 출력

Section 02 〉사용자 계정 관리

● user_002 계정의 정보를 tail 명령으로 출력

| 예제 6-8 |

Step 01 | tail 명령과 함께 /etc/passwd 파일을 지정하여 파일내용을 출력합니다.

tail /etc/passwd

기능 파일내용을 마지막 라인부터 10번째 라인까지만 내용출력

형식 tail [옵션] [파일명] Enter니

```
root@localhost:" - 미 *
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[root@localhost ~] # tail /etc/passwd
gnome-initial-setup: x: 986: 981::/run/gnome-initial-setup/:/sbin/nologin
avahi: x: 70: 70: Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
postfix: x: 89: 89::/var/spool/postfix:/sbin/nologin
sshd: x: 74: 74: Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
dovecot: x: 97: 97: Dovecot IMAP server:/usr/libexec/dovecot:/sbin/nologin
dovenull: x: 985: 980: Dovecot's unauthorized user:/usr/libexec/dovecot:/sbin/nologin
tcpdump: x: 72: 72::/:/sbin/nologin
```

(a) tail 명령 실행

Section 02 〉 사용자 계정 관리

```
root@localhost:~ . ㅁ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
sshd: x: 74: 74: Privilege- separated SSH: / var/empty/sshd: / sbin/nologin
dovecot: x: 97: 97: Dovecot IMAP server: / usr/libexec/dovecot: / sbin/nologin
dovenull: x: 985: 980: Dovecot's unauthorized user: / usr/libexec/dovecot: / sbin/nologin
tcpdump: x: 72: 72: */: / sbin/nologin
cskisa: x: 1000: 1000: cskisa: / home/cskisa: / bin/bash
user_001: x: 1001: 1001: */ home/ user_001: / bin/ bash
user_002: x: 1002: 1002: */ work_06: / bin/ bash
[root@localhost ~] #
```

(b) tail 명령으로 /etc/passwd 파일내용 출력

[그림 6-19] tail 명령 실행 및 /etc/passwd 파일내용 출력

Section 02 〉사용자 계정 관리

Step 02 | tail 명령은 기본적으로 마지막 라인부터 10번째 라인까지의 파일내용을 출력하기 때문에 user_002 계정에 대한 정보가 10라인 출력되었습니다. 이번에는 옵션 -n을 사용하여 필요한 라인만큼만 /etc/passwd 파일의 내용을 출력하도록 하겠습니다. 여기서는 마지막 라인부터 3개 라인만 출력되도록 수행합니다.

tail -3 /etc/passwd

기능 파일내용을 마지막 라인부터 3번째 라인까지만 내용출력

형식 tail [옵션] [파일명] [Enter]

```
root@localhost:~ - ㅁ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~] # tail -3 /etc/passwd
cskisa; x: 1000: 1000: cskisa: /home/cskisa: /bin/bash
user_001: x: 1001: 1001: :/home/user_001: /bin/bash
user_002: x: 1002: 1002: :/work_06: /bin/bash
[root@localhost ~] # ■
```

[그림 6-19] /etc/passwd 파일내용을 마지막 3라인만 출력

실습 6-1 다음 항목에서 주어진 지시사항을 수행하시오.

- 1. 사용자 계정의 홈 디렉터리 /work_06을 지정하여 master 계정 생성하기
- 2, master 계정의 암호를 123456으로 설정하기
- 3, master 계정의 UD 확인하기
- 4. master 계정의 GID 확인하기
- 5. master 계정의 홈 디렉터리 확인하기

```
# useradd –d /work_06 master
# pass master
암호: 123456
암호 재입력: 123456
# gedit /etc/passwd 또는 tail /etc/passwd
UID 확인
GID 확인
홈 디렉터리 확인
```

■ 사용자 계정정보 수정

● 이미 생성된 사용자 계정에 대한 정보 수정 : usermod 명령 사용

\$ usermod

기능 사용자 계정정보 수정

형식 usermod [옵션] [사용자 계정이름] Enter니

옵션 -c: 사용자 이름 등 부가적인 설명 수정

-d: 사용자 계정의 홈 디렉터리 수정

-e: 사용자 계정 만료일(유효기간) 수정 (YYYY-MM-DD)

-f: 사용자 계정 비활성화 일자 수정

-a: 사용자 계정의 로그인 기본그룹 GID 수정

-G: 2차 그룹의 GID 수정

-1: 사용자 계정 이름 변경

-o: 사용자 계정의 UID 중복 허용

-s: 사용자 계정의 기본 셸 수정

-u: 사용자 계정의 UID 수정

Section 02 가용자 계정 관리

● user_002 계정의 UID 변경: -u 옵션

| 예제 6-9 |

● Step ○1 | 사용자 계정 user_002에 대한 UID를 2001로 변경합니다.

usermod -u 2001 user_002
기능 user_002 계정의 UID를 2001로 변경
형식 usermod [옵션] [사용자 계정이름] Enter.

Section 02 〉 사용자 계정 관리

Step 02 | etc 디렉터리에 존재하는 passwd 파일의 내용 중에서 마지막 라인부터 2번째 라인
 까지만 내용을 출력합니다.

```
# tail -2 /etc/passwd
기능 파일내용을 마지막 라인부터 2번째 라인까지만 내용출력
형식 tail [옵션] [파일명] Enter-
```



[그림 6-20] user_002 계정의 UID를 2001로 변경

Section 02 사용자 계정 관리

● 홈 디렉터리와 로그인 ID 한꺼번에 변경: -I, -u 옵션 사용

| 예제 6-10 |

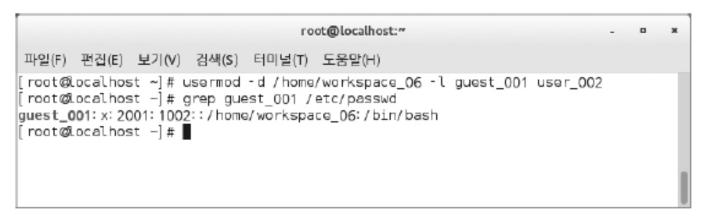
 Step ○1 | 사용자 계정 user_002에 대한 로그인 ID를 guest_001로 지정하고 홈 디렉터리를 workspace_06 디렉터리로 지정합니다.

```
# usermod -d /home/workspace_06 -l guest_001 user_002
기능 user_002 계정의 홈 디렉터리와 로그인 ID 변경
형식 usermod [옵션] [홈 디렉터리] [옵션] [새 로그인 ID] [기존 로그인 ID] Enter-
```

Section 02 사용자 계정 관리

• Step 02 | 사용자 로그인 ID guest_001에 대한 문자열의 정보를 출력합니다.

```
# grep guest_001
기능 지정한 guest_001 문자열을 찾아서 정보를 출력
형식 grep [옵션] [찾을 문자열] Enter니
```



[그림 6-21] user_002 사용자 계정의 홈 디렉터리와 로그인 ID 변경

지금까지 배운 내용을 실습을 통해 실력을 업그레이드 해 봅시다. 이 실습을 진행하기 위해서는 [실습 6-1]을 먼저 수행해야 합니다.

실습 6-2 다음 항목에서 주어진 지시사항을 수행하시오.

- 1. [실습 6-1]에서 생성한 master 계정의 UD를 2003으로 변경하기
- 2 tail 명령으로 변경된 master 계정의 UD 확인하기
- 3. master 계정의 홈 디렉터리를 workspace 06 디렉터리로 지정하기
- 4. grep 명령으로 변경된 master 계정의 홈 디렉터리 확인하기
- 5. master 계정의 로그인 ID를 manager로 변경하기
- 6. grep 명령으로 변경된 master 계정의 로그인 ID 확인하기

```
# usermod –u 2003 master
# tail -2 /ect/passwd
# usermod –d /home/workspace_06 master
# grep master
# usermod —l manager master
# grep manager
```

■ 사용자 암호 기한설정 및 변경 : chage 명령 사용

● 사용자 계정의 암호는 주기적으로 변경해 주는 것을 권장

chage

기능 시스템 보안을 위해 사용자 계정에 따른 암호 기한설정 및 변경

형식 chage [옵션] [사용자 계정이름] Enter니

옵션 -d: 암호를 마지막으로 변경한 일자 → 마지막 날짜

-E: 계정사용 종료일자 (YYYY-MM-DD) → 만료일자

-I : 암호를 유효기간 종료 이후에 계정 비활성화 될 일 수 → 장금 날짜

-m : 암호를 변경할 수 있는 최소 일 수 (0 이면 아무 때나 변경) → 최소 날짜

-M: 마지막 변경 이후의 암호 유효 일 수 → 최대 날짜

-₩: 암호 만료기간 몇 일전에 안내 메시지 출력 → 경고 날짜

-1 : 현재 사용자의 암호 만기에 대한 기본정보 출력 (L의 소문자)

-1 : 암호 만료일을 지정하지 않거나 설정 값 해제 (숫자 1임)

Section 02 사용자 계정 관리

● user_001 계정에 설정된 암호에 대한 정보 출력과 기간설정

| 예제 6-11 |

Step 01 | 사용자 계정 user_001에 설정되어 있는 암호정보를 출력합니다.

```
# chage -l user_001

기능 user_001 계정에 설정되어 있는 기본정보 출력
형식 chage [옵션] [사용자 계정이름] Enter-1
```

```
root@localhost:"
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@Localhost ~]# chage -l user 001
마지막으로 암호를 바꾼 날
                                                    : 12월 12, 2016
암호 만료
                                       : 안함
암호가 비활성화 기간
                                             : 안함
계정 만료
                                             : 아하
암호를 바꿀 수 있는 최소 날 수
                                : 0
암호를 바꿔야 하는 최대 날 수
                                : 99999
암호 만료 예고를 하는 날 수
                                : 7
[root@localhost ~]#
```

[그림 6-22] 사용자 계정 user_001에 설정된 암호정보 출력

Section 02 〉 사용자 계정 관리

Step 02 | 사용자 계정 user_001에 대한 계정사용 만료 일자를 2017년 01월 31일까지로 설정하고 변경된 정보를 확인합니다(연월일 구분은 - 또는 /를 사용).

```
# chage -E 2017/01/31 user_001
기능 user_001 계정의 만료 일자를 2017년 01월 31일로 설정
형식 chage [옵션] [사용자 계정이름] Enter-
```

```
root@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@Localhost -] # chage -E 2017/01/31 user_001
[root@Localhost ~]# chage -l user 001
마지막으로 암호를 바꾼 날
                                                     : 12월 12, 2016
암호 만료
                                        : 안 함
암호가 비활성화 기간
                                              : 안함
계정 만료
                                              : 1월 31, 2017
맘호를 바꿀 수 있는 최소 날 수
                                 : 0
암호를 바꿔야 하는 최대 날 수
                                : 99999
암호 만료 예고를 하는 날 수
                                 : 7
[root@Localhost -]#
```

[그림 6-23] 사용자 계정 user_001의 만료일자 설정

Step 03 | 사용자 계정 user_001에 대해 설정된 암호를 최대 20일까지 사용할 수 있도록 설정한 다음 변경된 내용을 확인합니다.

```
# chage -M 20 user_001

기능 user_001 계정의 암호를 바꿔야 하는 최대 일 수를 20일로 설정
형식 chage [옵션] [사용자 계정이름] Enter-
```

```
# chage -1 user_001

기능 user_001 계정에 설정되어 있는 기본정보 출력
형식 chage [옵션] [사용자 계정이름] Enter-
```

```
root@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@Localhost -]# chage -M 20 user 001
root@localhost ~] # chage -l user 001
마지막으로 암호를 바꾼 날
                                                     : 1월 15, 2017
암호 만료
                                       : 2월 04, 2017
암호가 비활성화 기간
                                              : 2월 09, 2017
                                             : 1월 31, 2017
계정 만료
암호를 바꿀 수 있는 최소 날 수
                               : 0
암호를 바꿔야 하는 최대 날 수
                                : 20
암호 만료 예고를 하는 날 수
                                : 7
[root@Localhost -]#
```

[그림 6-24] 사용자 계정 user_001의 암호를 바꿔야 하는 최대 유효일수 설정

지금까지 배운 내용을 실습을 통해 실력을 업그레이드 해 봅시다. 이 실습을 진행하기 위해서는 [실습 6-1]과 [실습 6-2]를 먼저 수행해야 합니다.

심습 6-3 다음 항목에서 주어진 지시사항을 수행하시오.

- 1. [실습 6-1]에서 생성한 master 계정에 설정되어 있는 암호정보 확인하기
- 2. master 계정사용 만료 일자를 2018년 05월 05일로 변경하기
- 3. 변경된 master 계정의 암호정보 확인하기
- 4. master 계정의 설정된 암호를 최대 30일까지 사용할 수 있도록 변경하기
- 변경된 master 계정의 암호정보 확인하기

```
# chage -1 master
# chage -E 2018/05/05 master
# chage -1 master
# chage –M 30 master
# chage -1 master
```

■ 사용자 계정 삭제 : userdel 명령 사용

● 사용자 계정을 삭제할 때는 관련된 홈 디렉터리와 파일을 모두 삭제

\$ userdel

기능 사용자 계정 삭제

형식 userdel [옵션] 계정이름 Enter니

옵션 -r: 홈 디렉터리 삭제

-f: 사용 중인 계정이름도 강제 삭제

-rf : 홈 디렉터리이외에 존재하는 파일 삭제

Section 02 〉 사용자 계정 관리

- 사용자 계정 삭제 수행방법
 - /etc/passwd 파일에서 사용자 계정 삭제:# userdel -r 계정이름
 - ② /etc/group 파일에서 사용자 그룹 삭제:# groupdel 계정이름
 - ③ /home 디렉터리에 있는 사용자의 홈 디렉터리 삭제: # rm -rf /home/계정이름
 - ④ /var/spool/mail에 있는 사용자 메일 삭제: # rm /var/spool/mail/계정이름
 - ⑤ 데이터베이스 연동에서 삭제하려는 계정이름과 암호 삭제
 - ⑥ 기타 응용 프로그램에 존재하는 사용자 계정의 정보 삭제
 - 🕜 리눅스 시스템을 재시작

Section 02 사용자 계정 관리

● UID를 이용하여 사용자 계정관련 모든 파일 삭제방법

```
# find -user UID -exec rm -r { } \;
기능 파일내용 중에서 UID를 찾아서 관련된 파일을 모두 삭제
형식 find [옵션] UID [옵션] [명령] {} \; Enter니
```

Section 02 사용자 계정 관리

● user_0001 사용자 계정 삭제

| 예제 6-12 |

Step ○1 | 홈 디렉터리에 존재하는 사용자 계정이름에 대해 먼저 확인합니다.

1s /home 기능 홈 디렉터리에 존재하는 사용자 계정이름 확인 형식 1s [옵션] [디렉터리명] Enter-

```
root@localhost:~ - □ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[root@localhost ~] # ls /home
cskisa user_001
[root@localhost ~] # ■
```

[그림 6-25] 홈 디렉터리에 존재하는 사용자 계정 목록

Section 02 〉 사용자 계정 관리

• Step 02 | 삭제하고자 하는 user_001 계정에 대한 UID 정보 등을 확인합니다.

```
# grep user_001 /etc/passwd
기능 /etc/passwd 파일에서 특정 문자열 user_001을 찾음
형식 grep [찾을 문자열] [대상 파일명] Enter-
```



[그림 6-26] 문자열 검색을 통한 user_001 계정의 기본정보 출력

Section 02 가용자 계정 관리

Step □3 | 사용자 계정 user_001을 userdel 명령과 옵션 -r을 함께 사용하여 계정을 삭제한다음 ls /home 명령으로 user_001 계정이 삭제되었는지를 확인합니다.

```
# userdel -r user_001
기능 사용자 계정인 user_001 계정을 삭제
형식 userdel [옵션] [계정이름] Enter-
```

```
root@localhost:~ - ㅁ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[root@localhost ~] # userdel - r user_001

[root@localhost -] # ls /home
cskisa

[root@localhost -] # ■
```

[그림 6-27] user_001 계정 삭제 후 홈 디렉터리 확인

Step ①4 | 삭제한 사용자 계정 user_001의 UID는 [그림 6-26]에서 확인한 바와 같이 1001이 므로 find 명령과 함께 옵션을 사용하여 user_001 계정과 관련된 모든 파일을 한꺼번에 삭제한 다음 grep 명령으로 파일이 남아있는지를 확인합니다.

```
# find -user 1001 -exec rm -r { } \;
기능 파일내용 중에서 UID를 찾아서 관련된 파일을 모두 삭제
형식 find [옵션] UID [옵션] [명령] {} \\; Enter-\]
```

```
root@localhost:~ - 미 *
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[root@localhost ~] # find -user 1001 -exec rm -r {} \\
[root@localhost ~] # grep user_001 /etc/passwd

[root@localhost ~] # [
```

[그림 6-28] 삭제된 user_001 계정관련 모든 파일 한꺼번에 삭제 후 확인



그룹 관리

• 그룹 생성과 그룹 정보 이해하기



- [**빨**] 1 그룹을 생성하는 방법에 대해 실습합니다. | 2 그룹의 정보를 수정하는 방법에 대해 살펴
 - 2, 그룹의 정보를 수정하는 방법에 대해 살펴봅니다.
 - 3. 그룹 암호와 그룹 삭제 방법에 대해 실습합니다.

▮ 새 그룹 생성

● 새 그룹은 /etc/group 파일에 저장되며 사용형식은 다음과 같음

\$ groupadd

기능 새 그룹 생성

형식 groupadd [옵션] 그룹이름 Enter니

옵션 -g gid: 그룹의 GID를 직접 지정

-o: GID의 중복사용 허용

Section 03 기료 관리

● 옵션 없이 새 그룹 생성

```
| 예제 6-13 |
```

Step 01 | 새 그룹 db_zone을 옵션 없이 groupadd 명령으로 생성합니다.

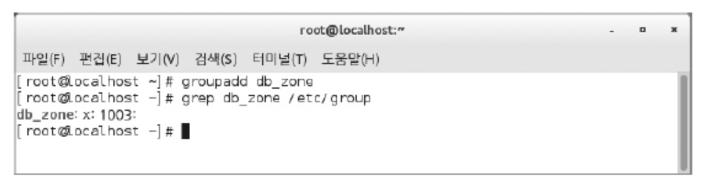
```
# groupadd db_zone
기능 새 그룹 db_zone을 생성
형식 groupadd [옵션] 그룹이름 EnterJ
```

```
root@localhost:~ - □ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~]# groupadd db_zone
[root@localhost -]# ■
```

[그림 6-29] 새 그룹 db_zone 생성

Step 02 | 새로 생성된 db_zone 그룹에 리눅스 시스템에서 자동으로 부여된 GID를 확인합니다.

```
# grep db_zone /etc/group
기능 /etc/group 파일에서 특정 문자열 db_zone을 찾음
형식 grep [찾을 문자열] [대상 파일명] Enter니
```



[그림 6-30] db_zone 그룹에 자동 설정된 GID 확인

● GID를 지정하여 새 그룹 생성

| 예제 6-14 |

GID를 2017로 설정하여 새 그룹 nt_zone을 생성한 다음 GID를 확인합니다.

```
# groupadd -g 2017 nt_zone
기능 GID를 2017로 설정하여 새 그룹 nt_zone을 생성
형식 groupadd [옵션] GID 그룹이름 EnterJ
```

```
root@localhost:~ - 교 *
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~] # groupadd - g 2017 nt_zone
[root@localhost -] # grep nt_zone / etc/group
nt_zone: x: 2017:
[root@localhost -] # ■
```

[그림 6-31] GID를 2017로 설정하여 새 그룹 nt_zone을 생성

● 기존 GID를 중복 지정하여 새 그룹 생성

| 예제 6-15 |

기존 GID 2017을 중복 지정하여 새 그룹 gp_zone을 생성한 다음 GID를 확인합니다.

```
# groupadd -g 2017 -o gp_zone
기능 기존 GID 2017을 중복 지정하여 새 그룹 gp_zone을 생성
형식 groupadd [옵션] GID [옵션] 그룹이름 Enter니
```

```
root@localhost:~ - ㅁ x
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~] # groupadd - g 2017 - o gp_zone
[root@localhost -] # grep gp_zone / etc/group
gp_zone: x: 2017:
[root@localhost -] # ■
```

[그림 6-32] 기존 GID 2017을 중복 지정하여 새 그룹 gp_zone을 생성

실습 6-4 다음 항목에서 주어진 지시시항을 수행하시오.

- 1, professor 그룹을 옵션 없이 새로 생성하기
- 2, 새로 생성한 professor 그룹에 자동으로 설정된 GID 확인하기
- 3, GID를 2018로 설정하여 student 그룹을 새로 생성하기
- 4. 새로 생성한 student 그룹에 설정된 GID 확인하기
- 5. 기존의 GID 2018을 중복 지정하여 db_space 그룹을 새로 생성하기
- 6. 새로 생성한 db_space 그룹에 설정된 GID 확인하기

```
# groupadd professor

# grep professor /etc/group

# groupadd -g 2018 student

# grep student /etc/group

# groupadd -g 2018 -o db_space

# grep db_space /etc/group
```

▋ 그룹 암호 설정 및 삭제

● gpasswd 명령을 사용하여 그룹 암호 설정 및 삭제

\$ gpasswd

기능 그룹 암호 관리 (/etc/group과 /etc/gshadow 파일을 관리)

형식 gpasswd [옵션] 그룹이름 [Enter니

옵션 -a 계정이름 : 계정을 그룹에 추가

-d 계정이름: 계정을 그룹에서 삭제

-r: 그룹 암호를 삭제

● 기존 그룹에 새 멤버 추가:-a 옵션 사용

```
| 예제 6-16 |
```

• Step 01 | 사용자 계정 sno_2017001, sno_2017002를 새로 생성한 다음 확인합니다.

```
# useradd sno_2017001
기능 새 사용자 계정 sno_2017001 생성
형식 useradd [옵션] 계정이름 Enter
```

```
root@localhost:~ - □ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[root@localhost ~] # useradd sno_2017001

[root@localhost -] # useradd sno_2017002

[root@localhost ~] # ls /home
cskisa sno_2017001 sno_2017002

[root@localhost ~] # ■
```

[그림 6-33] 새로운 계정 2개 생성 후 확인

Section 03 기를 관리

• Step 02 | 멤버를 추가할 그룹을 미리 확인해 보면 /etc/group 파일에는 멤버가 하나도 없음을 알 수 있습니다.

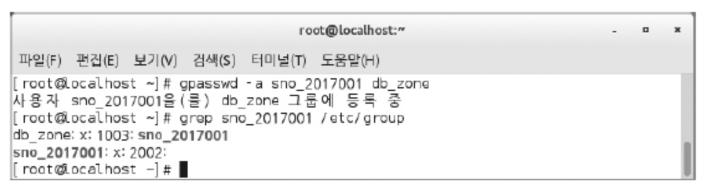
```
# grep sno_2017001 /etc/group
기능 /etc/group 파일에서 문자열 sno_2017001을 찾아서 출력
형식 grep [옵션] 파일명 Enter니
```



[그림 6-34] 그룹 2017에 존재하는 멤버 확인

 Step ○3 | db_zone 그룹에 사용자 계정 sno_2017001을 새 멤버로 추가한 다음 멤버가 정상 적으로 추가되어 있는지를 확인합니다.

```
# gpasswd -a sno_2017001 db_zone
기능 사용자계정 sno_2017001을 db_zone 그룹의 멤버로 추가
형식 gpasswd [옵션] 계정이름 그룹명 EnterJ
```



[그림 6-35] db_zone 그룹에 사용자 계정 sno_2017001 멤버 추가 후 존재확인

 Step 04 | nt_zone 그룹에 사용자 계정 sno_2017002를 새 멤버로 추가한 다음 멤버가 정상 적으로 추가되어 있는지를 확인합니다.

```
# gpasswd -a sno_2017002 nt_zone
기능 사용자 계정 sno_2017002를 nt_zone 그룹의 멤버로 추가
형식 gpasswd [옵션] 계정이름 그룹명 Enter-
```



[그림 6-36] nt_zone 그룹에 사용자 계정 sno_2017002 멤버 추가 후 존재확인

실습 6-5 다음 항목에서 주어진 지시사항을 수행하시오.

- 1, 사용자 계정 pro_001과 pro_002 계정을 새로 생성하기
- 2, 새로 생성한 2개의 사용자 계정이 홈 디렉터리에 존재하는지 확인하기
- 3, professor 그룹에 사용자 계정 pro_001계정을 추가 지정하기
- 4, professor 그룹에 사용자 계정 pro_002계정을 추가 지정하기
- 5, 사용자 계정 pro_001과 pro_002 계정에 지정된 그룹정보 확인하기

```
# useradd pro_001
# useradd pro_002
# ls /home
# gpasswd -a pro_001 professor
# gpasswd -a pro_002 professor
# grep pro_001 /etc/group
# grep pro_002 /etc/group
```

● 기존 그룹에 있는 특정 멤버 삭제:-d 옵션 사용

| 예제 6-17 |

nt_zone 그룹에 존재하는 사용자 계정 sno_2017002 멤버를 삭제한 다음 삭제된 멤버가 존재하는지에 대해 확인합니다.

gpasswd -d sno_2017002 nt_zone 기능 nt_zone 그룹에 있는 sno_2017002 멤버 삭제 형식 gpasswd [옵션] 계정이름 그룹명 EnterJ

```
root@localhost:~ - □ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~] # gpasswd - d sno_2017002 nt_zone
사용자 sno_2017002을(률) 그룹 nt_zone에서 제거하는 중
[root@localhost ~] # grep sno_2017002 / etc/group
sno_2017002: x: 2003:
[root@localhost ~] # ■
```

[그림 6-37] nt_zone 그룹에서 sno_2017002 멤버 삭제 후 확인

● 기존 그룹에 암호 설정 : gpasswd

| 예제 6-18 |

 Step ①1 | 앞에서 생성한 gp_zone 그룹에 암호를 그룹이름과 동일한 gp_zone을 설정한 다음 설정된 암호가 저장된 /etc/gshadow 파일의 내용을 출력해서 그룹 암호가 설정되었음을 확인 합니다.

gpasswd gp_zone 기능 gp_zone 그룹에 암호 설정 형식 gpasswd 그룹명 Enter니

```
root@localhost:~ - □ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[root@localhost ~] # gpasswd gp_zone
gp_zone 그룹의 암호를 바꾸는 중
새 암호:
새 암호를 다시 입력하십시오:
[root@localhost ~] # grep gp_zone /etc/gshadow
gp_zone: $6$Dt, PL/WFLw0/Dxk$BjwGW1bsaHOmgIO0NqcEzpZg1yT, / GucK3Yar0PGqICduZALuQXPG
B/eDr4QxxlmaSkCf0qZRnUxbmnkewZ2d0::
[root@localhost -] # ■
```

[그림 6-38] gp_zone 그룹 암호 설정 후 확인

Section 03 기름 관리

• Step 02 | 같은 방법으로 db_zone 그룹과 nt_zone 그룹에 대한 암호를 그룹이름과 동일하게 설정한 다음 /etc/gshadow 파일의 내용을 확인합니다.

```
root@localhost:~ - ■ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[ root@localhost ~] # gpasswd db_zone
db_zone 그룹의 암호를 바꾸는 중
새 암호:
새 암호를 다시 입력하십시오:
[ root@localhost ~] # gpasswd nt_zone
nt_zone 그룹의 암호를 바꾸는 중
새 암호:
새 암호를 다시 입력하십시오:
[ root@localhost ~] # ■
```

[그림 6-39] db_zone그룹과 nt_zone그룹 암호 설정

● 기존 그룹에 설정되어 있는 그룹 암호 삭제:-r 옵션 사용

| 예제 6-19 |

gp_zone 그룹에 설정되어 있는 암호 gp_zone을 삭제한 다음 /etc/gshadow 파일의 내용을 출력해서 그룹 암호가 삭제되었음을 확인합니다.

```
# gpasswd -r gp_zone
기능 gp_zone 그룹에 설정되어 있는 암호 삭제
형식 gpasswd [옵션] 그룹명 Enter니
```

```
root@localhost:~ - □ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[root@localhost ~] # gpasswd - r gp_zone
[root@localhost -] # grep gp_zone / etc/gshadow
gp_zone:::
[root@localhost -] # ■
```

[그림 6-40] gp_zone 그룹 암호 삭제 후 확인

지금까지 배운 내용을 실습을 통해 실력을 업그레이드 해 봅시다. 이 실습을 진행하기 위해서는 [실습 6-4]와 [실습 6-5]를 먼저 수행해야 합니다.

실습 6-6 다음 항목에서 주어진 지시사항을 수행하시오.

- 1, professor 그룹에 존재하는 pro_001 멤버를 식제하기
- 2, grep 명령으로 pro_001 계정이 professor 그룹에 존재여부를 확인하기
- 3, professor 그룹에 암호를 123456으로 설정하기
- 4. student 그룹에 암호를 123456으로 설정하기
- 5. grep 명령으로 professor 그룹과 student 그룹에 설정된 암호에 대한 정보 확인하기
- 6. student 그룹에 설정된 암호 삭제하기
- 7. 암호가 삭제된 student 그룹의 암호정보 확인하기

```
# gpasswd –d pro_001 professor
# grep pro_001 /etc/group
# gpasswd professer
새 암호 ; 123456
새 암호 다시 입력: 123456
# gpasswd student
새 암호 ; 123456
새 암호 다시 입력: 123456
# grep professer /etc/gshadow
# grep student /etc/gshadow
# gpasswd –r student
# grep student /etc/gshadow
```

Section 03 기료 관리

▮ 소속 그룹 관리

● 기존에 등록되어 있는 그룹 목록 출력 : cat 명령 사용

| 예제 6-20 |

```
# cat /etc/group

기능 리눅스 시스템에 등록되어 있는 그룹 목록 출력
형식 cat [옵션] 파일명 EnterJ
```

```
root@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost -] # cat /etc/group
root: x: 0:
bin: x: 1:
daemon: x: 2:
svs: x: 3:
adm: x: 4:
tty: x: 5:
disk: x: 6:
lp: x: 7:
mem: x: 8:
kmem: x: 9:
wheel: x: 10: cskisa
cdrom: x: 11:
mail: x: 12: postfix
man: x: 15:
```

(a) 리눅스 시스템에 등록되어 있는 그룹 목록-1

```
root@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
sshd: x: 74:
dovecot: x: 97:
dovenull: x: 980:
stapusr: x: 156:
stapsys: x: 157:
stapdev: x: 158:
tcpdump: x: 72:
cskisa: x: 1000:
user_002: x: 1002:
db_zone: x: 1003: sno_2017001
nt_zone: x: 2017:
gp_zone: x: 2017:
sno_2017001; x: 2002;
sno 2017002; x: 2003;
[root@Localhost -]#
```

(b) 리눅스 시스템에 등록되어 있는 그룹 목록-2

[그림 6-41] 리눅스 시스템에 등록되어 있는 그룹 목록 출력

Section 03 기를 관리

● 사용자 계정으로 접속모드 변경: su - 계정이름

| 예제 6-21 |

현재 소속된 그룹을 변경하기 위해서는 먼저 관리자 계정이 아닌 사용자 계정으로 접속한 다음에 진행해야 합니다. su명령으로 sno_2017001 사용자 계정으로 접속모드를 변경합니다.

```
# su - sno_2017001
기능 sno_2017001 사용자 계정으로 변경
형식 su [옵션] 계정이름 EnterJ
```

```
sno_2017001@localhost:~ - ㅁ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~] # su - sno_2017001
[sno_2017001@localhost -] $ ■
```

[그림 6-42] 접속모드를 사용자 계정으로 변경

● 소속 그룹정보 출력: id 명령 사용

\$ id

기능 현재 사용자의 실제 ID와 유효 사용자 ID, 그룹 ID를 출력

형식 id [옵션] [계정이름] [Enter↓]

옵션 -g: 사용자의 그룹 ID 출력

-G : 추가 그룹의 ID 출력

-u: 사용자의 UID 출력

-n: u, q, G 옵션과 함께 사용하며 해당하는 ID의 이름 출력

-r:u,g,G 옵션과 함께 사용하며 해당하는 실제 ID를 출력

Section 03 기를 관리

● 현재 사용자 계정의 그룹 정보 출력

| 예제 6-22 |

현재 사용자 계정의 그룹에 대한 정보를 출력합니다.

\$ id

기능 현재 사용자의 실제 ID와 유효 사용자 ID, 그룹 ID를 출력 형식 id [옵션] [계정이름] [Enter_]

```
sno_2017001@localhost:~ - ㅁ x
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[root@Localhost ~] # su - sno_2017001
[sno_2017001@Localhost ~] $ id
uid=2002(sno_2017001) gid=2002(sno_2017001) groups=2002(sno_2017001), 1003(db_zon
e) context=unconfined_u; unconfined_r; unconfined_t; s0-s0; c0, c1023
[sno_2017001@Localhost ~] $ ■
```

[그림 6-43] sno_2017001 계정의 그룹 정보 출력

Section 03 기를 관리

● 소속 그룹 변경: newgrp 명령 사용

\$ newgrp 그룹명 기능 소속 그룹을 지정한 그룹으로 변경 형식 newgrp 그룹명 Enter.」

● sno_2017001 사용자 계정이 속한 그룹 변경 후 내용 출력

| 예제 6-23 |

sno_2017001 사용자 계정으로 접속된 상태에서 이 계정이 속한 그룹을 nt_zone으로 변경한 다음 id 명령으로 변경된 내용을 확인합니다..

\$ newgrp nt_zone 기능 소속 그룹을 지정한 nt_zone 그룹으로 변경 형식 newgrp 그룹명 Enter-

```
sno_2017001@localhost:~ - ㅁ ×
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

[sno_2017001@localhost ~] $ newgrp nt_zone
암호:
[sno_2017001@localhost ~] $ id
uid=2004(sno_2017001) gid=2017(nt_zone) groups=2017(nt_zone), 1003(db_zone), 2004(sno_2017001) context=unconfined_u: unconfined_r: unconfined_t: s0-s0: c0, c1023
[sno_2017001@localhost -] $ ■
```

[그림 6-44] sno_2017001 계정의 소속 그룹변경과 정보 출력

실습 6-7 다음 항목에서 주어진 지시사항을 수행하시오.

- 1, 터미널 창에서 professor 그룹에 존재하는 pro_002 계정으로 접속하기
- 2, pro_002 계정의 그룹정보 출력하기
- 3, pro_002 계정의 소속 그룹을 professor 그룹으로 변경하기
- 4. pro_002 계정의 변경된 그룹 정보 출력하기
- 5, root 계정으로 프롬프트 변경하기

```
# su - pro_002
# id
# newgrp pro_002
# id
\# su - root
암호: spacezone
#
```



소유자와 소유그룹 변경

• 파일 및 디렉터리의 소유자와 소유그룹 변경하기



[1. 파일 및 디렉터리 소유자와 소유그룹에 대해 살펴봅니다. 2. 파일 및 디렉터리 소유자와 소유그룹 변경방법에 대해 실습합니다.

Section 04 〉 오유까와 오유그룹 변경

- ▌파일과 디렉터리의 소유권과 그룹 소유권 변경 : chown 명령 사용
 - chown 명령어 사용형식

chown

기능 파일이나 디렉터리의 소유권 및 그룹 소유권을 변경

형식 chown [옵션] [계정이름] [파일명 또는 디렉터리명] (Enter L)

옵션 -R: 하위 디렉터리를 모두 포함하여 소유권과 그룹 소유권을 변경

Section 04 〉 소유자와 소유그룹 변경

● chown 명령으로 파일과 디렉터리의 소유권 및 그룹 소유권 변경

| 예제 6-24 |

• Step □1 | 먼저 root 계정으로 로그인 된 상태에서 사용자 계정인 sno_2017002로 접속을 변 경한 다음 work 디렉터리와 빈파일 juso.txt를 생성합니다.

1s /home

기능 /home 디렉터리에 있는 사용자 계정 목록 출력

형식 1s 디렉터리명 (Enter...)

su - sno 2017002

기능 사용자 계정 sno_2017002로 접속 변경

형식 su - 계정이름 (Enter↓)

Section 04) 그룹 오유자와 오유그룹 변경

```
$ mkdir work
기능 새로운 work 디렉터리를 생성
형식 mkdir 디렉터리명 Enter니
```

```
$ touch juso.txt
```

기능 새로운 빈파일 juso.txt를 생성 형식 touch 파일명 Enter니

```
sno_2017002@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~]# ls /home
cskisa sno_2017001 sno_2017002
[root@localhost ~]# su - sno_2017002
sno 2017002@Localhost ~| $ ls
sno 2017002@localhost ~| $ mkdir work
sno_2017002@Localhost ~] { touch juso, txt
sno 2017002@Localhost -| $ ls -l
한계 0
-rw-rw-r--, 1 sno 2017002 sno 2017002 0 1월 6 08:30 juso.txt
drwxrwxr-x, 2 sno_2017002 sno_2017002 6 1월 6 08:30 work
sno 2017002@Localhost -|$
```

[그림 6-45] 사용자 계정으로 변경 후 work 디렉터리와 juso,txt 파일 생성

Section 04) 그룹 소유까와 소유그룹 변경

• Step 02 | chown 명령은 일반 사용자 계정에서는 사용할 수 없기 때문에 사용자 계정에서 work 디렉터리와 juso.txt 파일을 생성한 다음에는 다시 root 계정으로 접속해야 합니다.

S su - root

기능 root 계정으로 접속 변경

형식 su - 계정이름 [Enter...]

Section 04) 그룹 오유까와 오유그룹 변경

• Step 03 | root 계정으로 접속된 상태에서 sno_2017002 계정에 존재하는 파일의 목록과 소유 자 권한을 확인한 다음 chown 명령으로 work 디렉터리의 소유자를 cskisa 계정으로 변경하 고 소유자가 변경된 사항을 확인합니다.

1s -1 /home/sno 2017002

기능 사용자 계정 sno_2017002에서 생성한 디렉터리와 파일 존재 확인 형식 1s [옵션] 디렉터리명/계정이름 [Enter,]

chown cskisa /home/sno 2017002/work

기능 sno_2017002 계정에서 생성한 work 디렉터리의 소유자를 cskisa로 변경 형식 chown 계정이름 디렉터리명 Enter,

Section 04) 그룹 오유까와 오유그룹 변경

```
root@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~]# ls -l /home/sno 2017002
합계 0
-rw-rw-r--, 1 sno_2017002 sno_2017002 0 1월 6 08:30 juso, txt
drwxrwxr-x, 2 sno_2017002 sno_2017002 6 1월 6 08:30 work
[root@localhost -]# chown cskisa /home/sno_2017002/work
[root@localhost ~]# ls -l /home/sno 2017002
합계 0
-rw-rw-r--, 1 sna_2017002 sna_2017002 0 1월 6 08:30 jusa.txt
drwxrwxr-x. 2 cskisa sno_2017002 6 1월 6 08:30 work
[root@localhost ~]#
```

[그림 6-46] work 디렉터리에 대한 소유자를 cskisa 계정으로 변경

Section 04

오유자와 오유그룹 변경

● sno_2017002 계정에서 생성한 juso.txt 파일의 사용자 계정과 그룹을 동시에 지정하여 변경

| 예제 6-25 |

항상 chown 명령을 사용하기 전에는 반드시 root 계정으로 로그인 된 상태인지를 먼저 확인합니다. 그리고 사용자 계정 sno_2017002에서 생성한 juso.txt 파일의 소유자와 그룹을 모두 cskisa로 변경한 다음 변경된 사항을 확인합니다.

ls -l /home/sno_2017002

기능 사용자 계정 sno_2017002에서 생성한 디렉터리와 파일 존재 확인 형식 1s [옵션] 디렉터리명/계정이름 [Enter-]

Section 04 〉 소유까와 소유그룹 변경

```
# chown cskisa:cskisa /home/sno 2017002/juso.txt
```

기능 sno_2017002 계정에서 생성한 juso,txt 파일의 소유자와 그룹을 cskisa로 변경 형식 chown 계정이름:그룹이름 파일명 Enter니

```
root@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움맏(H)
[root@Localhost ~]# ls -l /home/sno 2017002
합계 0
-rw-rw-r--, 1 sno_2017002 sno_2017002 0 1월 6 08:30 juso.txt
drwxrwxr-x, 2 cskisa sno 2017002 6 1월 6 08:30 work
[root@localhost -] # chown cskisa: cskisa /home/sno 2017002/juso.txt
[root@localhost ~]# ls -l /home/sno 2017002
합계 0
-rw-rw-r--, 1 cskisa cskisa 0 1월 6 08:30 juso, txt
drwxrwxr-x. 2 cskisa sno 2017002 6 1월 6 08:30 work
[root@localhost ~]#
```

[그림 6-47] juso,txt 파일에 대한 소유자와 그룹을 cskisa로 변경

Section 04 〉 소유자와 소유그룹 변경

- 파일과 디렉터리의 소유그룹 변경 : chgrp 명령 사용
- chgrp 명령어 사용형식

chgrp

기능 파일이나 디렉터리의 그룹 소유권을 변경

형식 chgrp [옵션] [계정이름] [파일명 또는 디렉터리명] Enter.

옵션 -R: 하위 디렉터리를 모두 포함하여 그룹 소유권을 변경

Section 04 〉 소유자와 소유그룹 변경

● sno_2017001 계정에 새로운 디렉터리와 파일 생성 후 소유그룹 변경

| 예제 6-26|

• Step □1 | 먼저 root 계정으로 로그인 된 상태에서 사용자 계정인 sno 2017001로 접속을 변 경한 다음 계층적 디렉터리 workspace/test/ex dir 디렉터리와 scale.txt 빈 파일을 생성 합니다.

su - sno 2017001

기능 사용자 계정 sno_2017001로 접속 변경

형식 su - 계정이름 Enter.□

Section 04 오유자와 오유그룹 변경

\$ mkdir -p workspace/test/ex dir

기능 새로운 workspace/test/ex_dir 계층적 디렉터리를 생성

형식 mkdir [옵션] 계층적 디렉터리명 Enter↓

\$ touch scale.txt

기능 새로운 빈파일 scale,txt를 생성

형식 touch 파일명 Enter니

\$ 1s -1R

기능 현재 디렉터리에 존재하는 하위 디렉터리 리스트를 출력

형식 1s [옵션] 디렉터리명/계정이름 Enter↓]

Section 04 〉 소유까와 소유그룹 변경

```
sno_2017001@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~] # su - sno 2017001
마지막 로그인: 일 1월 1 16:17:43 KST 2017 일시 pts/0
[sno_2017001@localhost ~] $ touch scale txt
[sno 2017001@localhost ~] { ls -lR
합계 0
-rw-rw-r--, 1 sno_2017001 sno_2017001 0 1월 6 09:37 scale, txt
drwxrwxr-x. 3 sno 2017001 sno 2017001 17 1월 6 09:37 workspace
./workspace:
합계 0
drwxrwxr-x, 3 sno 2017001 sno 2017001 19 1월 6 09:37 test
./workspace/test:
한계 0
drwxrwxr-x, 2 sno 2017001 sno 2017001 6 1월 6 09:37 ex dir
./workspace/test/ex_dir:
한계 0
[sno 2017001@Localhost ~] $
```

[그림 6-48] sno_2017001 계정에서 계층적 디렉터리와 빈파일 생성

Section 04 〉 소유자와 소유그룹 변경

• Step 02 | chgrp 명령 또한 일반 사용자 계정에서는 사용할 수 없기 때문에 사용자 계정에서 다시 root 계정으로 접속해야 합니다.

```
$ su - root
기능 root 계정으로 접속 변경
형식 su - 계정이름 Enter↓]
```



[그림 6-49] 사용자 계정에서 root 계정으로 전환

Section 04 〉 소유자와 소유그룹 변경

• Step 03 | root 계정으로 접속된 상태에서 sno 2017001 계정에 존재하는 파일의 목록과 소유 자 권한을 확인한 다음 chgrp 명령과 함께 옵션 - R을 지정하여 소유자와 소유그룹을 cskisa 계정으로 변경하고 변경된 사항을 확인합니다.

ls -1 /home/sno 2017001

기능 사용자 계정 sno_2017001에서 생성한 디렉터리와 파일 존재 확인 형식 1s [옵션] 디렉터리명/계정이름 Enter↓]

chgrp -R cskisa /home/sno 2017001/workspace

기능 sno_2017002 계정에서 생성한 workspace 디렉터리의 소유자를 cskisa로 변경

형식 chown 계정이름 디렉터리명 Enter-

Section 04 〉 소유까와 소유그룹 변경

\$ 1s -1R

기능 현재 디렉터리에 존재하는 하위 디렉터리 리스트를 출력 형식 1s [옵션] 디렉터리명/계정이름 Enter.

```
root@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~] # ls -l /home/sno 2017001
합계 0
-rw-rw-r--, 1 sno 2017001 sno 2017001 0 1월 6 09:37 scale,txt
drwxrwxr-x. 3 sno_2017001 sno_2017001 17 1월 6 09:37 workspace
[root@localhost ~] # chgrp -R cskisa /home/sno_2017001/workspace
[root@localhost ~]# ls -lR /home/sno 2017001
/home/sno 2017001:
합계 0
-rw-rw-r--, 1 sno_2017001 sno_2017001 0 1월 6 09:37 scale.txt
drwxrwxr-x, 3 sno 2017001 cskisa 17 1월 6 09:37 workspace
/home/sno 2017001/workspace:
합계 0
drwxrwxr-x, 3 sno_2017001 cskisa 19 1월 6 09:37 test
/home/sno 2017001/workspace/test:
합계 0
drwxrwxr-x, 2 sno_2017001 cskisa 6 1월 6 09:37 ex_dir
/home/sno_2017001/workspace/test/ex_dir:
합계 0
[root@localhost ~]#
```

[그림 6-50] 계층적 디렉터리 workspace에 대한 소유그룹을 cskisa 계정으로 변경

Section 04 〉 소유자와 소유그룹 변경

● sno 2017001 계정에 생성한 scale.txt 파일의 소유그룹을 cskisa 계정으로 변경한 다음 변경된 사항 출력

| 예제 6-27 |

chgrp 명령 또한 사용하기 전에는 반드시 root 계정으로 로그인 된 상태인지를 먼저 확인합니다. 그리고 사용자 계정 sno 2017001에서 생성한 scale txt 파일의 소유자와 그룹을 모두 cskisa로 변경한 다음 변경된 사항을 확인합니다.

ls -1 /home/sno 2017001

기능 사용자 계정 sno_2017001에서 생성한 디렉터리와 파일 존재 확인 형식 1s [옵션] 디렉터리명/계정이름 Enter.

Section 04 〉 소유자와 소유그룹 변경

chgrp cskisa /home/sno 2017001/scale.txt

기능 sno_2017001 계정에서 생성한 scale,txt 파일의 소유그룹을 cskisa로 변경 형식 chgrp 그룹이름 파일명 Enter↓

```
root@localhost:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[root@localhost ~]# ls -l /home/sno 2017001
한계 0
-rw-rw-r--. 1 sno 2017001 sno 2017001 0 1월 6 09:37 scale.txt
drwxrwxr-x, 3 sno_2017001 cskisa 17 1월 6 09:37 workspace
[root@Localhost ~] # chgrp cskisa /home/sno 2017001/scale.txt
[root@localhost ~]# ls -l /home/sno 2017001
한계 0
-rw-rw-r--, 1 sno 2017001 cskisa 0 1월 6 09:37 scale, txt
drwxrwxr-x, 3 sno 2017001 cskisa 17 1월 6 09:37 workspace
[root@localhost ~]#
```

[그림 6-51] scale.txt 파일에 대한 소유그룹을 cskisa 계정으로 변경



리눅스 시스템 종료

• 리눅스 시스템을 종료하는 방법 이해하기



- 1. 현재 사용 중인 리눅스 시스템을 종료하는 방법의 종류를 살펴봅니다. 2. 시스템을 종료하는 명령어와 옵션에 대해 이해합니다.

 - 3. 리눅스 시스템을 재부팅하는 방법을 살펴봅니다.

■ 리눅스 시스템 종료 : shutdown 명령 사용

● shutdown 명령어 사용형식

shutdown

기능 리눅스 시스템 종료

형식 shutdown [옵션] [시간] [메시지] Enter↓]

옵션 -c: 이전에 수행했던 shutdown 명령 취소

-f: 빠른 재시작 (fsck 생략 가능)

-h: 종료하며 halt 상태로 이동

-k: 실제 종료는 아니고 사용자들에게 메시지만 전달

-r:종료 후 재시작

시간: 종료할 시간 (hh:mm, +m, now)

메시지: 접속된 모든 사용자에게 보낼 메시지

● 이외에도 poweroff, halt, reboot 명령으로도 리눅스 시스템 종료

리눅스 시스템 종료

시스템 즉시 종료

리눅스 시스템을 즉시 종료하려면 shutdown 명령과 함께 - h 옵션을 사용하여 현재 시각을 의미하는 now를 지정하면 됩니다.

shutdown -h now

종료 메시지 보낸 후 종료하기

리눅스 시스템은 여러 사용자가 동시에 접속하여 사용하는 시스템이기 때문에 리눅스를 종료하기에 앞서 접속된 사용자들에게 미리 종료 메시지를 전달해 주는 것이 바람직합니다. 만약 3분 후 리눅스 시스템을 종료한다는 메시지를 보내고자 하는 경우에는 다음과 같이 명령을 수행하면 됩니다.

shutdown -h +3 "It is 3 minutes before the shutdown."

Section 05

리눅스 시스템 종료

● 실제로 시스템을 종료하지 않고 종료 메시지만 보내기

| 예제 6-28 |

리눅스 시스템을 실제로 종료하지는 않고 5분 후 시스템을 종료한다는 메시지만 사용자들에게 전 달하려면 다음과 같이 명령을 수행하면 됩니다.

shutdown -k 5



[그림 6-52] 5분후 시스템 종료 안내

기타 명령어

shutdown 명령 외에 시스템을 종료하거나 재시작을 위해 사용할 수 있는 명령으로 런레벨 변경과 halt, poweroff, reboot 명령이 있습니다. halt, poweroff, reboot 명령은 /var/log/vtmp파일에 시스템 종료와 관련된 기록을 남기고 시스템을 종료하거나 재시작합니다. 터미널 창에서마운트 및 패키지 설치 등으로 인해 시스템을 재부팅할 경우가 발생하게 됩니다. 이와 같은 경우리눅스 시스템 전체를 종료하지 않고 터미널 창에서 시스템을 재부팅할 때는 reboot 명령을 사용하면 됩니다.

reboot

이상으로 관리자 계정접속과 사용자 관리계정 및 그룹관리, 소유자와 소유그룹 변경에 대해 살펴 봤습니다. 다음 장에서는 리눅스 파일시스템과 디스크 관리에 대해서 다루겠습니다.

Chapter 06

최상의 노력에 따른 인고의 가치는 반드시 증명될 수 있습니다!

Thank You