**섹션 09.인터넷, 네트워크 그리고 서버**

**01.인터넷1**

internet

network

server

인터넷의 원리

client -> request -> server

client <- respose <- server

서버에 접근하는 방법.

1)google.com과 같은것을 domain name 이라고 함.

2)ip(internet protocol) address로도 접속 가능함.

두 가지 주소체계의 관계

ip adress를 기억하는게 어렵기 때문에 domain name을 이용하는 것이다.

DNS Server는 이세상 모든 도메인이 어떤 ip인지 알고있음.

우리가 주소창에 domain을 입력하면 DNS로 연결되는것이다. (과정은 비밀)

**02.인터넷2**

#ip addr : inet이라는문자 뒤에 컴퓨터의 ip주소가 나옴.

이 ip addr로 접속시 컴퓨터를 방문할 수 있다.

https://ipinfo.io/ip

위 사이트에 접속시 접속자 ip를 알려줌

*curl은 "Client URL"의 줄임말로, 데이터를 주고받기 위해 사용하는 명령줄 도구*

위사이트도 curl의 일종.

ip addr 은 실제 컴퓨터의 ip addr이고,

curl은 자신에게 접속한 그 접속의 결과적 ip이다.

둘은 같을수도 다를수도 있음

많은경우에 두가지 방식이 다름.

집에서 컴퓨터를 쓸 때는 인터넷, 통신사와 계약을 맺음. 통신사가 집에 회선을 넣어주는데 인터넷을 사용하기 위해선 컴퓨터와 회선이 연결되어야한다. 가격 절감을 위해 공유기(Router)를 이용하여 여러기기가 하나의 회선을 사용한다. ip는 라우터가 갖게된다. private IP라는 개념이 등장하는데, 동일 회선을 사용하는 기기마다 ip의 일부만 달라지게 된다. router가 가지는건 외부에 공개되는 public address(IP)공인 IP, 기기가 가지는것은 private address(사설 IP, 사설 주소)라고 한다. curl을 통해 알아내는건 public ip, ip addr 명령어는 각 기기의 ip를 알아낸다. 라우터에 묶여있는 컴퓨터는 서버로 사용할 수 없음.

라우터에 묶여있는 컴퓨터를 서버로 사용하는 법

1) 같은 wifi를 사용한다.

+등등..

**03.apache 웹서버 1- intro**

web server를 리눅스에 설치하는 방법

internet은 server와 client의 request와 response로 이루어져 있다.

server는 request를 분석한다. 서버에 연결된 HDD,SDD와 같은 저장장치 에서 파일을 읽어서 가져온 다음 response 한다.

예시

web browser(client) : firefox, internet explore, chrome

web server : apache, nginx, IS

web browser를 설치하면 쓸 수 있듯이 web server도 설치하면 사용가능함

**04.apache 웹서버 2-install**

ex)apache ubuntu 등을 구글링 하여 명령어를 알아내서 아파치 설치

#sudo service apache2 start : 서버 켜짐

#sudo service apache2 stop : 서버꺼짐

#htop 을 통해 apache가 켜졌는지 확인한다.

Filter> apache

셸에서 웹브라우징 가능한 프로그램들이 있음

그 중 하나 elinks

#sudo ap-get install elinks

#elinks

#google.com > 구글이 켜짐.

#q키를 누르면 빠져나가짐

셸 환경에선 graphical 한 것의 한계가 있음

#ip addr을 하여 본인의 ip를 알아내기

#elinks https://ip addr결과/ : ubuntu apache2 라고 뜨면 성공적 접속.

자신의 웹주소를 알아내는법

->로컬 호스트 localhost

한 대의 컴퓨터 안에서 browser와 server를 사용중임.

브라우저 입장에선 서버가 같은 컴퓨터임. 브라우저가 자신이 설치된 서버컴퓨터에 존재하는 웹서버에 접속한다. 라고 하는 주소가 로컬호스트이다.

localhost는 domain 이고, ip와 동일하다.

*로컬호스트(localhost)는 컴퓨터 네트워크에서 사용되는 용어로, 현재 자신이 사용하고 있는 컴퓨터를 의미합니다.*

127.0.0.1은 자기자신 컴퓨터를 가르키는 ip임.

#elinks 127.0.0.1 : 자신의 컴퓨터 웹서버에 접속됨.

#elinks https://localhost : 로컬호스트의 도메인으로, 동일하게 접속가능

**05.apache 웹서버 3-configuration (어렵넹)**

웹서버의 스토리지(HDD나 SDD)

저장장치 어딘가에 저장되어있는 파일을 읽는다.

컴퓨터가 어떻게 작동될 것인가 : etc에 저장

#nano apache2.conf : 아파치 2 파일열기.

그곳에서

sites-enabled: \*/.conf : 라는 파일 있음. : 그 디렉토리에 있는 모든 확장자가 conf 파일을 읽는다는 뜻. 이 파일로 드렁가면 000-default.conf 라는 파일이 있음. 링크인데, 이 실제내용은 이 안에 있다는 뜻. 이 안으로 들어가면

Document Root가 있는데 그곳으로 들어가보자.

/var/www/html 디렉토리 (언제든 다른곳으로 옮길 수 있음)

사용자가 요청한 내용을 찾는 최상위 디렉토리를 document root 라고함.

현재의 디렉토리에

#sudo nano index.html 후 간단한 html파일을 만든다.

#ls-al 파일 생성을 확인

#elinks https:/127.0.0.1/index.html :파일이 생긴걸 확인할 수 있음.

**06.apache 웹서버 4-log**

apache라는 하나의 구체적 사례를 통해 전체적인것을 이해하게 돕기

저번시간 ) /var/www/html 이란 디렉토리에 있는 파일을 웹서버가 찾아서 사용자의 요청에 대해 응대함. 이유는 /etc/apache2/sites-ebabled/000-default.conf 라고 적혀있기 때문이다. document root를 바꾸면 다른곳에 설정할 수 있음.

>대부분의 서버는 설정파일 변경을 통해 동작방식을 변경 가능하다.

#access.log : 로그 디렉토리에 파일과 액세스 로그파일 등 정보가 기록됨.

#/var/log/apache2/

누군가 웹서버에 접속할 때 마다 로그가 마지막에 남음

#tail 명령어를 통해서 출력하자.

#tail -f 를 통해 실시간 출력가능.

#error.log : 아파치 웹서버의 특정 로그가 있을 때 확인하면 됨.

**07.원격제어 ssh1**

SSH (중요함)

리눅스가 조망받는 이유는 많은 서버가 사용하고 있기 때문이다.

client(자신의 컴퓨터)

server(유닉스 계열 컴퓨터, 리눅스가 서리된 컴퓨터)

client를 ssh를 이용하여 server를 원격제어 해야한다.

ssh서버가 request, response를 한다.

유닉스 계열엔 기본적으로 ssh가 설치되어있음.

**08.원격제어 ssh2**

#purge : remove 보다 강력한 삭제, 설치환경 등도 모두 날아감

#sudo apt-get install openssh-server openssh-client : ssh와 관련된것들 제공하는 ssh서버 설치, 다른컴퓨터로 원격제어시 openssh-client 필요함.

#sudo service ssh start

#sudo ps aux | grep ssh : sshd라고 되어있으면 성공적 실행

이 서버는 계속 실행되고 있다가 ssh client의 접속을 기다린다.

#ssh emfpdlzj@165.~... : open client 실행

ssh를 통해서 원격제어가 가능해진다.

**09. 포트(port)1 - 포트란 무엇인가?**

naver.com 이나

naver.com:80 이나 모두 접속은 가능하다.

여기서 뒤에 붙은 :80이라는 숫자가 바로 포트이다.

#ssh -p 22 emfpdlzj@162.~ : 22포트로 접속

모든컴퓨터에는 포트가 있다.

포트는 약 6만5천개의 구멍을 가짐.

1024번까지는 아주 유명한 프로그램들이 존재함.

#nano /etc/ssh/sshd\_config

여기서 포트넘버를 변경해주면 해당 포트로 웹서버를 대기시킬 수 있음

#sudo service ssh restart : 재시작을 통해 변경사항 적용.

*SSH(Secure Shell)는 네트워크 상에서 보안적으로 안전하게 데이터를 교환하기 위해 사용하는 프로토콜입니다. SSH는 암호화된 통신을 제공하며, 주로 원격 서버에 접속하여 명령어를 실행하거나 파일을 전송하는 데 사용됩니다. SSH와 관련된 포트는 중요한 개념 중 하나입니다.*

*기본 포트*

*SSH의 기본 포트 번호는 22번입니다. SSH 클라이언트가 서버에 연결할 때, 기본적으로 이 포트를 통해 연결을 시도합니다. 예를 들어, 다음과 같이 SSH를 통해 원격 서버에 접속할 수 있습니다:*

*#ssh user@remote\_server*

*다른 포트 사용*

*보안 이유로 또는 다른 이유로 SSH 서버의 포트를 기본 포트 22번이 아닌 다른 포트로 변경할 수 있습니다. 예를 들어, 포트 2222번을 사용하도록 설정한 경우, SSH 클라이언트는 해당 포트를 지정하여 접속해야 합니다*

*ssh -p 2222 user@remote\_server*

**10. 포트(port) - 포트포워딩 소개**

어려워서 안들음

port forwarding

#isp : 인터넷 제공자, ex 통신사

공유기, router를 이용

**11. 포트(port) -포트포워딩 실습**

어려워서 안들음