



리눅스마스터 2급 정리본(2023년도 ~ 2020년도 빈출개념 pdf)

<설명>

오타나 틀린 개념이 있을 수 있습니다.

중요도 순서는 아래처럼 표시했습니다.

(3회 이상 빈출은 ☆)

: ☆ > 형광펜 > 굵은글씨

1과목

RAID

오류발생해도 그치 -> RAID 1,5,6,10

(

0은 불가함...)

RAID 사용용량

- RAID-5에서 실제 데이터저장용량은 N-1개(파티티블록1개에 사용)

EX) 하드디스크 8개중 하나를 스파어디스크로 구성(-1개이므로 7개), 나머지를 RAID-5으로 구성(7-1=6개 사용가능)

ex) 디스크3개로 구성했을때 33.3%가 패리티공간으로 사용됨

- RAID-0은 그대로 다 사용가능 EX) 10GB 디스크 4개로 구성시 40GB 사용가능 (**but 복구불가**)
- RAID-1은 미러링(복제)하므로 절반만 사용가능 50%

RAID 관련기술

- 스트라이핑 - 데이터를 여러디스크에 나누어 저장 (RAID0)
- 미러링 - 데이터를 여러디스크에 복제하여 저장 (RAID1)
- 패리티 - 데이터복구를 위한 패리티비트 (RAID 5,6)

LVM

구성순서 : PV → VG → LV

- 물리디스크 1개로 여러 파티션 구성가능, 물리디스크2개로 하나의 파티션도 가능
- 파티션크기확장해도 손실없음(다만 파티션축소시 손실발생)
- 쉽게용량증설이 가능함!!!
- **LV**
사용자가 필요한만큼 할당, 물리적디스크에서 분할
- **PE (Physical Extent)**
LVM구성단위, 일반하드디스크블록

RAID-6는 2개의 패리티블록을 제외해야함 → N-2개

디스크용량(GTPE)

GB < TB < PB < EB

프린터관련 BSD계열 VS systemV계열

BSD계열(ip_)

- lpr: 출력(request)
- lpq: 목록 출력
- lprm: 대기작업 삭제
- lpc: 컨트롤

systemV계열(3가지)

- lp : 출력
- lpstat : 프린터큐의 작업정보확인, 상태확인
- cancel : 작업취소

사운드

- **oss (open sound system)**

사운드를 만들고 캡쳐
, 표준유닉스(POSIX)
프로젝트초기는 Free였으나 사유화

- **ALSA (Advanced Linux Sound Architecture)**
명령어는 alsactl
사운드카드를 자동으로구성

스캐너

- SANE(Scanner Access Now Easy)

rpm 명령

- 의존성 패키지 삭제(**erase**) : rpm **e** nmap --nodeps
- 패키지 설치 or 업그레이드 : **u**
- **qf** : 원하는 형태로 출력하기
- **qa** : 설치된 패키지
- **qi** : 패키지 상세정보

yum 명령

- 패키지 설치 : yum **install** 패키지 **★**
- **info** : 패키지관련 정보확인
- 패키지 제거 : yum **remove** 패키지

tar 파일별

파일별 확장자

- bz2 → j_
- gz → z_
- xz → J_

압축률 좋은 순서 : xz > bz2 > gz

compress명령어 → Z 파일이 생성

옵션

- **x** : 압축해제(extract)
- **t** : 파일내용확인(list)
- **c** : create만들기
- **r** : append 묶기
- **tvf** : 파일의 내용확인하기 **★**

make **clean** 명령어

컴파일과정에서 생성된 **중간파일들** 삭제

소스파일 설치단계

configure → make → make install

makefile생성 / 컴파일 / 설치

계열별 패키지관리도구 ★

- 데비안계열 - `apt-get` (`apt`), `aptitude`, `dpkg`
- SUSE(수세)계열 - `zypper`, `YaST`
- 레드햇계열 - `rpm`, `yum`, `dnf`
- 온라인 패키지 (인터넷으로 원격저장소 접속가능) - `apt-get` (`apt`), `dnf`, `zypper`, `pacman`, `yum`
- 저수준 패키지 (로컬에서 직접수행) - `dpkg`, `rpm`, `alien` (rpm을 deb로 변환)

vi 편집기 ★

ex 명령모드

- ! → 강제!!!!
- q → 변경사항이 없는경우에만 종료
- set number → 행번호 표시
- set ai → 바로윗줄에 커서위치
- set nu → 자동으로 행 번호 표시 ↔ set nonu → 행번호 삭제
`set list` → 개행문자, TAB문자

입력모드

- x → 문자삭제
- dd → 라인삭제

문자열 검색과 파일등....

/ : 아래방향으로 검색

n : 같은방향으로 다음문자열검색 next

u : undo, 직전에 삭제한 줄 복원

+ :커서를 파일 마지막줄에

\$: 마지막줄로 이동

.exrc 파일 :환경설정한 내용을 계속사용하기위해
키보드 hjkL ← 아래 → 위 순서임

.vimrc 파일 : 설정파일로 행번호표시등..

`q` : 변경내용 저장안하고 종료 quit

`e` : 명령모드 → 편집모드로 전환 exit

문자열치환 ★

- 줄의 시작 -> ^
- 줄의 끝 → \$
- 특수문자 → \(백슬래시) 아래 단어시작끝과 같이씀
- 단어의 시작 → <
- 단어의 끝 → >

ex)

`linux`로 끝나는 줄의 마지막에 마침표를 덧붙이도록 치환 -> :% s/linux\$/linux./

ex2) 단어의 시작이 `linux`인 경우 전부 Linux로 치환 → :% s/\<linux/Linux/g

편집기 ★

등장순서 : vi → pico , nano → gedit

- **vi편집기** ★ : 빌조이 ★, 모드형
- **emacs편집기** : 리처드스톨만, 매크로, 텍스트교정편집기
(종료하기는 `ctrl+x` 후에 `ctrl+c`)
- **vim 편집기** : 브람(vim브-빈브격차) 무레나르, 다양한색상으로 가시성
- pico랑 nano(사용법동일)
 - pico는 워싱턴대학에서 만든 유닉스용, 최근에는 pico의 복제본인 nano가 사용됨
 - **nano - GNU**

현재커서 줄의 처음a으로 이동 <`ctrl+a`>

마지막(end)으로 이동 <`ctrl+e`>

파일저장 <`ctrl+o`>

프로그램종료 <`ctrl+x`>

X원도

편집기

x원도에서만 사용가능 -> `kwrite`, `gedit`

(nano, pico, vim은 터미널(콘솔) 가능)

kill %2

작업번호가 2번인 백그라운드프로세스 종료

-> kill은 특정프로세스를 종료시킴

ex) PID가 513,514,515 프로세스 종료 -> kill 513,514,515

KILLALL

인자로가진 이름의 프로세스를 모두 종료

(

`nice` 와 함께 **프로세스명**을 인자값으로 사용하는)

KILL VS KILLALL

renice VS nice

PID VS 프로세스명

ps명령 ★

실행중인 프로세스정보를 출력(PID,TTY,TIME,CMD)

백그라운드로 실행중인 데몬 확인

로그인한 후에 이용중인 셀 확인 ★

`z` : 작업은 종료되었으나 회수되지 않아 메모리를 차지한 상태

`1` : 프로세스 우선순위값 확인

`aux` : 실행중인 모든 프로세서 확인

NI 값

범위: (높은우선순위)-20 ~ 19

nice VS renice

- `nice` → 프로세스명

renice → PID

ex) pid가 1222인 bash프로세스의 우선순위가 0이다. ni값을 10으로 변경하자 → `nice -10 bash`

ex) pid가 1222인 우선순위가 0이다. ni값을 10으로 변경하기 → `renice 10 1222`

ex) `nice bash` → bash기본ni는 0, 따로 ni값 지정안하면 기본 10으로 설정됨 -> bash가 10으로 되면서 우선순위가 낮아짐

cron

분 시 일 월 요일

- **요일 (0,7 일요일)**
1:월 ~ 6:토

포어그라운드와 백그라운드

- **포어그라운드 종료** `<ctrl+c>`
포어그라운드 일시중지
`<ctrl+z>`
- **실행한 백그라운드 확인** `jobs` ★
- **포어→백으로 전환** `<Ctrl+z>`
- **백그라운드로 실행** &

데몬 프로세스 실행방식 standalone vs inetd

standalone - 메모리에 독자적 상주

inetd - 그때그때, 메모리관리에 효율적

(데몬daemon은 주기적이고 지속적인 서비스요청을 처리하기위해 실행되는 백그라운드 프로세스임)

시그널번호

kill 명령어로 확인가능 (kill%2 이런식으로 pid를 씀)

SIGHUP - 1

SIGINT - 2, 종료, `<CTRL+C>`

SIGQUIT - 3

SIGKILL - 9, 프로세스 즉시 강제 종료

SIGTERM -

15, kill

SIGSTOP - 20, 대기, `<CTRL+Z>`

SIGQUIT -

`<CTRL + \>`

프로세스 호출방법

- fork : 복제
- exec : 새로운거로 대체

→ cent os 7 리눅스에서는 `systemd`라는 프로세스를 발생시키고 pid는 1이다.

cent os 6 에서는

`init` 였음..

전역과 지역파일

구분

- 전역 : /etc
- 지역 : ~/

파일별

- `bash_profile` : 환경변수
- `bashrc` : alias, 함수
- `profile` : 시작관련 프로그램

\$echo "user"

사용자이름을 출력

-> [
ihduser@ihd ~] 라면 ihduser를 출력

기호 명령어

`!!` : 가장 최근 명령 실행(바로직전)

`!0` : 0번째 명령어 실행

`!?al` : 최근 명령중 al문자열 포함한 명령 실행

환경변수와 셀변수

환경변수파 - printenv, env(환경변수만 출력)

둘다 포용 - set (환경변수와 셀변수 출력) ↔

`unset` (제거, 해제)

셀변수 관련 명령

`set` - 셀변수를 전부 확인, 설정 ↔ `unset` 삭제

`chsh` - 셀 변경

`env` - 현재셀의 환경변수 출력

파일

로그인당시 VS 현재

로그인시 정보 : /etc/passwd (명령은 `echo $SHELL`) ★ (로그인셀 변경시에도 저장됨)

로그인 후 /

현재 사용가능한 정보 : /etc/shells (명령은 `ps`) ★★

현재마운트정보 : /etc/mtab

실시간마운트정보 : /etc/mounts

부팅시마운트할거, 자동지정해두기 : /etc/fstab (필드별로 문제나옴)

(userquota는 설정안함!!!!)

- 1번째 필드 - UUID, LABEL, 장치파일명 가능 - 마운트할 장치나 파일이름 (**마운트포인트안됨**)
- 2번째 필드 - 마운트경로, **마운트포인트**
- 3번째 필드 - 시스템유형
- 4번째 필드 - 마운트 옵션(noexec)
- 5번째 필드 - 백업시 포함여부
- 6번째 필드 - 점검시 순서

chsh

- **l** : 변경가능한 셸의 목록정보확인 (list) / 사용가능한
- **s** : shell, 변경 , 셸경로 붙음 (chsh -s /bin/csh)
- **u** : help 도움말

셸 종류 ☆

순서: sh → csh → tcsh → bash

- **csh** : 버클리 빌조이, 히스토리/alias/작업제어 유용한기능
- **bash** : GNU프로젝트, 다양한운영체제
- **bourne shell** : 가장 오래된
- **dash** : /bin/sh를 가장 작게 구현 , POSIX와 호환
- **tcsh** : c셸(csh)과 명령행 완성기능이 통합

디스크 명령어 (d_)

du : 디스크 사용량(use), 사용자 그리고 디렉터리

df : 디스크

남은(avail) , 마운트된 사용량, 파티션

fdisk

w : 변경사항 저장하고 종료

q : 변경사항 저장안하고 종료

d : 파티션삭제

l : 파티션테이블 정보 확인

mkfs.xfs

xfs 파일시스템을 생성

명령어관련(blkid=uuid)

- **blkid 명령어** : uuid와 파일시스템정보확인 ★(uuid-blkid 세트로!!!)
(리눅스에서 파티션생성시 고유한 uuid가 부여되고, uuid를 확인할때는 blkid를 사용)
- **lsblk 명령어** : 연결된 저장장치 정보확인가능(rm이동식여부)

quota 명령어들

`repquota` : 쿼터 확인 (soft와 hard 확인가능)

`edquota` : 쿼터 설정

`quota` : 설정정보를 확인

`xfs_quota` : 디스크쿼터 설정(사용량제한)

umask

파일 : 666

ex) umask가 0022인 경우 파일의 허가권은 $666 - 22 = 644$

-/rw-/r--/r--

파일/421/421/421

chmod (허가권 관련)

_ / _ / _ / _

특수비트 소유자u 그룹g 태o

`R` : 하위까지 포함~~

파일이나 디렉터리의 허가권을 변경함

특수비트

- setuid : 소유자 U = 4
- setgid : 그룹 G = 2
- stickybit : 본인이 생성한 파일만 삭제가능하게 = 1

(파일에만 가능!! 디렉터리안됨) ex) `chmod g+t /project` 의 경우 안됨!!!

ex) 삭제는 본인이 생성한 파일만 가능하도록 설정하고(stickybit 1), 파일생성시 자동으로 그룹소유권이 부여되도록설정(setgid 2) => 특수비트 $1+2=3$

->

`chmod 3770 /project`

ex2)

`chmod g+r lin.txt`

파일 허가권

- /rw-/r--/r--
파일임/소유자/그룹/타
ex) chmod g=r lin.txt → 그룹g에 read권한부여
- 첫번째필드(-는 파일, d는 디렉터리, l은 심볼릭링크)
- r(read) 읽기
w(write) 수정
x(execute) 실행

chgrp vs chown (소유권 관련)

그룹만 vs 파일소유자+그룹 같이 변경

ls 명령어 (소유권 확인용) ★

파일이나 디렉터리 소유자 확인

특수권한(setuid, setgid등) 확인

소스설치방법(Mysql ↔ PHP)

cmake 선택 : MySQL, KDE, LMMS

(cmake는 Make의 대체프로그램!!)

PHP는 cmake 아님!!

명령어 관련

nohup 명령어

터미널이 닫혀도 백그라운드로 전환해서 계속 작업하게해줌

top 명령어 ★

실시간 모니터링 (PID, USER, PR, NI값, CPU사용량)

로그인 당시 vs 현재

로그인 후에 입력한 명령들 -> .bash_history 파일

XSANE (X based interface for the SANE)

X윈도기반, GTK+, 스캐너

스캐너를 사용하기위해 설치해야함

dpfg 명령어

r : 패키지 제거 ch

셀의 기능

명령행 완성 기능: 이름이 긴 파일명 입력시 앞글자만 입력하고 TAB키 누르면 나머지부분 불러옴

명령행 히스토리 기능: 기존실행한 명령들을 위/아래방향키를 사용해서 검색편집가능

앨리어스 기능 : hd 입력하면 홈디렉터리안의 내용을 확인할 수있음 (별칭)

파일별 점검 명령어

.ext4 → fsck명령

.xfs → xfs_repair 명령 ★

mount 명령어

ex) 파일시스템이 ext4인 /dev/sdb1 파티션을 /data 디렉터리로 마운트하자

→ mount -t ext4 /dev/sdb1 /data

ex) 관련정보 변경후 다시 마운트하는 경우

→ mount -o remount /home

특수권한이 설정된!!! (Set-UID나 Set-GID같은)

/usr/bin/passwd ★

프린터 관련

CUPS 프로그램(Common Unix Print system)

리눅스에서 프린터 서버로 사용하기위해 설치해야함

애플, 공개 프린팅 시스템

유닉스를 프린트서버로

system-config-printer 명령어 ★

CentOS7에서 X윈도 기반으로 프린터 설정시 실행명령어

IPP

인터넷상에서 원격으로 인쇄하기위한 프로토콜

CIFS ↔ SAMBA

윈도우 ↔ 리눅스, 유닉스

PS1

로그인 프롬포트 변경

PS2

2차 프롬포트 변경할때 쓰는 환경변수

/etc/shadow

일반사용자가 내용 못 봄!!!

NFS

한대의 리눅스 디렉터리 만들고 나머지사용자들이 손쉽게 접근!!!!

저널링

ext3부터!!, ext4, xfs

2과목

클라우드 서비스

이용자 많은 순서(**IPS**) : IaaS > PaaS > SaaS

IaaS	서버, 스토리지같은 인프라
PaaS	플랫폼(Platform)
SaaS	응용소프트웨어(Software)

빅데이터관련

- R : 데이터분석기술, 분석된 데이터의미와 가치를 시각적으로 표현
- Hadoop : 파일시스템, 대량자료처리가능, 분산

가상화관련

- Xen : 반가상화
- KVM : 전가상화, 레드햇사, Qumranet, x86
- VirtualBox

VMware의 파일은 **VMDK**

클러스터

고가용성 클러스터(HA) ★

다수의 웹서버 운영, 부하를 분산

주노드(primary Node), 백업노드(backup Node)

-> 이상시 백업노드가 이어받음

베어울프 클러스터 = HPC(high performance computing) 클러스터

고성능의 계산능력, 주로 **과학계산**용, 슈퍼컴퓨터 시스템구축

파일정리

/etc/resolv.conf 파일

DNS서버주소!!!

도메인 자동 등록 호출

nameserver 주소 ★

/etc/hosts

```
192.168.56.102      www.ihd.or.kr
```

IP주소와 호스트명 매팅, **DNS**기능

가상도메인설정, IP주소변환

네트워크문제해결(올바른 IP로 매팅되는가)

/etc/services

포트번호확인

/etc/protocols

사용 가능한 프로토콜

/etc/sysconfig/network-scripts

네트워크 인터페이스 환경설정 저장

/etc/sysconfig/network (기본정보)

네트워크 사용 유무, 호스트명, 게이트웨이 주소 등

명령어 정리

- `mii-tool`
네트워크 인터페이스 상태
물리적연결여부
- `netstat`
네트워크연결상태 SYN Flooding 탐지,
`-r` 라우팅테이블, 게이트웨이도 가능
- `route`
라우팅테이블, 게이트웨이주소 확인
- `ethtool`
이더넷카드 출력or변경, 물리적 케이블상태
- `ip`
라우팅테이블

IP주소

클래스별

- 루프백(Loopback) 네트워크는 A클래스 (127.0.0.1)
- C클래스의 사설ip 주소의 네트워크는 256개

B클래스 : 128~

C클래스 : 192~

D클래스 : 224~

네트워크(서브넷)과 브로드캐스트주소

EX) IP주소 및 서브넷마스크값이 아래와 같음

`192.168.5.189/26`

1. 서브넷마스크가 26 -> 8/8/8/8중에 $24+2=26$

즉 네번째가 1100/0000

192 -> 1100/0000 (0이 6개)

2. 호스트주소 $\$2^6\$ = 64$ 개

3. 서브 네트워크 나누기

- 0~63

- 64~127

- 128~191 ->189는 여기속함

첫번째 128은 네트워크/서브넷용(

내세에는..) 마지막 191는 브로드캐스트용

첫번째+1인 129 나 마지막-1인 190은 게이트웨이용

FTP 관련(21번)

20번 포트로 데이터전송하고 21번 포트로 제어

로컬을 서버에 업로드 - `put`

서버파일을 다운로드 -

`get`, `recv`

파일진행상태를 # 기호로 - `hash`

active모드랑 passive모드

익명계정가능

FTP사용하려면 FTP서버 필수!!

원격지 관련

EX) A시스템에 있는 프로그램을 원격지에 있는 B시스템에 전송해서 실행하도록 했음

-> A시스템은 x클라이언트, 환경변수 DISPLAY를 변경

-> B시스템은 X서버가 되고, xhost 명령으로 제어(호스트에 대해 x서버접근을 제한or허용)

원격지 ip주소서버에서 계정변경

`ssh -l kaitman @192.168.5.13(ip주소)`

이때

`-l` 옵션은 개인키파일지정

telnet관련

원격서버에 접속, 아이디와패스워드사용, 보안상위험으로 사용x

계정변경시

`telnet -l kaitman(계정id) 192.168.5.13(서버ip)`

텔넷서버로 사용시

`telnet-server` 패키지를 설치해야함

서비스 활성화 확인

`telnet www.kait.or.kr 443`

(↔ ssh명령어는 원격서버용)

메일관련(전자우편)

메일교환가능 - **SMTP**

수신만 가능 -

IMAP, **POP3**

SAMBA ★

리눅스와 윈도우간의 연결 ★

SMB 프로토콜쓰다가 **CIFS** 프로토콜로

국제기구 ★

ICANN : IP주소, 도메인(DNS)

→ 초기에는 IANA에서 관리했다가 ICANN으로 변경

IEEE : LAN 표준(e가 3개나 있으니까 표준,,)

ISO : OSI

E

AI : LAN케이블

EIA : 전자산업협회, RS-232

OSI 7계층 ★

번호	계층이름	설명	사용프로토콜	사용장치	데이터단위
1	물리계층			Hub , Repeater(허리)	bit
2	데이터링크계층			Bridge, Switch (부드럽게스위치)	frame
3	네트워크계층	통신경로선택(라우팅)	IP(IPv4, IPv6) ICMP, IGMP, BGP	Router(라우팅!) (그다음많음!!)	
4	전송계층		UDP, TCP	Gateway(게이)(가장많은지원!)	segment
5	세션계층	동기화유지 데이터전송순서			data
6	표현계층	번역, 압축, 암호화			
7	응용계층화	네트워크제공			

ARP - MAC ★

IP주소랑 MAC 주소 매칭

MAC주소 확인시 사용하는 명령어이기도..

프로그램

이미지(에고, 김)

EOG(뷰어용), Gimp(사진 편집), ImageMagick

동영상

totem - 동영상재생

X윈도

TCP번호는 6000

클라이언트/서버구조, X프로토콜 사용

X클라이언트는 X윈도에서 동작하는 응용프로그램임

파일 및 환경변수, 라이브러리

- .Xauthority 파일
관련키정보저장
- DISPLAY환경변수
X클라이언트를 원격지로 전송할때 변경

EX) echo \$DISPLAY해서 :0 인 경우 → X클라이언트 실행시 로컬시스템 첫번째 X서버에서 실행됨

명령

EX) X서버에서 IP주소가 192.168.12.22번인 X클라이언트를 허가하는 명령 xhost 192.168.12.22
(

- 는 접근차단, + 는 모든접근허용)

EX) 해상도변경시

xrandr

구성요소

- 디스플레이매니저 : 사용자로그인 세션관리
- 데스크톱환경 : GUI환경(환경!)
- 윈도매니저 : 윈도배치(윈도!)
- 유저인터페이스 : 컴퓨터-사람간 상호작용

X윈도 부팅시 ☆

- set-default : 설정
- get-default : 확인
- graphical : x윈도
- multi-user : 텍스트모드
→ 여기에서 x윈도 실행은

startx

- user.target: 모드 부팅설정(GUI나 텍스트,,)
- user.service: 사용자별 프로그램, 데몬..

- `runlevel3.target`: 텍스트기반
- `runlevel5.target`: x윈도

XFree86

x윈도프로젝트, IBM 호환

xauth vs xhost

키값 vs 호스트명, ip주소

윈도매니저 종류 (XFce만 외톨이)

- `x-window` : XFce (외톨이) ★
- 윈도매니저 : Metacity, Enlightenment, Window Maker

라이브러리정리

- KDE : x윈도, 오픈데스크톱, QT
- XCB : Xlib대체, x서버와 대화, 저수준
- GNOME : GTK+, LGPL, nautilus
- GNOME2: metacity
- GNOME3: mutter

LibreOffice 관련

- LibreOffice Impress : ppt, 파워포인트
- LibreOffice Calc : 계산, 엑셀, 스프레드시트
- LibreOffice Writer : 워드

LAN구성

- 버스(Bus)형 : 하나의 회선에 여러개 연결, 모든 장치 동일하게 경쟁, 한번에 한개만 전송
- 스타(Star)형 : 중앙제어기, Point-to-Point방식, 중앙고장나면 전체가 중단...
- 링(Ring)형 : 전송상 충돌없음, 숫자늘어도 성능저하 적고 노드추가는 어려움. 전체 네트워크 중단될 수 있음..
- 망(Mesh)형 : 장애발생시에도 영향적음 BUT 설치비용과 운영이 힘들고 고장지점 찾기 어려움

xauth list \$DISPLAY ★

COOKIE와 쿠키값이 나옴

인증키값 확인

Eternet(이더넷)

LAN을 위해 개발됨

MAC주소기반, 48비트

CSMA/CD방식(다중환경의 충돌탐지)

회선교환 VS 패킷교환

고정된 경로 VS 동적, 오버헤드비트, 무제한가능, 우선순위부여

FDDI

광섬유케이블, 최대 100Mbps, LAN전송방식
x3커미티, 토큰패싱방법

웹브라우저 정리

- lynx : 텍스트기반 (x원도없이도가능)
- chrome, opera, safrai : 모두 GUI (x원도)

chrome : 구글개발, 초기에는 웹킷 레이아웃 엔진, 현재는 포크된 브링크

WWW

하이퍼텍스트방식의 정보검색 시스템
웹서버에 접근하기위해 URL 사용

ip 명령어

ip : 이더넷카드의 mac주소확인
ip route show : 게이트웨이 주소 정보 출력

netstat과 ss 세트

네트워크상태정보확인가능 (SYN Flooding 탐지)
운영중인 서버의 특정포트에 접속해서 연결정보가능 (ESTABLISHED) 정보확인가능
이때 ss에서 -a 소켓의 모든 정보를 출력함

netstat 상태값

ESTABLISHED - 3Way-Handshaking이 완료되고 서버-클라우드 연결

ssh

패킷 암호화, 안전한 원격로그인, 원격 셸 가능

scp, sftp는 ssh기반(ssh, scp, sftp 3개 세트로 알아두기)
(nfs는 아님!!!!)

p : 변경된 포트번호로 접속

```
ssh ip주소 -p 포트번호
```

포트번호

FTP	21
sftp ssh	22
Telnet	23
SMTP	25
DNS	53
HTTP	80
POP3	110
IMAP	143
SNMP	161
HTTPS	443
SMB	445
MYSQL	3306

TCP 프로토콜 패킷

3-way handshaking을 수행 (SYN → SYN/ACK → ACK)

UDP 프로토콜 -DNS

DNS서비스

수신여부x 오류정정x

NFS

파일 공유 디렉터리 생성, 나머지사용자들이 쉽게 접근

네트워크 종류

LAN < MAN < WAN

Local / Metropolitan / Wide

프로그램 정리

Ansible

여러운영체제에서 사용가능, 공개프로그램

15년도 레드햇 인수

Docker

프로세스 격리, 컨테이너

경량화!!!!!!

하이퍼바이저나 게스트운영체제 안쓸!!!!

Kubernetes

오픈소스, docker같은 컨테이너관리, CNCF에서 개발

Arduino

단일보드, 마이크로컨트롤러

하드웨어와 소프트웨어 결합

라즈베리파이

영국잉글랜드, 신용카드크기의 싱글보드컴퓨터

GENIVI

자동차용 IVI(In-Vehicle Infotainment)

IRC

실시간 채팅, Xchat

Tizen

인텔+삼성

IPv6

주소표현단위 128bit

흐름제어, 호스트주소자동설정, 인증과보안

(헤더단순함!!!!)

(클래스확장 아님!!!!)

Ethernet

MAC기반 데이터 주고받기

`enp0s3` : 이더넷카드장착시 나타나는 장치명

ICMP프로토콜

오류보고, 피드백을 보고

webOS

리눅스의 모바일 운영체제, Palm OS계승, 현재 LG전자

리눅스커널 운영체제

webOS, **GENIVI**, **Tizen**

(QNX는 아님!!!!!!)

Evince ★

PDF 문서 확인

파이어폭스

모질라, 게코

Apache OpenOffice

oocalc - 스프레드시트

oowriter - 워드프로세서

ooimpress - 프레젠테이션

oodraw - 벡터그래픽, 다이어그램

Rocky Linux

CentOS가 종료함에 따라 탄생

최상위 도메인

com, net, kr

출처

무한 우주에 순간의 빛일지라도 : 네이버 블로그

당신의 모든 기록을 담는 공간

 https://blog.naver.com/dan9_9



다들 꼭 합격하세요 ☺