|  |
| --- |
| * Ch\_1 \_ apt 명령어 * Ch\_2 \_ apt 작동방식과 설정파일 * Ch\_3 \_ 파일 압축, 묶기, 시스템설정 * Ch\_4 \_ cron 명령어와 at명령어 * Ch\_5 \_ 네트워크 관련 필수 개념 * Ch\_6 \_ 파이프, 필터, 리디렉션 * Ch\_7\_ 서비스 소켓 * Ch\_8\_ 응급복구, GRUB 부트로더 * 편리한 패키지 설치, apt (1) * apt 명령 * “dpkg” 명령의 패키지 의존성 문제를 완전하게 해결 * 인터넷을 통하여 필요한 파일을 저장소(Repository)에서 자동으로 모두 다운로드해서 설치하는 방식      * apt 기본적인 사용법 * 기본 설치 : apt install 패키지이름   주로 “apt -y install 패키지이름” 으로 사용  “-y”는 사용자의 확인을 모두 “yes”로 간주하고 설치를 진행한다는 옵션   * 패키짐 목록의 업데이트 : apt update * 삭제 : apt remove/purge 패키지 이름 * 사용하지 않은 패키지 제거 : apt autoremove * 내려 받은 파일 제거 : apt clean 또는 apt autoclean * 편리한 패키지 설치, apt (2) * apt-cache * 패키지를 설치하기 전에 패키지에 대한 정보나 의존성 문제를 미리 확인 * 패키지 정보 보기   apt-cache show 패키지이름   * 패키지 의존성 확인   apt-cache depends 패키지이름   * 패키지 역의존성 확인   apt-cache rdepends 패키지이름     * 파일이 없는 상태에서 정보 확인 * apt-cache show 패키지이름      * apt 명령어를 사용하여 패키지 다운로드 * apt -y install 패키지이름 (-y로 동의 생략) * 편리한 패키지 설치, apt (3) * apt 작동 방식 설정 파일 * ‘apt install 패키지이름’ 명령이 작동하는 방식      * 편리한 패키지 설치, apt (4) * 우분투 패키지 저장소 * main : 우분투에서 공식적으로 지원하는 무료(Free) SW * universe : 우분투에서 지원하지 않은 무료 SW * restricted : 우분투에서 공식적으로 지원하는 유료 (Non-Free) SW * multiverse : 우분투에서 지원하지 않은 유료 SW * 저장소가 기록된 파일 * /etd/apt/sources.list 파일 * 형식 : deb 우분투\_저장소\_URL 버전\_코드명 저장소\_종류        * gedit으로 URL주소 비활성화 (주석처리함)      * 패키지를 찾을 수 없다는 오류 * apt작동 흐름은 패키지 목록 파일만 받아온 뒤 y를 입력하여 설치에 필요한 패키지를 받는 형식이라 /etc/apt/sources.list 파일의 url이 없으면 불가        * /ete/apt/sources/list 파일에 url값 추가      * 추가 후 정상 다운로드 되는 모습      * focal - “updates” 추가하면 업데이트된 버전의 패키지를 자동으로 다운로드   (기존의 버전도 다운로드됨)     * apt upgrade 명령어로 업데이트가 필요한 모든 패키지를 업데이트 가능   (권장하지 않음. 필요한 패키지만 찾아서 업데이트 권장)     * ubuntu Software화면 * apt install은 패키지 내용을 알아야 하는 반면 ubuntu Software는 GUI로 편리한 설치 가능 * 파일의 압축과 묶기 (1) * 파일 압축 * 압축파일 확장명은 xz, bz2, gz, zip, Z 등 * xz나 bz2 압축률이 더 좋음 * 파일 압축 관련 명령 * xz: 파일명 xz로 압축을 하거나 풀어준다   예) xz 파일명  xz -d 파일명.xz   * bzip2: 확장명 bz2로 압축을 하거나 풀어준다   예) bzip2 파일명  bzip2 -d 파일명.bz2   * gzip: 확장명 gz으로 압축을 하거나 풀어준다   gzip 파일명  gzip -d 파일명.gz   * zip/unzip: 확장명 zip으로 압축하거나 풀어준다   예) zip 새로생성될파일이름.zip 압축할파일이름  unzip 압축파일이름.zip   * 파일의 압축과 묶기 (2) * 파일 묶기 * 리눅스(유닉스)에서는 ‘파일 압축’과 ‘파일 묶기’는 원칙적으로 별개의 프로그램으로 수행 * 파일 묶기의 명령어는 ‘tar’이며, 묶은 파일의 확장명도 ‘tar’이다 * 파일 묶기 명령(tar) * tar: 확장명 tar로 묶음 파일을 만들어 주거나 묶음을 풀어준다   동작: c(묶기), x(풀기), t(경로확인)  옵션: f(파일), v(과정보이기), j(tar+xz), z(tar+gzip), k(tar+bzip2)   * 사용 예   # tar cvf my.tar /etc/systemd/ > 묶기  # tar cvfj my.tar.xz /etc/systemd/ >묶기 + xz 압축  # tar xvf my.tar > tar 풀기  # tar xvfj my.tar.xz /etc/systemd/ > xz압축 해제 + tar 풀기   * 파일 위치 검색 * find [경로] [옵션] [조건] [action] : 기본 파일 찾기 * [옵션] -name, -user(소유자), -newer(전,후), -perm(허가권), -size(크기) * [action] -print(디폴트), -exec(외부명령 실행) * 사용 예   # find /etc -name “\*.conf”  # find /bin -size +10k -size -100k  # find /home -name “\*.swp” -exec rm {} \;     * which 실행파일이름 : PATH에 설정된 디렉터리만 검색 * whereis 실행파일이름 : 실행 파일, 소서, man페이지 파일까지 검색 * locate 파일이름 : 파일 목록 데이터베이스에서 검색 * 시스템 설정 * 다양한 환경 설정 (gnome-control-center)      * 네트워크 설정 ( nmtui) * 방화벽 설정 (ufw, gufw) * CRON과 AT * cron * 주기적으로 반복되는 일을 자동적으로 실행될 수 있도록 설정 * 관련된 데몬(서비스)은”crond”, 관련 파일은 “/etc/crontab” * /etc/crontab 형식   분 시 일 월 요일 사용자 실행명령  예) 00 05 1 \* \* root cp -r /home /backup   * at * 일회성 작업을 예약 * 사용 예   예약 : # at <시간>  예) # at 3:00am tomorrow > 내일 새벽 3시  # at now + 1 hours > 1 시간 후  at> 프롬프트에 예약 명령어 입력 후 [Enter]  완료되면 [Ctrl] + [D]  확인: # at -l  취소: # atrm <작업번호>     * crontab 파일에 분 시 일 월 요일 사용자 실행명령 설정 * /etc/crontab 파일을 gedit에디터로 열기 * 01 03 15 \* \* root /root/myBackup.sh   분 시 일 월 요일 사용자 실행명령     * myBackup.sh파일을 생성 후 실행 가능으로 권한변경      * 에디터로 myBackup.sh 파일 수정 * gedit myBackup.sh 명령어      * backup 폴더 생성 후 corn 서비스 재시작 (설정 완료) * at 명령어 실습      * rdate , at 설치      * at명령어를 사용하여 일회성 작업 예약 * at 4:00 tomorrow // 내일 새벽 실행 * apt -y upgrade // 시스템 업그레이드 * reboot // 업그레이드 후 재부팅 * Ctrl + D // 마무리 * 네트워크 관련 필수 개념 (1) * TCP/IP * 컴퓨터끼리 네트워크 상으로 의사소통을 하는 “프로토콜” 중 가장 널리 사용되는 프로토콜의 한 종류 * 호스트 이름(Hostname)과 도메인 이름(Domain name) * 호스트 이름은 각각의 컴퓨터에 지정된 이름 * 도메인 이름(또는 도메인 주소)는 hanbit.co.kr과 같은 형식 * IP 주소 * 각 컴퓨터의 랜카드에 부여되는 중복되지 않는 유일한 주소 * 4바이트로 이루어져 있으며, 각 자리는 0~255까지의 숫자 * 예) Server의 IP 주소는 192.168.111.100 * 네트워크 주소 * 같은 네트워크에 속해 있는 공통된 주소(예: 192.1668.111.0) * 네트워크 관련 필수 개념 (2) * 브로드캐스트(broadcast) 주소 * 내부 네트워크의 모든 컴퓨터가 듣게 되는 주소 * 현재 주소의 제일 끝자리를 255로 바꾼 주소(C클래스) * 게이트웨이(Gateway),라우터(Router) * 라우터 = 게이트웨이 * 네트워크 간에 데이터를 전송하는 컴퓨터 또는 장비 * VMware의 게이트웨이 주소는 192.168.111.2로 고정 * 넷마스크(Netmask) & 클래스(Class) * 넷마스크: 네트워크의 규모를 결정(예:255.255.255.0-C클래스) * DNS(Domain Name System) 서버(=네임 서버) 주소 * URL을 해당 컴퓨터의 IP주소로 변환해 주는 서버 * 설정 파일은 /etc/resolv.conf * VMware를 사용하면 VMware가 192.168.111.2번을 게이트웨이 및 DNS 서버로, 192.168.111.254를 DHCP 서버로 설정함 * 네트워크 관련 필수 개념 (3) * 리눅스에서의 네트워크 장치 이름 * 우분투는 랜카드를 ens32 또는 ens33으로 할당함 * 명령 예   # ifconfig ens32 또는 ens33 > 네트워크 설정 정보를 출력  # ifdown ens32 또는 ens33 > 네트워크 장치를 정지  # ifup ens32또는ens33 > 네트워크 장치를 가동   * 중요한 네트워크 관련 명령어 (1) * nm-connection-editor 또는 nmtui * 네트워크와 관련된 대부분의 작업을 이 명령어에서 수행   자동 IP 주소 또는 고정 IP주소 사용 결정  IP주소, 서브넷 마스크, 게이트웨이 정보 입력  DNS 정보 입력  네트워크 카드 드라이버 설정  네트워크 장치(ens32)의 설정   * systemctl < start/stop/restart/status > networking * 네트워크의 설정을 변경한 후에, 변경된 내용을 시스템에 적용시키는 명령어 * 중요한 네트워크 관련 명령어 (2) * ifconfig <장치이름> * 장치의 IP주소 설정 정보를 출력 * nslookup * DNS서버의 작동을 테스트하는 명령어 * ping <IP주소 또는 URL> * 해당 컴퓨터가 네트워크상에서 응답하는지를 테스트하는 간편한 명령어 * 네트워크 설정과 관련된 주요 파일 * 네트워크 기본 정보가 설정된 파일 * X 윈도 모드: Server, Client   ‘/etc/NetworkManager/system-connections/유선 연결1’ 파일   * 텍스트 모드: Server(B)   /etc/netplan/\*.yaml 파일   * /etc/resolv.conf * DNS 서버의 정보 및 호스트 이름이 들어 있는 파일      * /etc/hosts * 현 컴퓨터의 호스트 이름 및 FQDN이 들어 있는 파일 * 네트워크 관련 실습      * nm-connection-editor 명령어로 네트워크 IP 변경 * 192.168.111.100 > 192.168.111.50 변경 후 저장      * 변경된 IP주소 확인 가능 * cat /etc/NetworkManager/system-connections/유선/연결/1.nmconnetction      * nano 에디터로 /etc/resolv.conf 파일 수정 * nameserver 8.8.8.8 은 구글에서 지원하는 DNS 서버   (전세계 어디서도 연결 가능 // 절대 다운되지 않음)   * 재부팅 시 다시 초기화 // 영구적으로 변경 원할 시 DNS 주소를 변경      * 테스트를 위해 namerserver에 아무 숫자 기입      * 네임서버를 통해서 url을 ip로 변경해야 하는데 엉뚱한 ip를 기입하여 접속 불가 * 하지만 해당 화면만으로는 정확한 문제 진단 어려움      * nslookup 명령어로 네임서버 확인가능 * server 8.8.8.8 구글서버로 변경 후 url입력으로 응답확인 가능   (구글 네임서버는 응답을 하므로 네임서버가 고장 난 것을 확인 가능)   * 파이프, 필터, 리디렉션 * 파이프(pipe) * 두 개의 프로그램을 연결해 주는 연결통로의 의미 * “|” 문자를 사용함 * 예) # ls -l /etc | more * 필터(filter) * 필요한 것만 걸러 주는 명령어 * grep, tail, wc, sort, grep, awk, sed 등 * 예) # ps -ef | grep bash * 리디렉션(redirection) * 표준 입출력의 방향을 바꿔 줌 * 예) ls -l > list.txt * soft < list.txt > out.txt * 프로세스, 데몬 (1) * 정의 * 하드디스크에 저장된 실행코드(프로그램)가, 메모리에 로딩되어 활성화된 것 * 포그라운드 프로세스(Foreground Process) * 실행하면 화면에 나타나서 사용자와 상호작용을 하는 프로세스 * 대부분의 응용프로그램 * 백그라운드 프로세스(Background Process) * 실행은 되었지만, 화면에는 나타나지 않고 실행되는 프로세스 * 백신 프로그램, 서버 데몬 등 * 프로세스 번호 * 각각의 프로세스에 할당된 고유번호 * 작업 번호 * 현재 실행되고 있는 백그라운드 프로세스의 순차번호 * 프로세스, 데몬 (2) * 부모 프로세스와 자식 프로세스 * 모든 프로세스는 부모 프로세스를 가지고 있음 * 부모 프로세스를 Kill하면, 자식 프로세스도 자동으로 kill됨 * 프로세스 관련 명령 * ps   현재 프로세스의 상태를 확인하는 명령어  “ps -ef | grep <프로세스 이름>”을 주로 사용함   * kill   프로세스를 강제로 종료하는 명령어  “Kill -9 <프로세스 번호>”는 강제 종료   * pstree   부모 프로세스와 자식 프로세스의 관계를 트리 형태로 보여줌   * 프로세스 실습 * 서비스와 소켓 (1) * 서비스 * 시스템과 독자적으로 구동되어 제공하는 프로세스를 말한다. 예로 웹 서버, DB서버, FTP 서버 등이 있다. * 실행 및 종료는 대개 ‘systemctl start/stop/restart 서비스이름’으로 사용된다. * 서비스의 실행 스크립트 파일은 /lib/systemd/system/ 디렉터리에 ‘서비스이름.service’라는 이름으로 확인할 수 있다. 예로 Cron 서비스는 cron.service 라는 이름의 파일로 존재한다      * 부팅과 동시에 서비스의 자동 실행 여부를 지정할 수 있는데, 터미널에서 systemctl list-unit-files 명령을 실행하면 현재 사용(enabled)과 사용 안 함(disabled)을 확인할 수 있다.      * 서비스와 소켓 (2) * 소켓 * 서비스는 항상 가동되지만, 소켓은 외부에서 특정 서비스를 요청할 경우에 systemd가 구동 시킨다. 그리고 요청이 끝나면 소켓도 종료된다 * 그래서 소켓으로 설정된 서비스를 요청할 때는 처음 연결되는 시간이 앞에서 설명한 서비스에 비교했을 때 약간 더 걸릴 수 있다. 이유는 systemd가 서비스를 새로 구동하는 데 시간이 소요되기 때문이다. 이와 같은 소켓의 대표적인 예로 텔넷 서버를 들 수 있다. * 소켓과 관련된 스크립트 파일은 /lib/systemd/system 디렉터리에 소켓이름.socket 라는 이름으로 존재한다      * 응급 복구 * 시스템이 부팅이 되지 않을 경우에 수행 * root 비밀번호를 분실했을 시      * 재부팅 후 검은색 로딩 화면에서 esc 매우 연타 시 나오는 GRUB 화면      * Ubuntu 선택 후 e를 눌러 편집 화면으로 전환 * 방향키 아래 (↓)로 linux /boot …를 찾은 뒤 end 키로 맨 뒤로 이동 후 init=/bin/bash작성 * 이후 Ctrl + x 로 적용 후 부팅 (접속)      * root 접속 완료 후 패스워드 재설정 시 오류 * 읽기 전용 모드로 마운트 되었기 때문에 변경을 불가능 * 읽기 쓰기 모드로 재 접속 (mount -o remount,rw / 명령어)      * passwd 명령어가 잘 작동되는 모습 (새 패스워드 ‘1234’) * 하지만 이러면 다른 사람들도 GRUB에서 패스워드 변경을 할 수 있는 것이 아닌가?   ↑맞음. 그렇다면 보안적으로 매우 위험한 상황이기 때문에 보통은 GRUB 화면에서도 패스워드를 설정하는 방법을 사용한다   * GRUB 부트로더 (1) * GRUB 부트로더의 특징 * 부트 정보를 사용자가 임의로 변경해 부팅할 수가 있다. 즉, 부트정보가 올바르지 않더라도 수정하여 부팅할 수 있다. * 다른 여러 가지 운영체제와 멀티부팅을 할 수 있다. * 대화형 설정을 제공해줘서, 커널의 경로와 파일 이름만 알면 부팅이 가능하다. * GRUB2의 장점 * 셸 스크립트를 지원함으로써 조건식과 함수를 사용할 수 있다. * 동적 모듈을 로드할 수 있다. * 그래픽 부트 메뉴를 지원하며, 부트 스플래시(boot splash)성능이 개선되었다. * ISO 이미지를 이용해서 바로 부팅할 수 있다. * GRUB 부트로더 (2) * GRUB2 설정 방법 * /boot/grub/grub.cfg 설정파일 (직접 변경하면 안됨) * /etc/default/grub 파일과 /etc/grub.d/ 디렉터리의 파일을 수정한 후에 ‘update-grub’ 명령어를 실행해 설정함. * /etc/default/grub 파일      * GRUB 부트로더 변경 * 부트로더를 변경하는 방법 익히기 * GRUB에 비밀번호를 지정하는 방법 익히기        * grub 파일 편집 * nano 에디터로 /etc/default/grub 파일 열기 * 이후 GRUB\_TIMEOUT을 주석처리 GRUB\_DISTRIBUTOR는 마음대로 수정      * 업데이트 후 재부팅. 정상적으로 변경된 모습      * /etc/grub.d/00\_header 파일을 편집 * pg dn으로 문서 끝으로 이동 이후 새 명령어 추가 * update-grub 이후 재부팅      * GRUB 화면에 username password 입력 창 생성 성공 |