

제14장 시스템 관리

VEDA

14.1 시스템 관리자

시스템 관리자

- 슈퍼유저(superuser)
 - 시스템을 관리할 수 있는 사용자
 - 슈퍼유저가 사용하는 계정이 root이다
- 슈퍼유저 로그인
 - 직접 root 계정으로 로그인
 - 다른 계정으로 로그인 후
\$ su [사용자명]

▶ 3

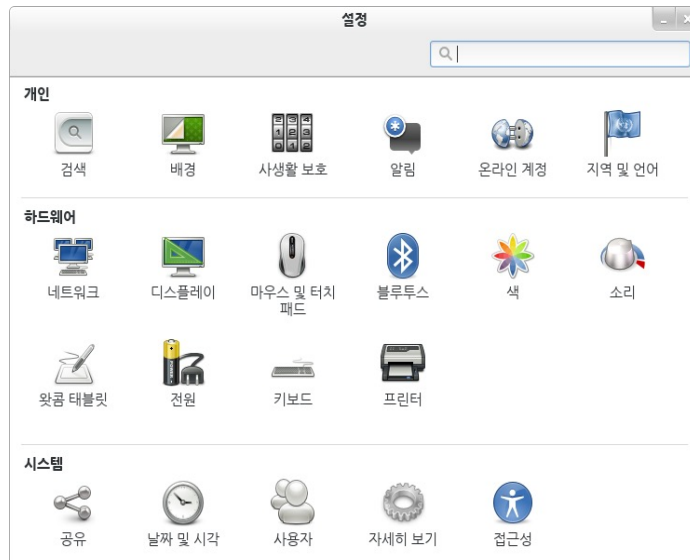
시스템 관리자

- 시스템 관리자의 역할
 - 사용자 등록 및 삭제
 - 소프트웨어 설치, 업그레이드, 삭제
 - 하드웨어 추가 설치
 - 시스템 보안
 - 데이터 백업

▶

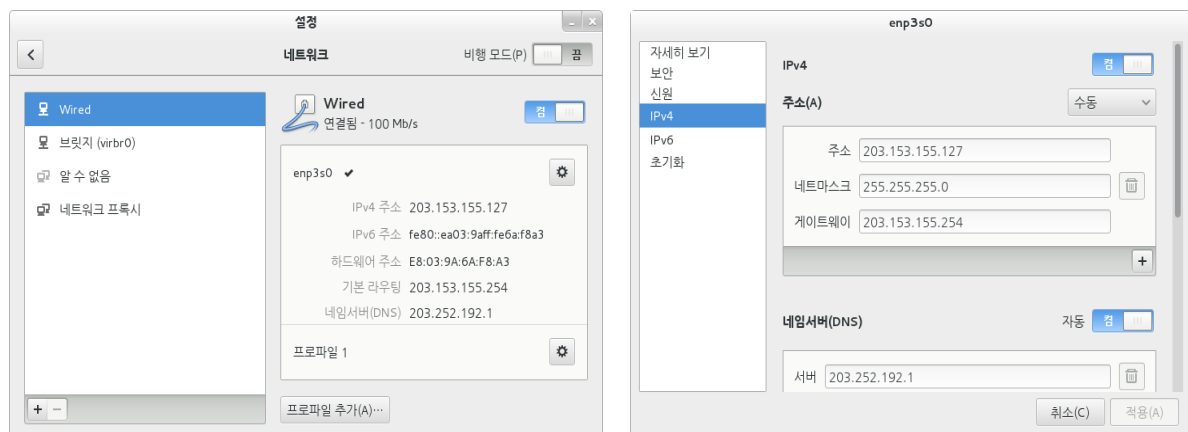
시스템 설정

- 시스템 설정은 [프로그램] -> [시스템 도구] -> [설정] 메뉴



네트워크 설정

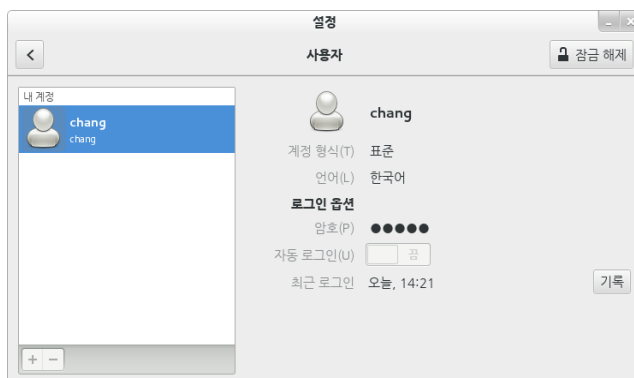
- 네트워크 설정 창
[시스템 도구] -> [설정] -> [네트워크]
- 수동설정
IP 주소, 게이트웨이, DNS 서버



14.2 사용자 계정 관리

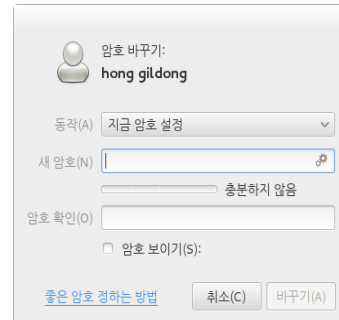
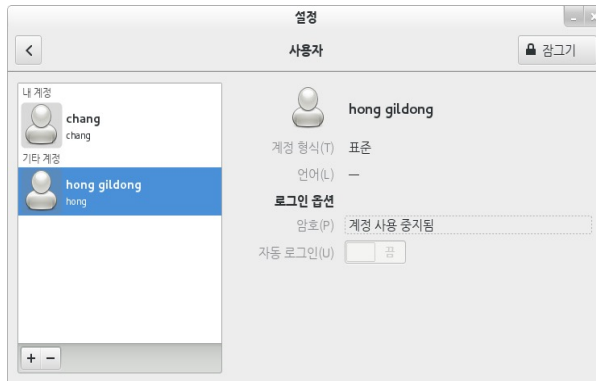
사용자 계정 추가

- 사용자 관리자 도구
 - [시스템 도구] -> [설정] -> [사용자]



사용자 계정 암호 바꾸기

- 암호(P) 클릭하여 암호 바꾸기



사용자 계정 추가 명령어

- 사용자 계정 추가

```
# useradd [옵션] 사용자명
```

새로운 사용자 계정을 생성한다.

- 패스워드 설정
 - # passwd 사용자명
 - 관련 파일: /etc/passwd, /etc/shadow

- 사용자 계정 삭제

```
# userdel [-r] 사용자명
```

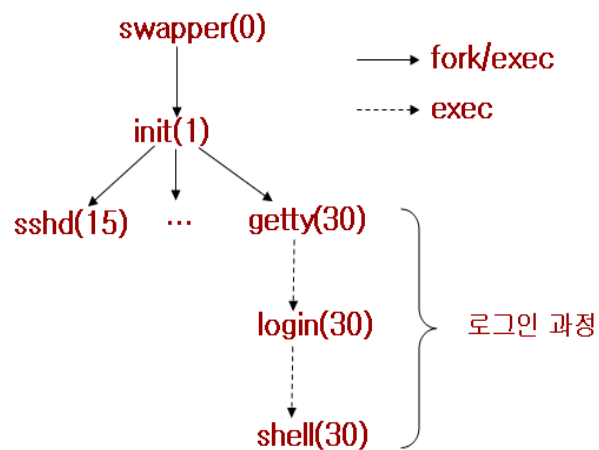
사용자 계정을 삭제한다.

-r 옵션을 사용하면 사용자의 홈 디렉터리도 삭제한다.

14.3 시스템 부팅과 종료

시스템 부팅

- 시스템 부팅은 fork/exec 시스템 호출을 통해 이루어진다.



시스템 부팅

- swapper(스케줄러 프로세스)
 - 커널 내부에서 만들어진 프로세스로 프로세스 스케줄링을 한다
- init(초기화 프로세스)
 - /etc/inittab 파일에 기술된 대로 시스템을 초기화
- 서비스 데몬 프로세스
 - 서비스들을 위한 데몬 프로세스들이 생성된다. 예: ftpd
- getty 프로세스
 - 로그인 프롬프트를 내고 키보드 입력을 감지한다.
- login 프로세스
 - 사용자의 로그인 아이디 및 패스워드를 검사
- shell 프로세스
 - 시작 파일을 실행한 후에 쉘 프롬프트를 내고 사용자로부터 명령어를 기다린다

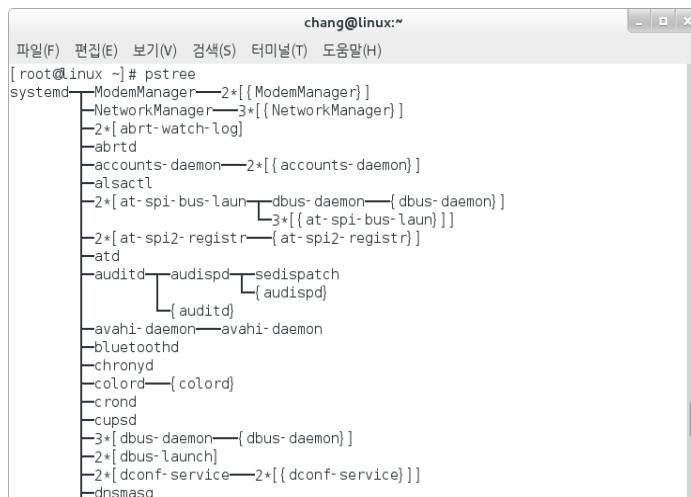
▶ 13

프로세스 트리 출력

● 사용법

```
$ pstree
```

실행중인 프로세스들의 부모자식 관계를 트리 형태로 출력한다



▶ 14

런레벨

- init 프로세스(/etc/init)
 - /etc/inittab 파일에 따라 시스템을 시작하고 초기화한다.
 - **런레벨(runlevel)**에 따라 다음과 같이 시스템을 초기화한다.

런레벨	부팅 환경
0	시스템을 정지한다.
1	단일 사용자 모드로 부팅하며 네트워크 기능을 사용 안 한다. root로만 로그인 가능하다.
2	다중 사용자 모드로 부팅하며 네트워크 기능을 사용 안 한다.
3	정상 작동 모드로 다중 사용자 모드, 텍스트 인터페이스로 부팅한다.
4	사용되지 않으며 특별한 시스템을 구현하기 위한 예약된 레벨이다.
5	정상 작동 모드로 부팅하며 x-윈도우 인터페이스로 부팅한다.
6	시스템을 재부팅 한다.

런레벨

- Runlevel 명령어를 이용하여 현재 부팅 레벨을 확인할 수 있다.

```
$ runlevel
```

시스템의 현재 부팅 레벨을 알려준다.

- 예
 - \$ runlevel
 - N 5
- 런레벨을 변경
 - 시스템 실행 후에도 init 명령어로 런레벨을 변경 가능
- 예
 - \$ init 6

프로세스 관리 도구 top

\$ top [-d 숫자]

프로세스별 CPU와 메모리 점유율 등을 출력한다.

-d 옵션을 사용하면 숫자만큼의 시간마다 다시 출력한다.

\$ top

top - 14:42:23 up 114 days, 2:53, 5 users, load average:0.14, 0.07, 0.02

Tasks: 228 total, 1 running, 227 sleeping, 0 stopped, 0 zombie

Cpu(s):0.0%us, 0.2%sy, 0.0%ni, 99.8%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st

Mem: 2061948k total, 1938932k used, 123016k free, 286124k buffers

Swap: 4128760k total, 0k used, 4128760k free, 766936k cached

PID USER PRI NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND

9818 chang 20 0 2656 1172 872 R 0.3 0.1 0:00.14 top

1 root 20 0 2828 1292 1096 S 0.0 0.1 0:07.21 init

2 root 20 0 0 0 0 S 0.0 0.0 0:00.13 kthreadd

3 root RT 0 0 0 0 S 0.0 0.0 0:00.25 migration/0

4 root 20 0 0 0 0 S 0.0 0.0 0:00.01 ksoftirqd/0

5 root RT 0 0 0 0 S 0.0 0.0 0:00.00 watchdog/0

6 root RT 0 0 0 0 S 0.0 0.0 0:00.21 migration/1

7 root 20 0 0 0 0 S 0.0 0.0 0:00.03 ksoftirqd/1

▶ 8 root RT 0 0 0 0 S 0.0 0.0 0:00.00 watchdog/1

시스템 종료

- 우측 상단 메뉴

- [컴퓨터 끄기]
- [사용자 바꾸기]
- [로그아웃]
- [잠그기]

- shutdown 명령어

- # shutdown +5
- # shutdown now

shutdown 시간 [경고메시지]

지정된 시간에 시스템을 종료한다.

모든 사용자에게 보낼 경고 메시지를 명시할 수 있다.



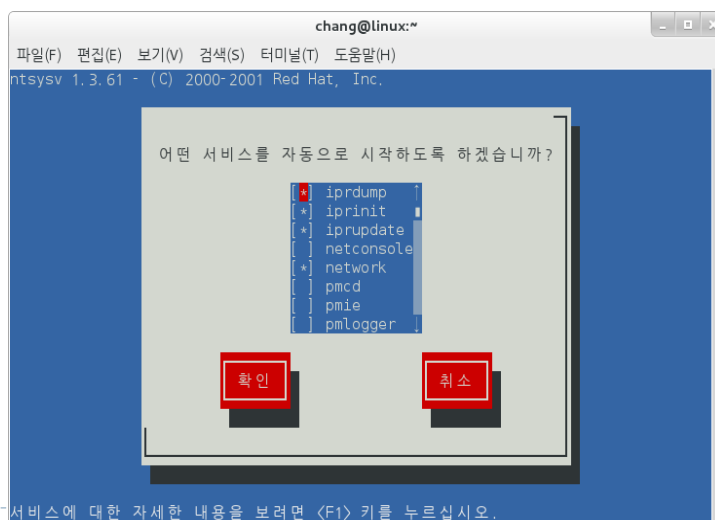
14.4 서비스 관리

서비스 설정 유틸리티: ntsysv

- 부팅 시에 실행할 서비스 설정 유틸리티

ntsysv [옵션]

부팅 시 런레벨 별로 init 시스템이 실행하거나 실행하지 않을 서비스들을 설정한다.



서비스에 대한 자세한 내용을 보려면 <F1> 키를 누르십시오.

서비스 설정 유틸리티: chkconfig

- 부팅 시에 자동 실행할 서비스들을 설정하는 유틸리티

```
# chkconfig [옵션] 서비스명 [상태]
```

부팅 시에 init 시스템에 의해 자동 실행할 각종 서비스들을 설정한다.

- #chkconfig -list

- chkconfig 관리항목에 특정 서비스를 등록 혹은 삭제

```
# chkconfig --add 서비스명
```

chkconfig 관리항목에 특정 서비스를 등록한다.

```
# chkconfig --del 서비스명
```

chkconfig 관리항목에 특정 서비스를 삭제한다.

- # chkconfig --add httpd

서비스 설정 유틸리티: chkconfig

- 부팅 시 특정서비스가 실행 또는 실행되지 않도록 설정

```
# chkconfig 서비스명 on
```

부팅 시 지정된 서비스가 실행되도록 설정한다.

```
# chkconfig 서비스명 off
```

부팅 시 지정된 서비스가 실행되지 않도록 설정한다.

- # chkconfig httpd on

서비스 직접 제어: service

- 서비스 데몬에 대해서 실행, 중지, 재시작 등을 직접 제어

service 서비스 명령어

시스템 서비스를 조작하기 위해 서비스 데몬을 명령어에 따라 제어한다.

명령어	설명
start	서비스 데몬을 실행한다.
stop	서비스 데몬을 중지한다.
restart	서비스 데몬을 재시작한다.
status	서비스 데몬의 상태를 보여준다.

- 예

```
# service httpd status
```

openssh-daemon (pid 1698)를 실행하고 있습니다.

```
# service httpd stop
```

httpd 를 정지 중: [OK]

```
# service httpd start
```

httpd (을)를 시작 중: [OK]



14.5 systemd를 이용한 부팅

init과 systemd

- Init과 system
 - CentOS 7 등 최신 배포판은 systemd가 init를 대체하여 부팅 서비스
 - init는 없어지지는 않았지만 제 기능을 하지 못함.
- Systemd의 기능
 - 서비스, 시스템 로그, 사용자, 시간, 원격, cgroup 관리 등
 - 시스템의 전반적인 관리를 전부 systemd가 담당
- Systemd 관련 명령어와 설정 파일
 - Systemd 관련 명령어들: /usr/lib/systemd 디렉터리
 - Systemd 설정파일들: /etc/systemd 디렉터리
 - # ls /etc/systemd
 - Bootchart.conf journal.conf logind.conf system system.conf user user.conf
 - Service, target 설정 파일: /usr/lib/systemd/system 디렉터리

부팅 관련 systemd 명령어

- 부팅 관련 systemd 명령어

명령어	설명
systemd-analyze	부팅에 걸린시간 표시
systemd-analyze blame	부팅 시 서비스별 걸린 시간표시
systemd-analyze plot	부팅 시 서비스별 걸린 시간을 정렬해서 표시
systemd-analyze critical-chain	부팅 시 시간이 많이 걸리는 서비스들을 트리 형태로 엮어서(chain) 표시
journalctl	부팅을 포함한 전체적인 시스템 로그
journalctl -b	마지막 부팅 이후 시스템 로그
hostnamectl	호스트 이름표시
hostnamectl set-hostname 이름	호스트이름 변경

```
# systemd-analyze
Startup finished in 677ms (kernel) + 1.824s (initrd) + 31.308s (userspace) = 33.809s
```

서비스 상태 표시: systemctl 명령어

- systemd의 서비스를 관리하기 위한 명령어로 서비스 상태를 표시할 수 있다

명령어	서비스 상태 표시 기능
systemctl	전체 서비스 목록
systemctl list-unit-files	서비스 목록 (활성화 여부 표시)
systemctl --failed	부팅 시 실행에 실패한 서비스목록
systemctl is-enabled 서비스명	특정 서비스 활성화 여부
systemctl is-active 서비스명	특정 서비스 현재 동작 여부
systemctl status -l 서비스명	특정 서비스의 자세한 상태

```
# systemctl status -l crond
crond.service - Command Scheduler
Loaded: loaded (/usr/lib/systemd/system/crond.service; enabled)
Active: active(running) since 토 2016-10-15 13:17:12 KST; 1 weeks 6 days ago
Main PID: 852 (crond)
CGroup: /system.slice/crond.service
└─852 /usr/sbin/crond -n
```

서비스 제어: systemctl 명령어

- systemd의 서비스를 관리하기 위한 명령어로 서비스를 제어할 수 있다

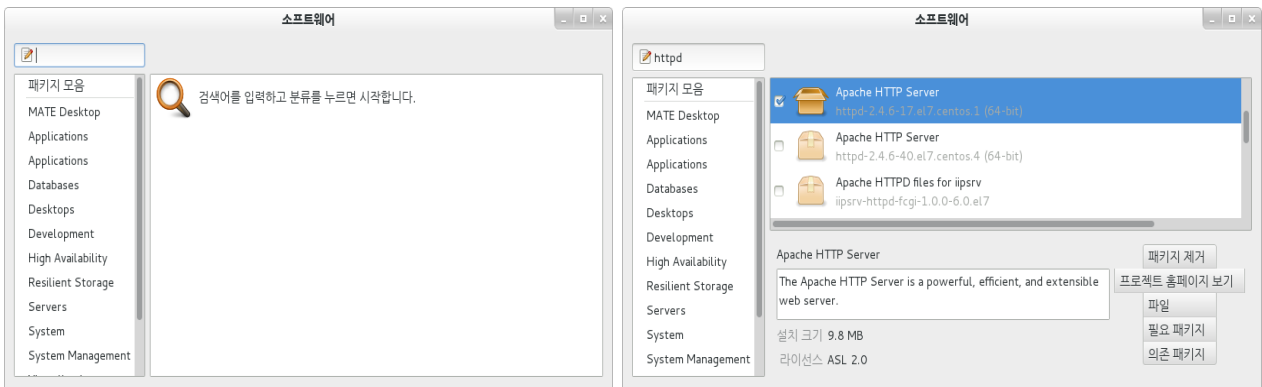
명령어	서비스 제어 기능
systemctl enable 서비스명	서비스 활성화
systemctl disable 서비스명	서비스 비활성화
systemctl start 서비스명	서비스 시작
systemctl stop 서비스명	서비스 종료
systemctl restart 서비스명	서비스 재시작
systemctl reload 서비스명	서비스 갱신
systemctl kill 서비스명	서비스 관련 프로세스도 모두 죽임
systemctl daemon-reload	서비스 설정 정보를 데몬에 반영

- 예: httpd 서비스를 활성화하여 부팅 시 자동으로 실행
 - # systemctl enable httpd
- 예: 서비스를 바로 직접 실행
 - # systemctl start httpd

14.6 소프트웨어 설치

소프트웨어 설치 도구

- 자동 소프트웨어 설치 및 삭제
 - [프로그램] -> [시스템 도구] -> [소프트웨어] 도구
 - 검색어 입력: 설치 혹은 삭제를 위한 소프트웨어 패키지 검색
 - 원하는 패키지를 선택하고 [패키지 설치] 버튼을 클릭하여 설치



레드햇 패키지 매니저 RPM

- RPM(RedHat Package Manager)의 특징
 - 쉬운 패키지 설치/삭제 가능
 - 패키지 업그레이드가 간단
 - 설치 시간이 컴파일 하는 경우보다 빠름
 - 패키지의 검증이 가능
 - 질의를 통하여 패키지에 대한 자세한 정보 확인 가능

모드	사용법
설치 모드	# rpm -i [설치옵션] 패키지파일+
검색 모드	# rpm -q [질문옵션] [패키지명]
검증 모드	# rpm -V [-y] [--verify [검증옵션]
서명 확인 모드	# rpm --checksig 패키지파일+
삭제 모드	# rpm -e 패키지명+
제작 모드	# rpm -bO [제작옵션] 패키지스펙+

자동 소프트웨어 설치 도구 yum

- rpm을 이용한 패키지 설치
 - RPM 패키지 파일을 다운로드 받아야 한다.
- yum(Yellowdog Updater, Modified)
 - 패키지 파일 다운로드부터 설치까지 모든 것이 자동
 - 패키지 설치, 삭제 및 업데이트를 위한 자동 패키지 설치 도구

기능	사용법
업데이트 목록 확인	# yum check-update
패키지 리스트	# yum list
패키지 업데이트	# yum update 패키지*
패키지 설치	# yum install 패키지+
패키지 삭제	# yum remove 패키지+
패키지 정보 확인	# yum info 패키지
패키지 검색	# yum search 패키지

핵심 개념

- 슈퍼 유저는 root 계정으로 로그인하여 시스템 관리의 모든 측면을 수행할 수 있다.
- init 혹은 systemd 프로세스는 보통 fork/exec를 반복적으로 수행하여 시스템 운영에 필요한 다양한 프로세스들(주로 서비스 데몬 프로세스)을 새로 생성한다.
- init 프로세스는 부팅 시에 런레벨에 따라 설정된 서비스 프로그램들을 로딩하여 실행한다.

