

Date → 23/22/22

Date
Page

Session - 14

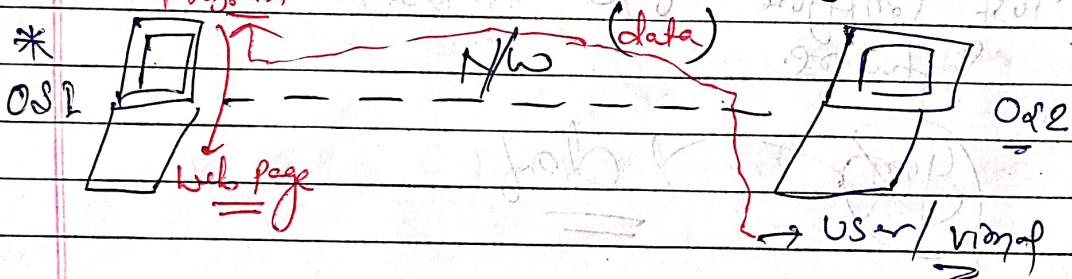
- * ~~If pod A → Kubernetes controller for pod~~
- * If something failed how they can fix it and re-launch it.?

* Label →

- ① Why we need labels.
- ② How to use labels.

* ~~Frontend - Backend~~

- * If a system give some kind of services.



* Virtual hosts using N/w from (OS2) key send data send to OS2.

* (OS2) we have program that contains web page, we can't go to that system directly we can tell OS(2) you have any way through N/w I can get you info & see to Content.

* we can't a program in (OS2), that tells somebody what somebody is saying, i.e. server.

* The program / OS2 that provides some kind of services i.e. server, & the system who take the services those are Client.

Servers - Client Model.

* Viral → one kind of servers providing teaching services.

Client → we are clients.

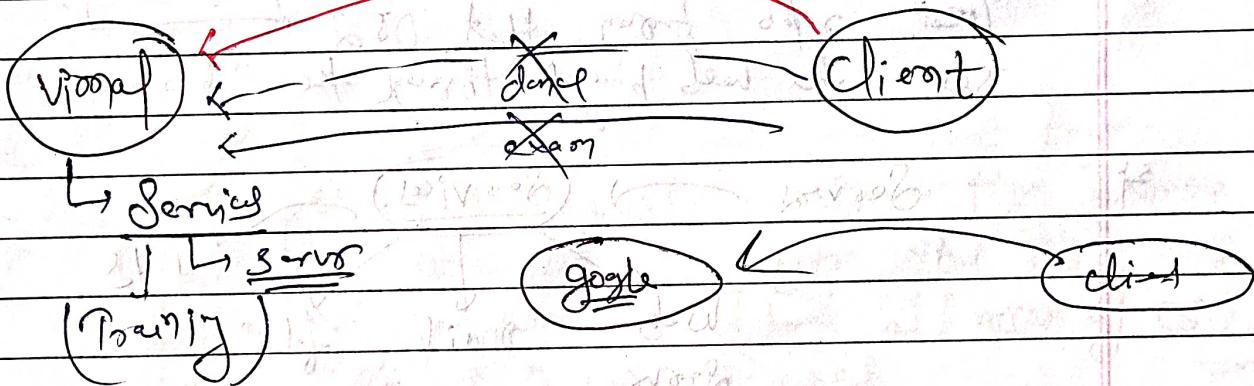
* Server which kind of services they are providing.

Viral will decide which kind of services

↳ (Server) want to provide.

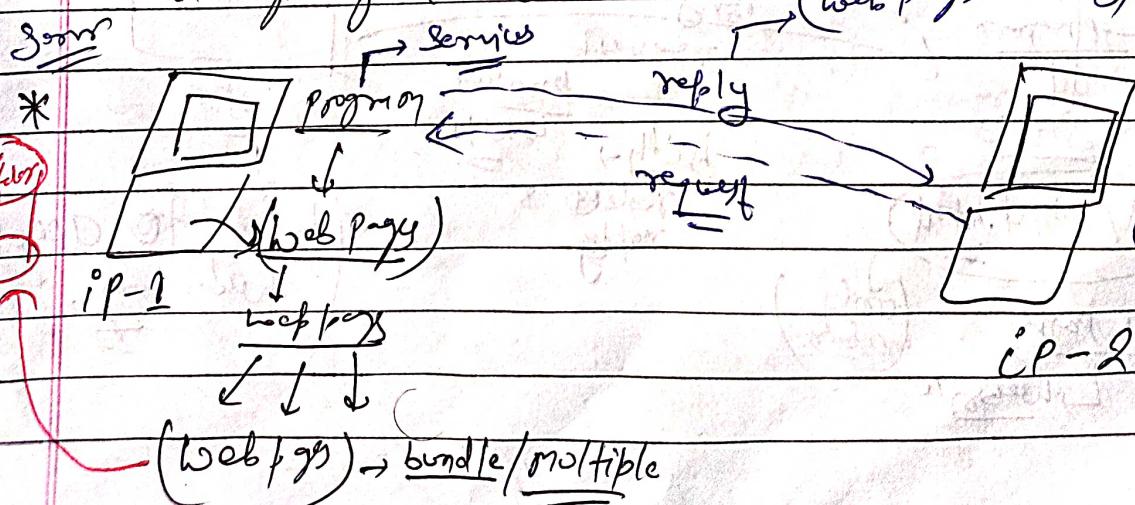
* If any client comes to viral tell his kind of thing he need, here ~~server~~ asked already provide those services i.e., Service requesting.

* Client always go to Server, Server never go to anywhere.



* Example, Google searching → we always go to Provider, Server try to provide.

* Server is the one who provide services, Client always go to Server.



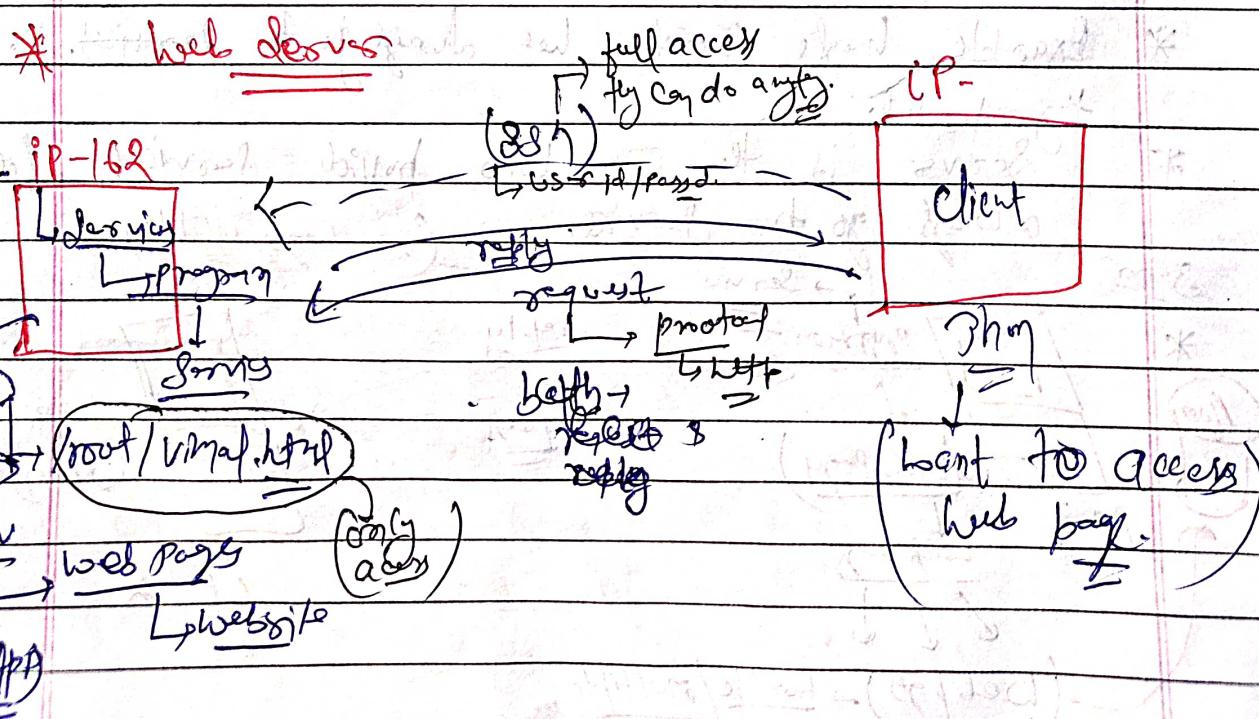
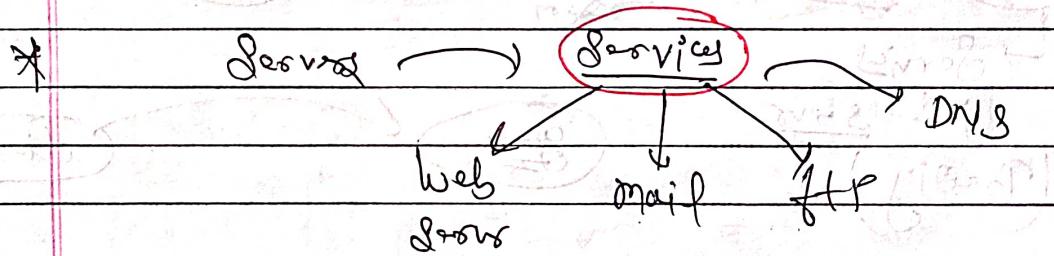
* We have two OS → 1st → In OS one we have web pages → website (multiple web pages here) → this website is in my local system.

* From 2nd OS, Then want to see the web site.
* But Then will never go directly to that laptop & see the web site.

* But Then have a way through 2/3rd way
on the request to 2nd OS.

* OS2 is running with the program, by providing services so OS2 will request to that OS1.

* The web pages are stored in [1 folder], in local repo from 1st OS
see the web pages through the program.



* In IP-162, we start some service that need program, any program do some service to you i.e. server.

*

nslookup

gedit /var/www/html

This is html

→ inside HTML, locally we can see this page.

* Now, Then can read that page they have two way - Goal from other laptop. They can access page.

(1) We can do SSH, come to that system

& get the webpage.

OS2

SSH 192.168.0.162 cat /root/www/html
password: xxxxx

This is html.

→ we have go to that system & read that page.

* But when Q OS-2, will do SSH → they have to get (→ user id) → (After that OS2 will have all access of OS2.
→ password.)

SSH 192.168.0.162 reboot

→ (They can do anything)

We want to give only access to their web pages, web sites. for this they have dedicated program/services.

* This service do one kind of service they go to that location, they have only power to go to that location & get web pages & give access to the OS2.

*

Goal

OS2

for webpage

access that page

OS2

* When any client want to access web page from odd. They can use one program.

* For this Client has to request to that server, they request to that server, they use some kind of protocol. i.e., HTTP (Hyper Text Transfer Protocol)

* Mail Server → (Frontend)

* Every fly will do fly. ⇒ have protocol.

* We are going to use instead of SSH, we will use NGINX program instead of web servers.

* How to Configure Web Servers?

→ 8 Step Process →

NGINX

If we want to configure
NGINX then we have
to follow some steps

one kind of
service → Audio/Video

Client are like
guy we like

NGINX

(1) Install software.

(2) Setup (Look & feel)
→ Configure

(3) Execute. Log / Start Service

→ GUI → file

* The file where we configure P.C, Configuration file.

- * ① rpm
- ② Configure → file
- ③ on → start services.



Media Player

Post a Concept / name

Software / product

(VLC)

(Conda)
media
player

- * Web Server → Concept name

Lots of products of web → ① nginx

② httpd.

- * Software is already available we are going to config on that web server.

- * Configure web zones

→ HTTPD (or)

→ (Apache) → Apache web server

Step ① → httpd - rpm → yum / dnf

Step ② → web page → Dev → Configure.

Step ③ → Services → Start.

- * If you like web server to put your web pages, and also want to provide that web pages to client i.e., web hosting.

rpm -q httpd
dnf install httpd.

Step - 2

dnf list httpd.

* Tell web to put web pages.

* keep file in /var/www/html

(document root)

(web program know the file)

→ If you want to host website you have to copy that page in /var/www/html file.

* /var/www/html → why client come to only retrieve page in few folder.
(by default) folder created

cd /var/www/html

lf (short for list)

prod (hosting)

gedit default.html

prod

lf

page

copy/hosting

Copy

web page in

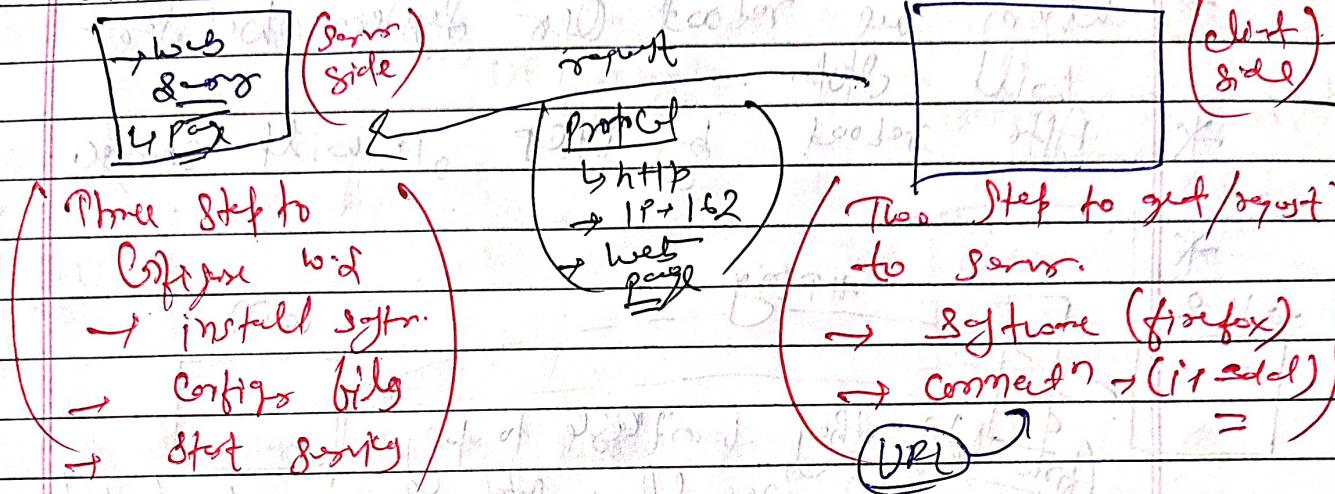
few folder.

/var/www/html

[# systemctl start httpd]

(HTTP 4/6 → Script
HTTP 7/8 → Dynamic))

* Now 162 become server. = (web client)



* Now, if client request to 162, then three they they need.

[http://192.168.0.162 / web page name]

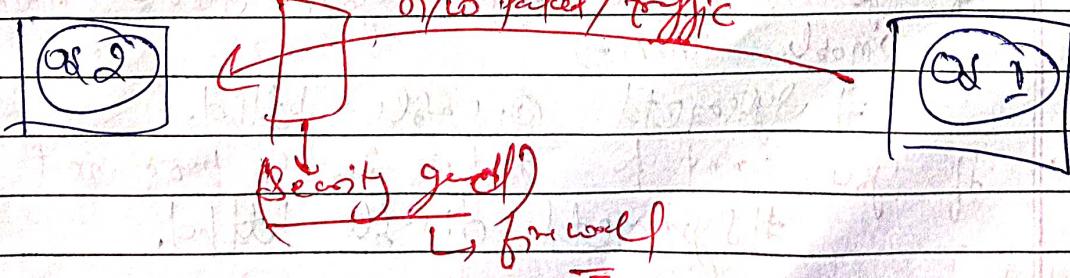
Protocol

IP

Page

→ URL → Uniform Resource Location

* When any one from outside world able to connect to 162, they through σ/π packets, i.e. 162 they have one security guard (they have rule they allows or not) \rightarrow i.e. firewall.



Systemctl stop firewalld

* One of motto of ws, they restricted clients to use / access only (/var/www/html) folder.