

AMAZON'S WEB SERVICESCHARACTERISTIC OF CC

- ① On-demand services
- ② Broad Network Access → access servers from many locations
- ③ Resource pooling → multiple customers are serviced from same physical resources
- ④ Pay-as-you-go mode
- ⑤ Rapid elasticity & Scalability → copy data various servers → fail data issue on other server
- ⑥ Security → copy data various servers → fail data issue on other server

means refers to the servers that are accessed over the Internet present a remote location

Cloud Computing

↳ We store, manage & process data on remote servers

Model Types

- \* ⑨ Data
- \* ⑧ Application
- \* ⑦ Runtime
- \* ⑥ Middleware
- \* ⑤ Operating System
- \* ④ Virtualization
- \* ③ Storage
- \* ② Servers
- \* ① Network

SaaS

PaaS

IaaS

models

SaaS

PaaS

IaaS

\* Cloud computing →

- \* It means storing, managing and accessing the data & programs on the remote servers that are hosted on Internet instead of computer's hard disk
- \* Cloud computing is the on-demand availability of computer system resources

## \* Linux Environment → Flavour of linux (Family)

- ① debian
- ② fedora
- ③ ubuntu
- ④ kubuntu
- ⑤ kubuntu
- ⑥ Red hat
- ⑦ kali
- ⑧ centos
- ⑨ slack
- ⑩ slackware linux
- ⑪ open suse
- ⑫ mandriva
- ⑬ pc linux os
- ⑭ sabayon
- ⑮ linux mint
- ⑯ arch linux
- ⑰ gentoo

## \* cloud computing providers →

- ① AWS
- ② Azure
- ③ google cloud
- ④ Alibaba
- ⑤ IBM
- ⑥ oracle

## \* Types of cloud

- ① public → accessible to all
- ② private → servers accessible within an org.
- ③ hybrid → private + public
- ④ community → Access of group of organization

# Command

-linux os

① **pwd** → present working directory show

② **man** → Info about command show  
(It is manual)  
exit - q

③ **apt-get install** package Name → Install package  
**yum** → Install package

④ **apt-get update** → update command.  
**yum** update.

⑤ **ls -l** → show all folders with all info

⑥ **ls -la** → show all folder with hidden folder

⑦ **cd** Name → change directory  
directory

⑧ **cd ..** → back form directory

⑨ **mkdir** Name → make folder/directory  
directory

## absolute path

- \* starts from root (/)
- \* complete path given to directory or file

- \* In this path move forward or backward

## Relative path

- \* start from current folder
- \* the only path from current folder provide to reach destination

- \* In this path move only forward.

(10)

**touch**

Name

→ make file

cntr S - save

(11)

**nano**

name

→ text editor

cntr X - exit

(cntr O - Name edit)

Enter - exit .)

(12)

**echo**

→ print data on terminal

(13)

**cat**

Name file

→ Show the content in file

File / directory	Format	User Access	Group Access	Other Access	User Name	Group Name	Size
	d	rwx	r-x	r-x	root	root	4096

(14)

**tac**

filename

→ show Reverse Content of file  
from bottom

(15)

**history**

→ show all command you use

(16)

**rm**

→ delete the file

**rmdir**

→ remove directory empty format

**rm -r**

→ remove directory with all data

**rm -rf**

→ remove directory recursive forceful

by

(17)

**mv**source path  
(absolute)destination  
path  
(absolute)→ Move and Rename  
Command

(18)

**cp**

source path

destination

Path

→ Copy & paste  
Command

(19)

**cat >**

file name

→ make file /data print data

cntr D - save

is overriding

(20)

**cat >>**

filename

→ make file /data No override

cntr D - save

(append data)

(21)

**| (pipe)**

→ use for run two command at same time.

(22)

**Head**

filename → document top to line show

**tail**

file name → document bottom to line show

**less**file name → file show one by one exit**Head -n**

15 filename → top 15 line show

**tail -n**

15 filename → bottom 15 line show

(23)

**grep**→ print lines that match pattern  
(filter the data)Cat filename | grep pattern

(date, time, IP, --)

(24)

**wc**

file → Word count show

number of file	word in file	bytes size
----------------	--------------	------------

root user id - always  
DATE  
0 (zero)

## User Management command

cat /etc/shadow

→ In this file user data

exit →

ctrl c

cat /etc/group

- In this file group data

cat /etc/passwd

- In this File password data

① **Add user** name → Adding user with  
password  
other info

② **useradd** name → Adding user without  
make directory

③ **userdel** name → delete the user

## File permission command

u → user

g → group

o → other

The diagram illustrates the breakdown of file access permissions. It shows three columns under the heading "Access": "user", "group", and "other". Each column has a red "rwx" label above it, indicating the permissions for that category. Below each column, there is a red bracket spanning its width, with a small red arrow pointing down from the center of the bracket to the word "Access" below it.

root root  
↑ ↓  
User group  
Name Name  
(owner)

$x \rightarrow$  exists

$W \rightarrow u\bar{u}n\bar{n}$

$\gamma \rightarrow \text{teard}$

①

① chmod

→ change the permission

chmod u+x filename → giving execute permission  
u-x filename → remove execute permission

chmod 777 filename → giving all permission

2

chown

→ change ownership of file

3

~~chgrp~~

→ change group of file

chown    ubuntu    filename  
(name)  
provide  
user  
attributable

chgrp      ubuntu      file name  
(name)  
provide  
group  
Available)

Chow

→ change both ways

group name

at a same time

## Event manager command

- ① **ps** → current/Recent working process show ( PID + process ID )
- ② **ps aux** → detail information of working process  
( CPU utilization, memory utilization )
- ③ **top** → live utilization  
**exit - q**
- ④ **kill** process ID → terminate process  
( End task )

## linux file system (storage)



We save the data on storage & how is retrieve, download, upload the data

### Types of filesystems →

linux  
file  
system

- ① ext4 → latest filesystem
- ② ext2
- ③ ext3
- ④ ext4 → (latest & New  
btrfs Not in use)
- ⑤ \*FS → media specific

window  
file system

- ① FAT
- ② NTFS

df - display file system.

① **lsblk** → shows all block are connected

② **df** → all files show

③ **df -T** → file types show  
(capital)

④ **df -h** → shows all file system with  
size, utilization, available

## Networking Command

\* IPConfig

- ① **IP a** → IP address check
- ② **ping** IP/domain  
Ex-gmail.com → ping to IP address exit-C
- ③ **telnet** domain  
(server) port → any domain specific port connect by this machine
- ④ **curl -F** url (system)  
(https://www.google.com) → connected to other system/server  
(AES  
Call)
- ⑤ **tcpdump -d** → Call traffic network  
see all logs
- ⑥ **netstat** → show all connection with instance
- ⑦ **netstat -tulpn** → see port connection  
(see active internet connection)
- ⑧ **nslookup** domain name → det IPV4, IPV6 of the domain  
(IP check)
- ⑨ **wget -log** copy url → download the file from google

\* For netstat → install net-tools

\* For increase CPU utilization manually in linux.

Install → stress (package)

\* stress -c 5 → Increase CPU utilization

## \* IAM → Identity & Access Management.

- \* IAM ~~use~~ use for create a user under Root
- \* IAM services Not in Region is Global that why every user get all Region.

Define \* IAM Refers to a framework or policies and technology for ensuring that proper people in an organization have the appropriate access to technology resources.

- limit → \*
- \* IAM User limit - 5000 per AWS
  - \* IAM group limit - 300 per AWS
  - \* IAM Role limit - 1000 per AWS

\* Default limit for policies. → IAM Role - 10  
IAM User - 10

- \* IAM User can be member of 10 groups
- \* we can Assign two Access key (max) for IAM user

## \* MFA → multifactor Authentication

- \* It is use for extra security
- \* we can add two-factor Authentication

## FHS - File hierarchy system

(/) → mount point

① /bin → user binary files  
- command used by Normal user

② /boot → Boot loaders files  
(group1, group2) work in

③ /dev → Devices file  
- here storage connected to this pointer  
directory

④ /etc → Configuration files  
user data, passwd, group data

⑤ /home → Home directories  
we create user created data storage in home

⑥ /lib → System libraries  
some help / help directory

⑦ /media → mount point for removable media  
USB mounted

⑧ ~~/tmp~~ /mnt → mount point for temporary filesystem

⑨ /opt → optional Add-on Application

⑩ /sbin → System binaries | super binaries  
super user command

⑪ /srv → service data

⑫ ~~/tmp~~ /tmp → Temporary files

(13) / user → user programs

(14) / var → variable files

(15) / root → Root user directory

(16) / proc → process information

(17) / lost + found → misplaced data

## OSI - open system Interconnection

### TCP/IP Model

① link layer

② Internet layer

③ Transport layer

④ Application layer

### OSI model

① physical layer

② Data link layer

③ Network layer → IPv4, IPv6

④ transport layer → TCP, UDP

⑤ session layer

⑥ presentation layer

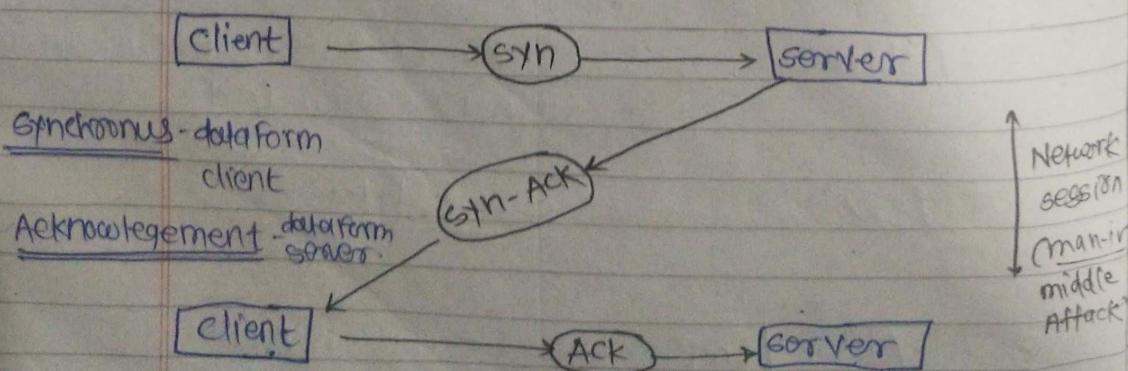
⑦ Application layer - DNS, RTP, DHCP, telnet, FTP

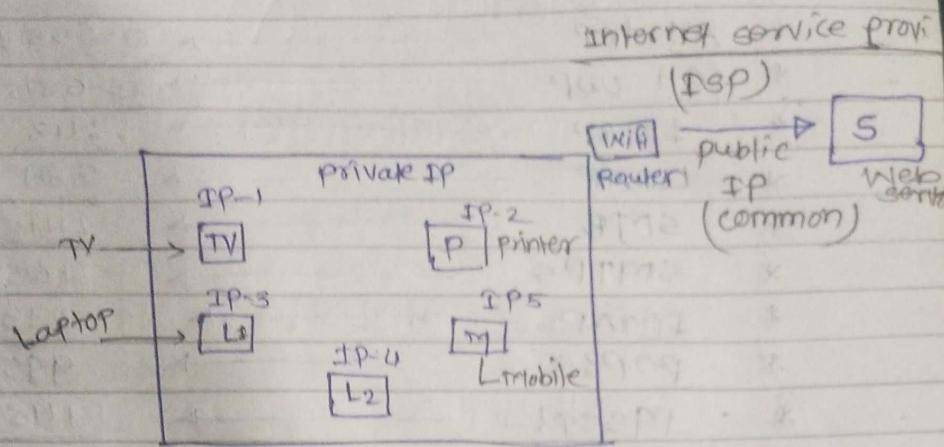
(single Receiver)

TCP → connection oriented (one-to-one Receiver sender)

UDP → In UDP multiple Receiver of packet

3-Way handshake in TCP





- \* UDP → User datagram protocol
- \* TCP → Transmission control protocol

RTP, DHCP,  
et, FTP

- \* private ip for inside the Range → static & dynamic
- \* public IP for over the internet connection

\* In private subnet all ip are dynamic

( All protocol and port No )

proto)

port No.

HTTP	80
DNS	UDP-53    TCP-53
FTP	21
DHCP	67-server    68-client
SSH	22
FTP	21
HTTPS	443
POP3	110
RTP	5004
SMTP	25
telnet	23
RTEP	5005

Network  
segs (81)  
(man-in-  
middle  
Attack)

## Protocol

## Port

* All TCP	→	<u>0-65535</u>
* All UDP	→	<u>0-65535</u>
* IMAP	→	<u>143</u>
* LDAP	→	<u>389</u>
* SMB	→	<u>445</u>
* SMTPS	→	<u>465</u>
* IMAPS	→	<u>993</u>
* POP3S	→	<u>995</u>
* MSSQL	→	<u>1433</u>
* NFS	→	<u>2049</u>
* RDP	→	<u>3389</u>

IP-Internet protocol Address → IPV4  
→ IPV6

## \* classfull IP Addressing \*

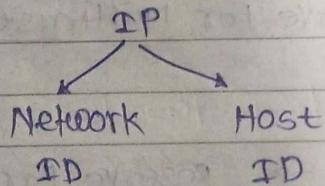
\* [Binary to decimal] → ~~1001~~

$$\begin{array}{r} 1001 \\ \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\ 2^3 \quad 2^2 \quad 2^1 \quad 2^0 \\ \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\ 2^3 \times 1 \quad 2^2 \times 0 \quad 2^1 \times 0 \quad 2^0 \times 0 \\ \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\ 8 + 0 + 0 + 1 = 9 \end{array}$$

1 byte = 8bit (octet)

$$2^8 - 2^0 = \frac{1 - 256}{256} = 256$$

\* IPV4 → 198.168.12.34



\* This two part of IP separation depend on class of IP Address

\* Total 5 class of IP

\* In this A, B, C use for Normal

\* D, E class Reserve for multicasting, Broadcasting, Research & development

\* IPV4 use contain 32 bit

(bit Reserve for to Identify the class)

\* first bit / did  
last bit / not mean

class

	Range	Length	Size of N/W	Size of N/W ID	Size of Host ID
A	0 - 126	1 bit - 0 Reserve	$2^{24}$	126	$2^4 = 16 \text{ million}$
B	128 - 191	2 bit - 10 Reserve	$2^{16-2} \rightarrow 2^{14}$	16884	$2^{16} = 65536$
C	192 - 223	3 bit - 110 Reserve	$2^{24-3} \rightarrow 2^{21}$	2 million	254

In Host First bit Reserve for N/W and last bit Reserve for Broadcast

D

Reserve for Multicast, R&D

E

Range - 127

→ Reserve for loop back

(127.0.0.0/8)

Difference IPv4 & IPv6

IP V4

IP V6

\* 32-bit

\* 128-bit

\* 8-bit \* 4 group

\* 16-bit \* 8 group

\* 32-bit, 128-bit

\* 2128

\* 232

\* 128-bit

\* Max number of IP.

bit did not mean.

When we creating subnet → N/W ID denoted by 1  
Host ID denoted by 0

size of Host ID

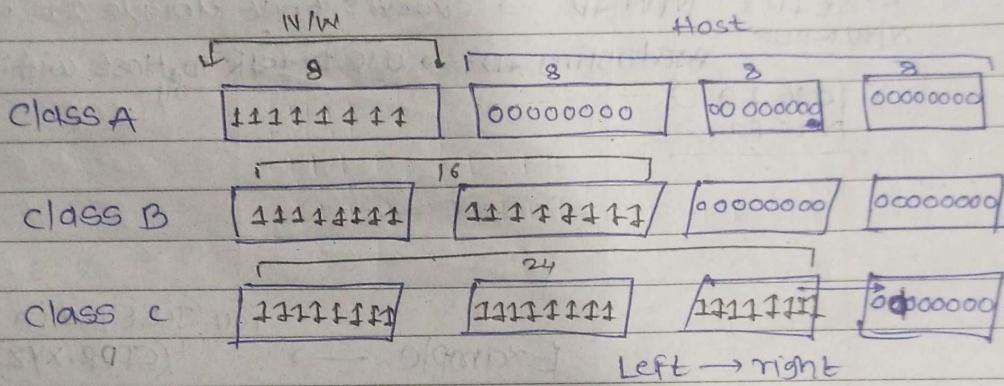
$$2^8 = 16 \text{ Million}$$

$$2^{16}$$

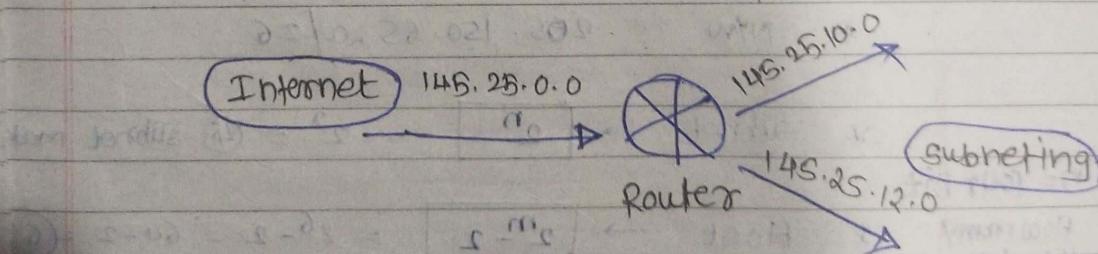
$$2^{16} - 1 = 65535$$

Subnet → The practice of dividing network into two or more networks is called subnetting

$$\begin{array}{cccccccccc} \rightarrow & 2^0 & 2^1 & 2^2 & 2^3 & 2^4 & 2^5 & 2^6 & 2^7 & 2^8 \\ - \rightarrow & 1 & 2 & 4 & 8 & 16 & 32 & 64 & 128 & 256 \end{array}$$



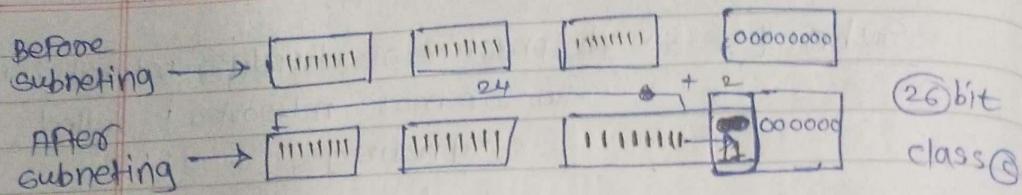
- \* In classfull IP addressing too much waste of Host IP that's why ~~CIDR~~ (Classless Inter Domain Routing) Established
- \* CIDR use your utilization You create a subnets.
- \* Network bandwidth use intelligently.



- \* subnet mask → class A - 255.0.0.0  
class B - 255.255.0.0  
class C - 255.255.255.0

- \* subnetting always do left to right

CIDR Block → decide the ~~host ID~~ which bit goes to N/W ID & Host ID



\* → How many bits to N/W & Host

\* Subnet Mask → How many bits to N/W & Host  
\* CIDR → decider which bit is N/W & Host  
(CIDR Notation) (/ Notation)

First IP & N/W ID → every subnet unique ID based on IP Range  
& Broadcasting IP → use to talk to Host within your all  
last IP ↑  
N/W Range.

class-c

$$2^7 = 128$$

$$2^6 = 64$$

$$2^5 = 32$$

$$2^4 = 16$$

$$2^{10}$$

You just put in on  
Example → (CIDR, xyz) web

Class C → 205.150.65.0 / 26

2 bits remain

\* Subnet AND Mask → 255.255.255.192 →  $2^7 + 2^6$   
 $128 + 64 = 192$

\* N/W 205.150.65.0 / 26

\* Subnet →  $2^n$  =  $2^2 = 4$  subnets made

n = your bit  
How many bit give to Network

\* Host →  $2^m - 2$  =  $2^6 - 2 = 64 - 2 = 62$

m = your bit

How many Remaining

\* CIDR Base IP / N/W → 205.150.65.0

+ Broadcast IP → 205.150.65.63

\* Host IP → 205.150.65.1 First use

205.150.65.62 Last use

205.150.65.0 / 24

PAGE NO. \_\_\_\_\_  
DATE: \_\_\_\_\_  
create 10 subnets  $\rightarrow 2^h = 16$   
need 4 bits

class-c IP address 205.150.65.0 / 28

Subnet Mask: 255.255.255.240

$$\underline{2^7} = 128$$

$$\underline{2^6} = 64$$

N/W

205.150.65.0 / 28

$$\underline{2^5} = 32$$

$$\underline{2^4} = 16$$

(240)

$$* \text{ subnet } - \frac{2^7}{2^m} = 2^4 = 16$$

$$* \text{ Host } - \frac{2^m - 2}{2^m - 2} = 2^4 - 2 = 16 - 2 = 14$$

\* GIDR Base IP = 205.150.65.0

\* Broadcast IP = 205.150.65.15

\* Host IP = 205.150.65.1

205.150.65.14

(1000000)

(1111111)

bits are given

$$2^7 + 2^6$$

$$128 + 64 = 192$$

let make

$$2^2 = 62$$

# DORA Process

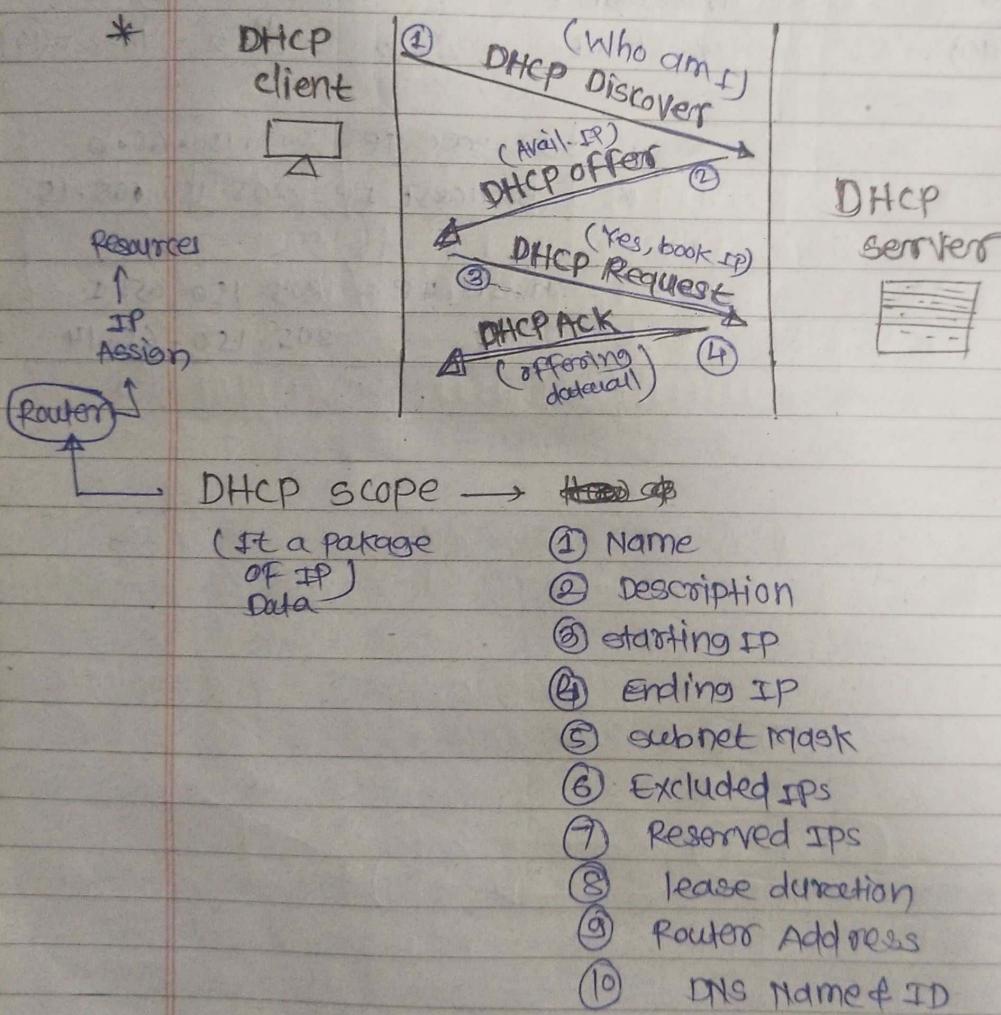
PAGE NO.  
DATE:

## Dynamic Host Configuration protocol

\* DHCP give you your Identity



It work on 4 step process

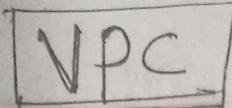


\* DORA - Discover Offer Request Acknowledge

\* Router

Region-Physical  
things  
AZ's → log  
1

Region - physical  
AZ's → thing logical thing



PAGE NO.  
DATE:  
→ create a virtual network

Where EC2 instance connect to communicate

## Virtual private cloud

WPT

- \* When we created VPC → Automatical (3) default thing generate
  - ① DHCP
  - ② Route table
  - ③ N/W ACL
- VPC create in Region
- VPC can't private or public.
- Restricted users N/W level

- \* Internet Gateway → Work on over the internet
- Attach on VPC level
- one VPC only one IGW
- IGW Entry Route table → Attach
- Entry → always 0.0.0.0/0

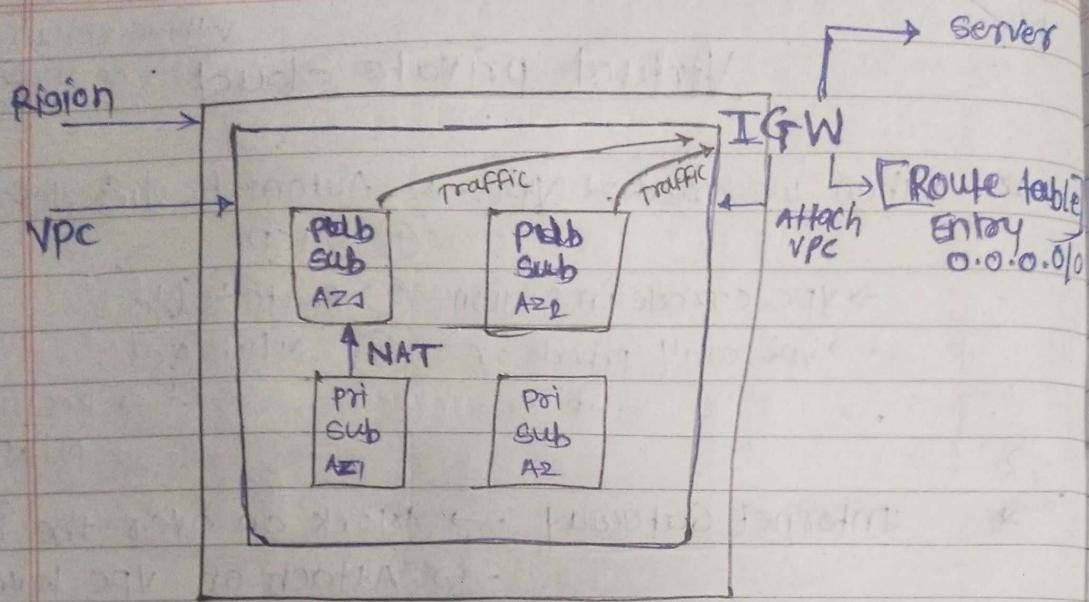
security of data  
high connectivity

- \* Subnet → Subnet created in AZ's
- security purpose we can create public subnet & private subnet
- work on over the internet.
- Route table essential for subnet without Route table subnet not create
- we add → IGW entry in Route table then our subnet converted as public subnet
- If IGW entry not in Route table subnet are private.
- In one Route table we can Associate multiple subnet.

ion  
ness  
2 ID

est Acknowledge

- \* Route table → It work as gateway
- Via Route table we can send out traffic to the server and receive the traffic



private IP ← → public IP

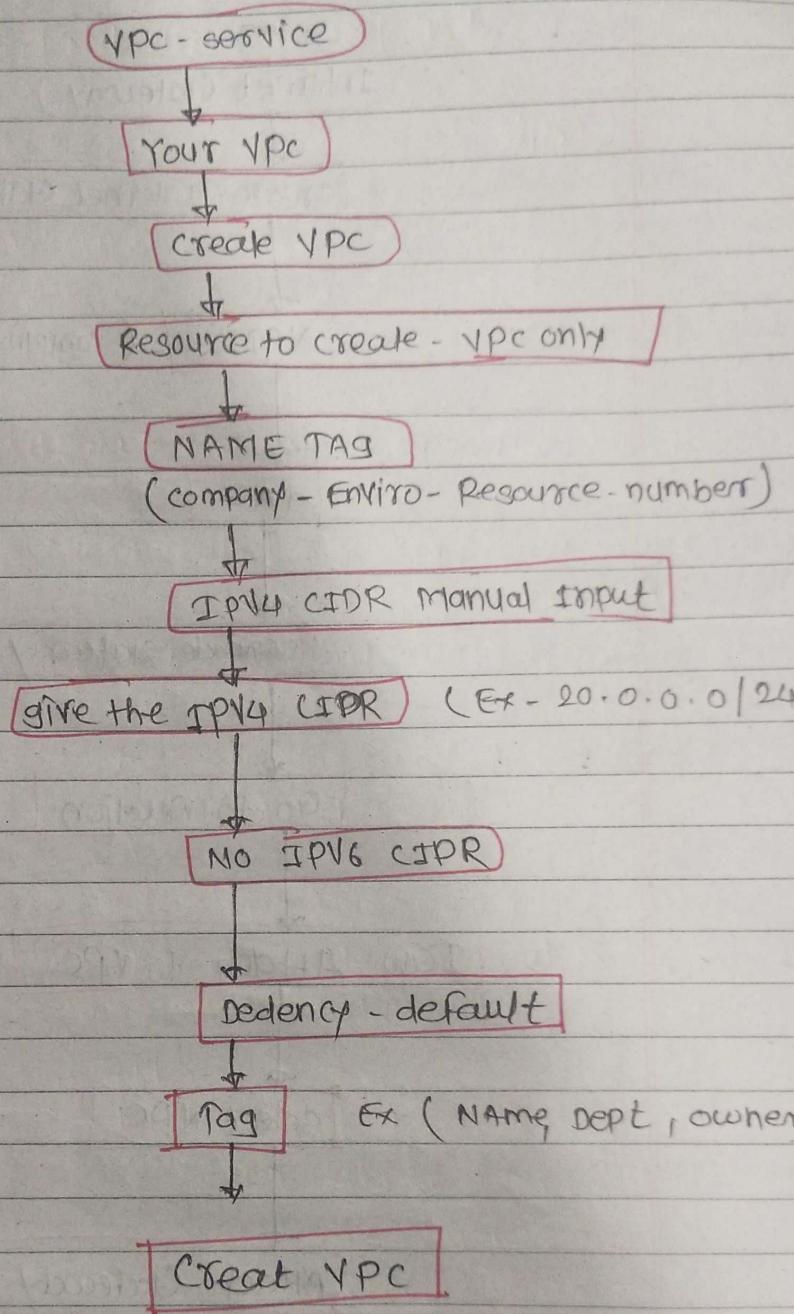
IP NAT

(Network Address translation)

→ Server

[Route table]  
entry  
0.0.0.0/0

↑ Region  
Create VPC → SOP

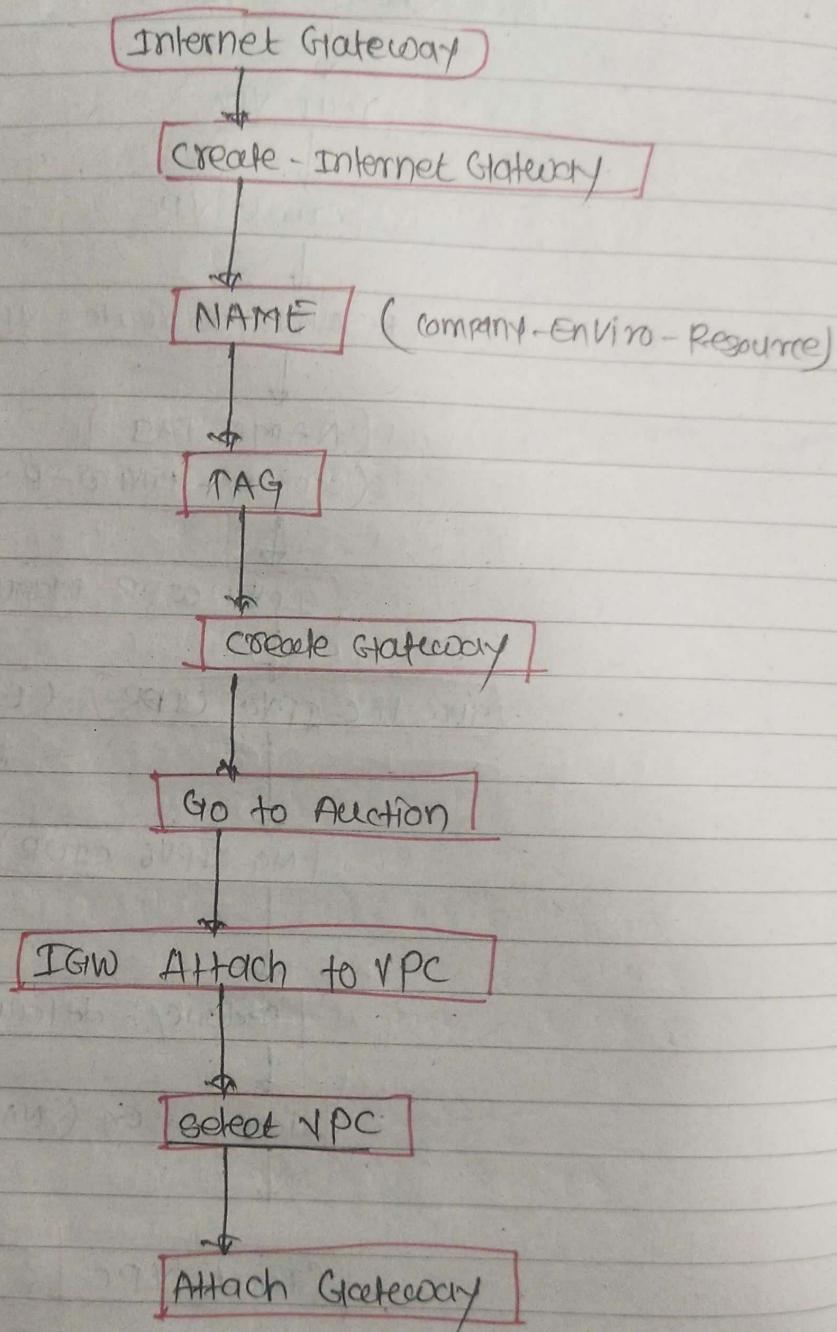


Now Automatically -

② DHCP

③ N/W ACL Make

## create Internet Gateway - IGW

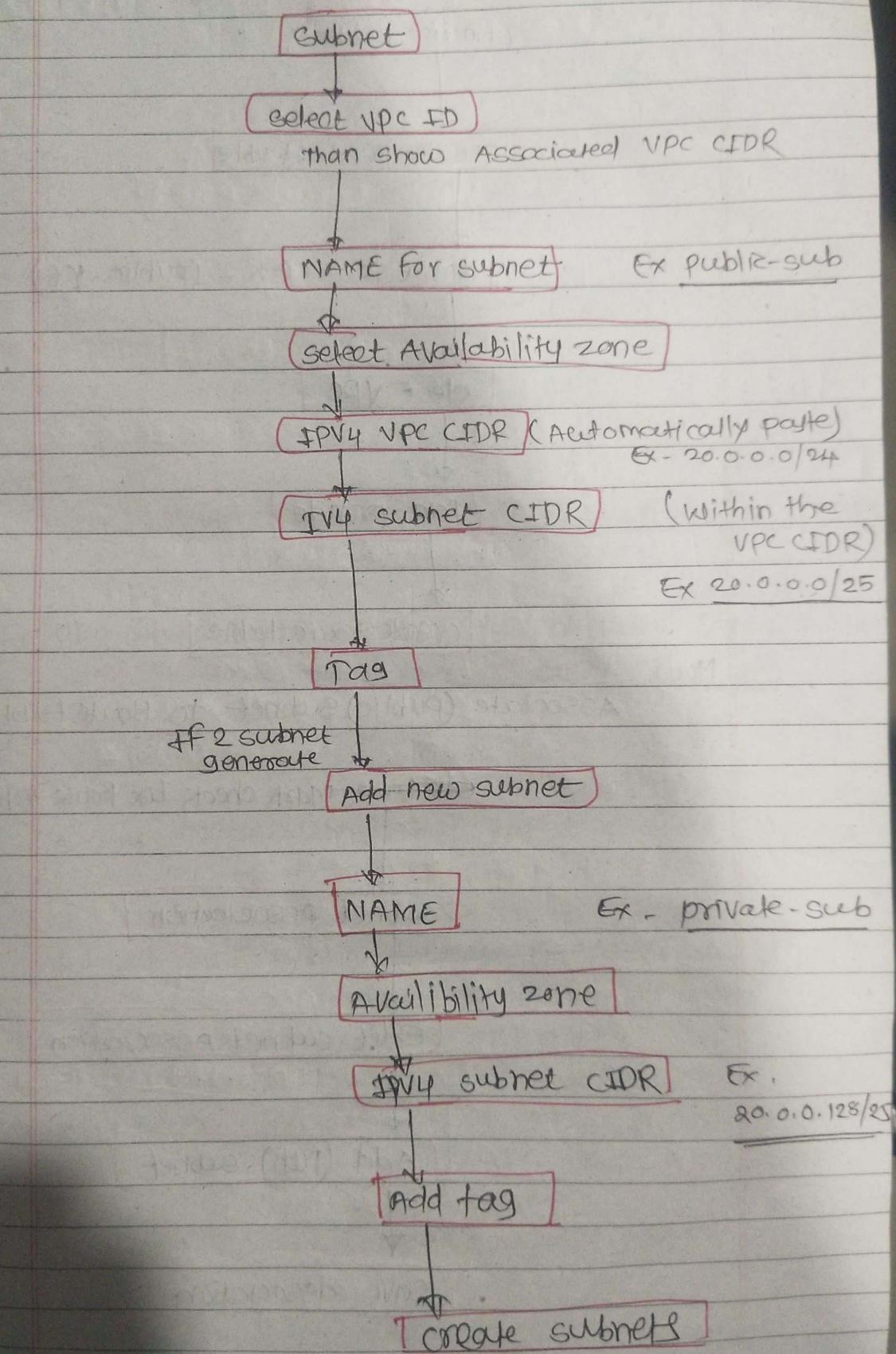


ARZ

PAGE NO.

DATE

create subnet →

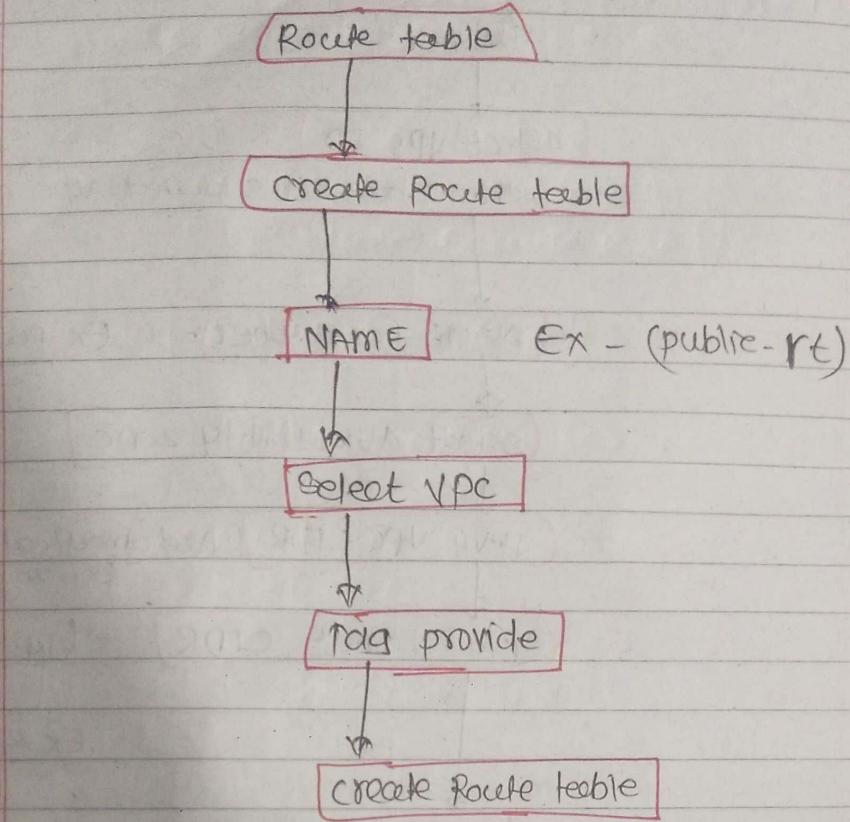


\* Subnet Associate with directly default - route table

## Route table

PAGE NO.

DATE:



Now,

Associate (public) Subnet to Route table

~~Select~~  ~~Mark~~ check box Route table checkbox

Subnet Association

Edit subnet association

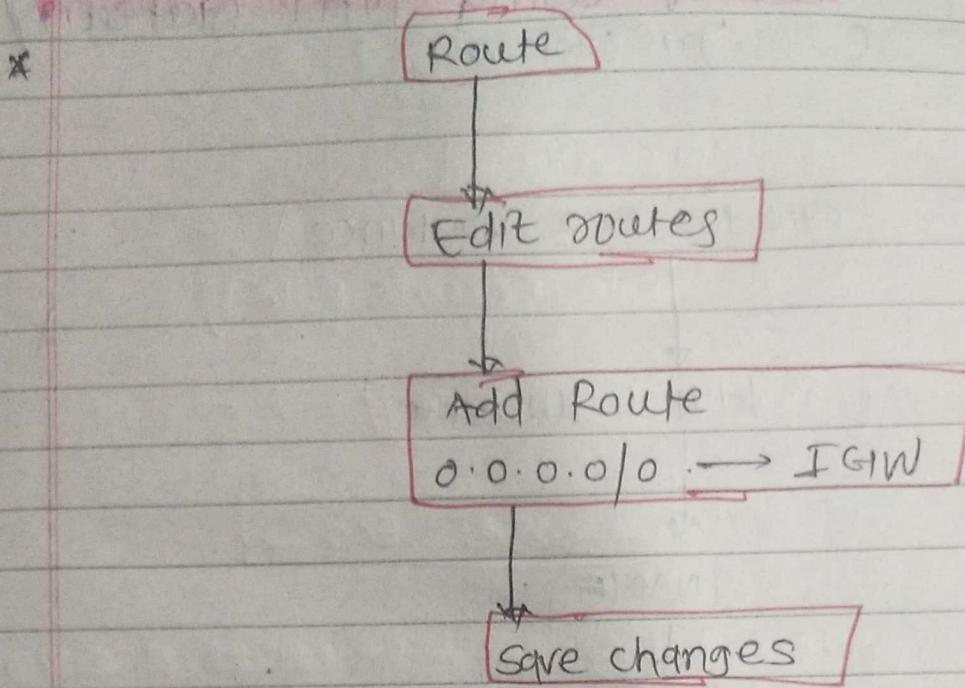
Add (pub) subnet

Save association

Now -

\* give Entry of IGW in Route table

## Entry of IGW in Route



Now Your subnet are public subnet (Authorized)

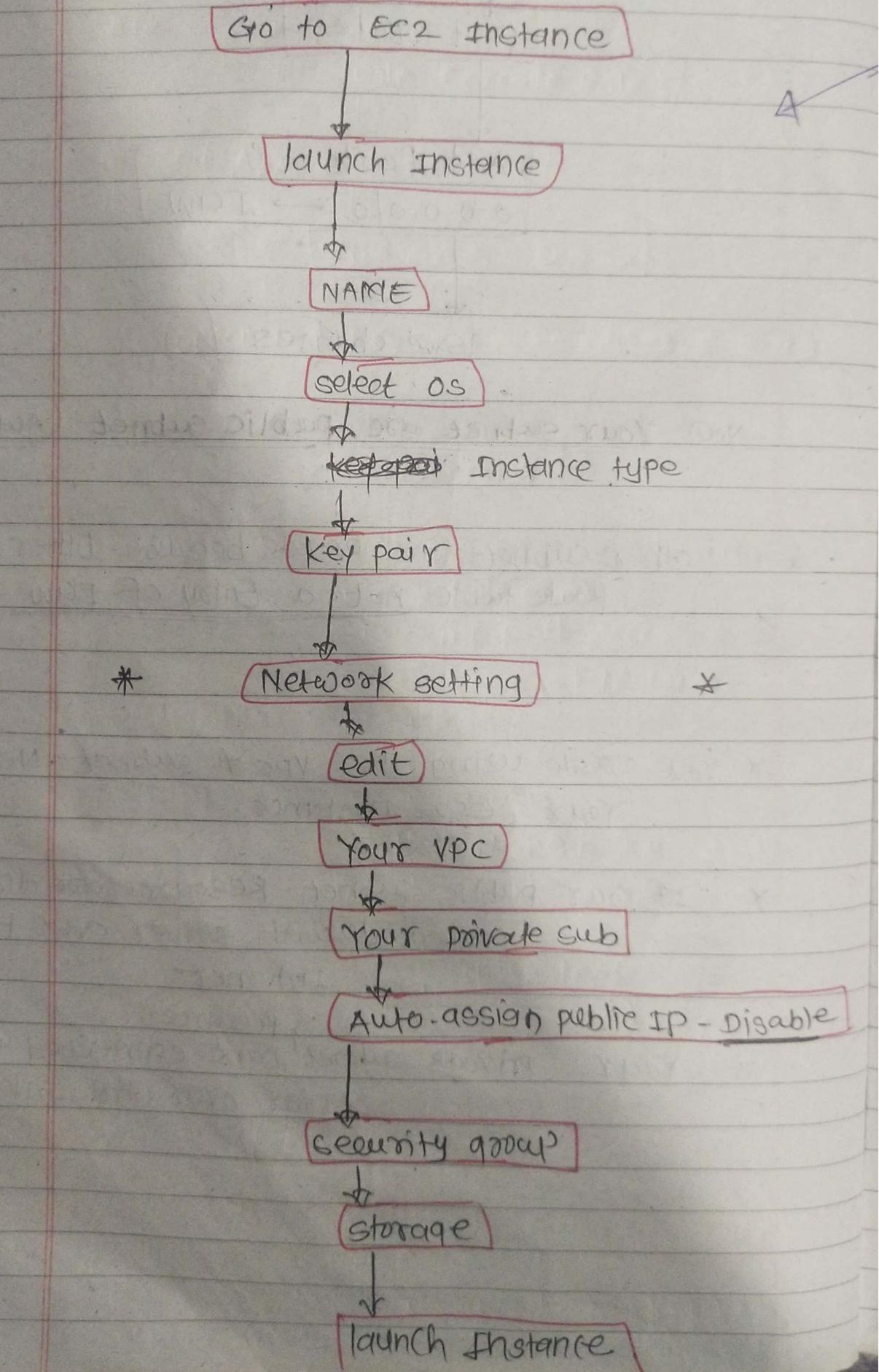
\* and other subnet are private because Other subnet  
route table not a Entry OF IGW

\* You create using this VPC & subnet Network  
Your EC2 Instance.

\* ~~your~~ public subnet Resource are directly  
connected with server over the  
Internet

\* Your private subnet <sup>Resource</sup> can't connected the  
server over the Internet

# \* private subnet Resource connected to server using NAT Gateway . (over the internet)



# EC2 instance require public IP

Now

①

Go to VPC service

②

NAT gateways

Create NAT gateway

NAME

Select subnet

(where NAT will create which is our public sub)

Connection type - public

Allocate Elastic IP

Tag

Create NAT-Gateway

Now Entry of NAT in Route table.

Go-to-private Route table

Associate gateway (PUB)

Route

Edit

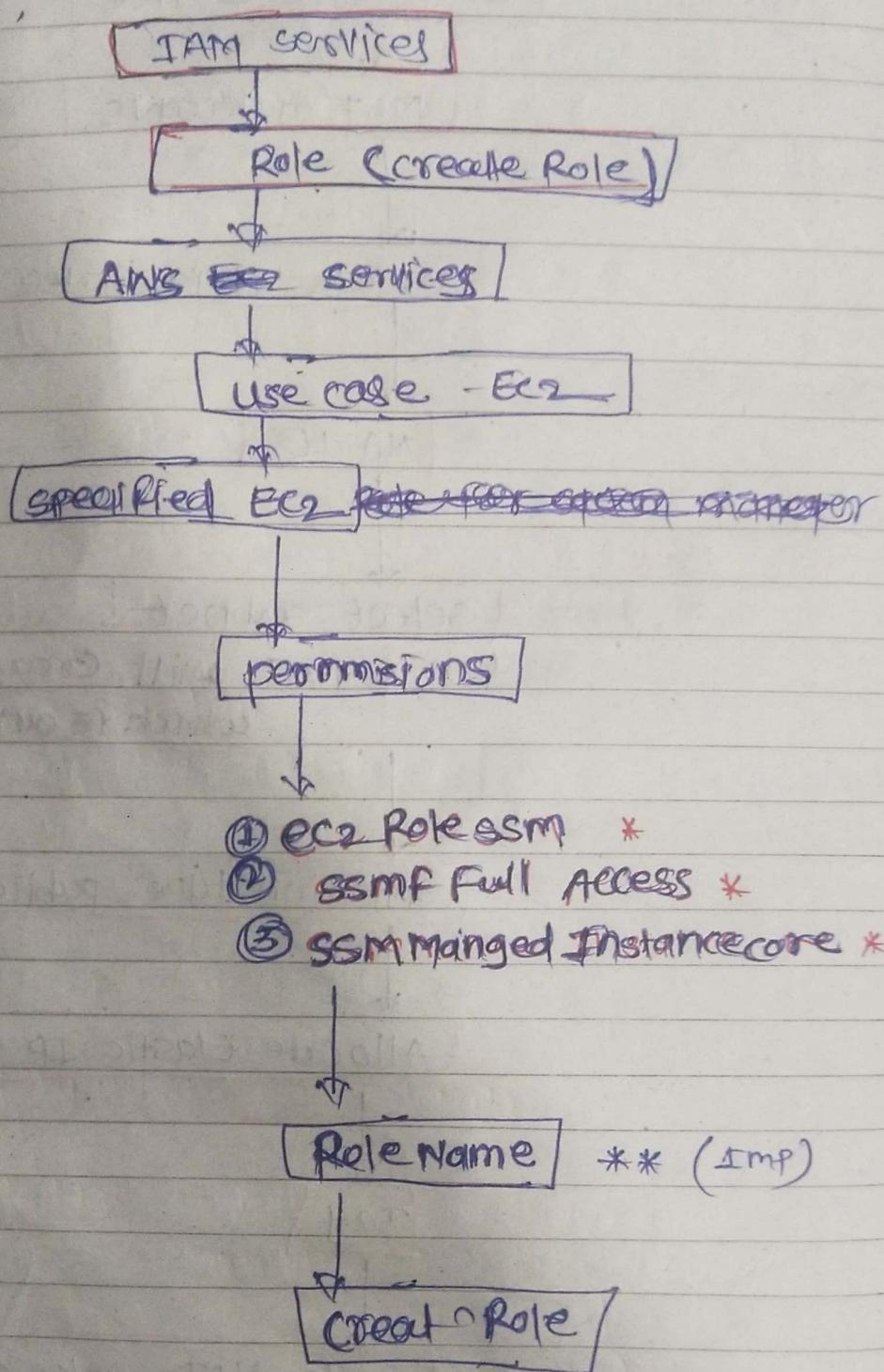
0.0.0.0/0. NAT gateway

PAGE NO. \_\_\_\_\_  
DATE: \_\_\_\_\_

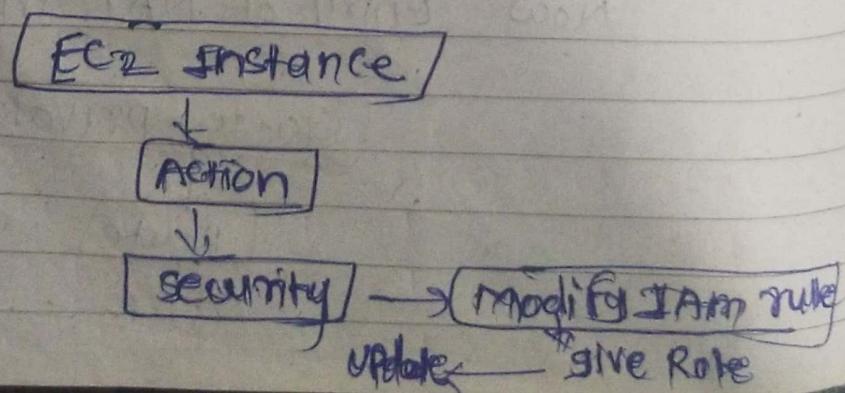
Now EC2 instance connection Established without public IP

### Role

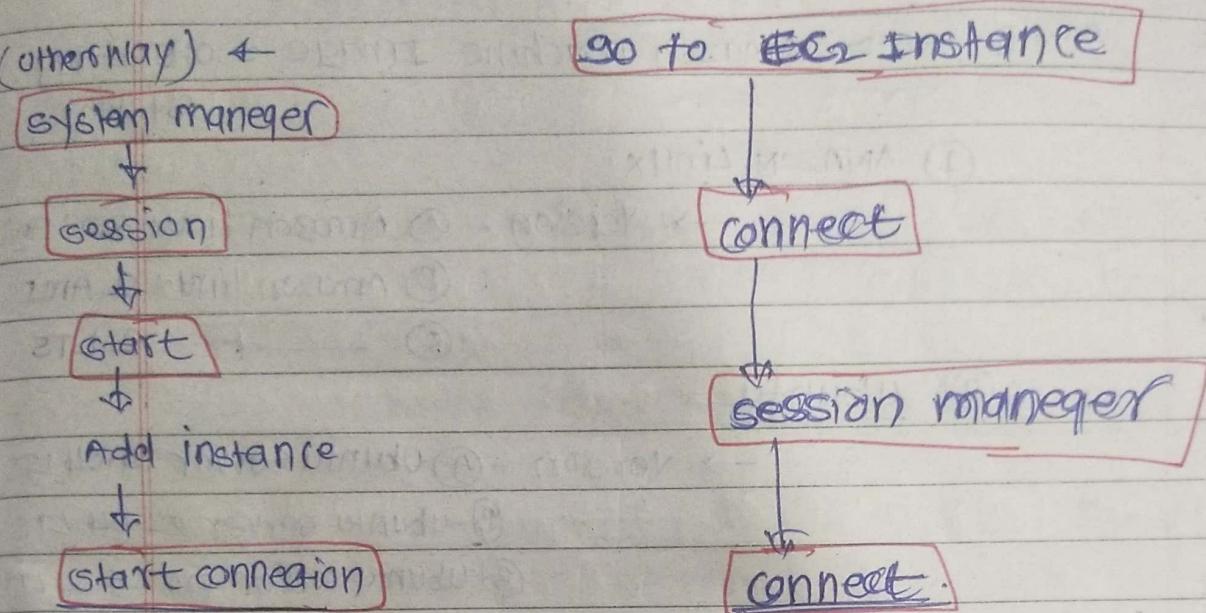
Now,



Now



Now



\* Route entry - Remove NAT

\* Elastic IP → Release

Imp

↳ If instance are  
running  
elastic IP  
can't release

\* elastic IP means it is a Static public IP  
using this public IP go to the over the  
internet

Ans limitation →

It is a service in AWS  
DATE \_\_\_\_\_  
EC2 → provided Resizable compute capacity in the cloud

## \* elastic compute cloud \*

\* AMI → Amazon machine Image = operating system.

### ① AMAZON Linux

- Version - ① Amazon Linux 2023 AMI  
② Amazon Linux 2 AMI  
③ — + 2 LTS

### ② ubuntu

- Version - ① Ubuntu Server 24.04 LTS  
② Ubuntu Server 22.04 LTS  
③ Ubuntu Server Pro 24.04 LTS

### ③ Windows

- Version - ① Windows Server 2022 Base  
② Windows Server 2022 Core Base  
③ Windows Server 2019 Base  
④ — + L — 2019 Core Base  
⑤ Windows Server 2016 Base  
⑥ — + L — 2016 Core Base

## \* Architecture →

Two types of Architecture

- ① 64-bit (x86)  
② 64-bit (Arm)

## \* Instance type →

### ① General purpose

### ② Compute optimized

### ③ Memory optimized

### ④ Storage optimized

### ⑤ Accelerating computing

### ⑥ High-performance computing

LTS  
long term support

### Instance type

### Family

#### ① General purpose

T2, T3, T3a, T4g,  
M5, M5a, M5d, ~~M5dn~~  
M5dn, M7a, M7g

#### ② Compute optimized

C5, C5a, C5d, C5ad,  
C6, C6a, C6g, C6ad

#### ③ Memory optimized

R5, R5a, R5ad, R5d

#### ④ Storage optimized

D3, D2, D2en, H1

#### ⑤ Accelerated computing

P2, P3, G5, G6

#### ⑥ High performance computing

Hpc6a, Hpc7a

### \* Storage →

\* Pre-tier eligible customers can get up to  
30 GB of EBS General purpose SSD.

### \* Types of storage

- ① general purpose SSD GP3
- ② general purpose SSD GP2
- ③ provisioned IOPS SSD io1
- ④ provisioned IOPS SSD io2
- ⑤ cold storage (SC1)
- ⑥ throughput optimized (HDD) ST1

\* Default storage → mac AMI → 100 GB storage

Debian + Ubuntu AMI → 8 GB storage

Windows AMI → 30 GB storage

Amazon AMI → 8 GB storage

\* SUSE + Redhat AMI → 10 GB storage

## \* Security group →

It is used as firewall  
allow or deny the traffic

\* NAME → Description provided to  
New created SG

Add rule

\* Inbound Rule → client → server

source - port - protocol

\* outbound Rule → server → client

destination - port - protocol

\* Automatically Make ENI

↳ Existing Network interface

Attach to the Instance

\* Security group work on ENI

\* Security group are stateful

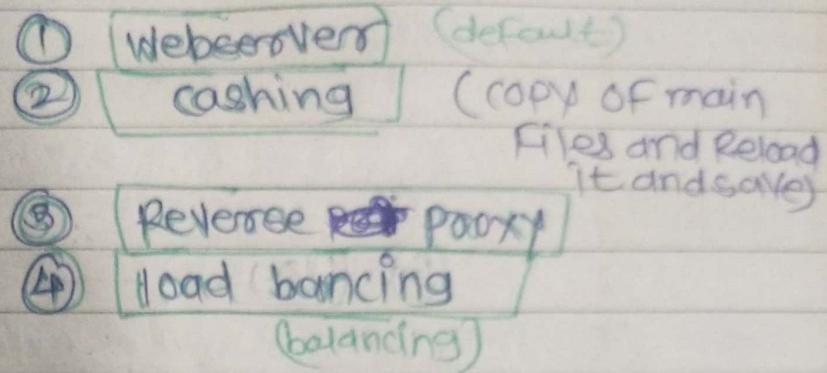
↳ means we edit inbound rule  
we not require edit  
outbound rule  
it work automatically.

# Nginx Server

PAGE NO.  
DATE

- \* It is the server to helps us Response efficiently

## Types of work of Nginx



- \* Nginx is open source.
- \* Nginx is work into the public subnet

## Command

- \* service Nginx status → use for check status running or not.
- \* service Nginx reload → if any packing there than we use to Reload
- \* service Nginx start → start form the Starting Nginx.
- \* service Nginx stop → stop Nginx
- \* IF Website down → Three parameters to check

- (A) CPU utilization (Heavy load)
- (B) security group (Inbound Rule not present)
- (C) server Running or not

## \* IMP Files

(A) on Nginx.conf → all info are present in this file which is imp to run the Nginx

(B) logs → all traffic logs are present.

(C) conf.d → all endpoint rule are made present in this file

(D) sites-enabled → uses for webpages

\* there are two ways to make Homepage

① cd /var/www/Html

② cd /usr/share/nginx/Html

→ using nano edit file  
→ cp the file in

/var/www/Html

## AMI → Amazon machine Images

- \* An AMI is an image provided by AWS that provides the information required to launch an instance.
- \* You can launch multiple instances from a single AMI when you require multiple instances with same configuration.
- \* When AMI created
  - ② EBS snapshot are automatically made.
  - (\* When EBS snapshots are created that time AMI not created)

### \* TYPES AMI →

- ① owned by me
- ② private image
- ③ public image
- ④ disabled image

- \* AMI created when before packing of instance after packing of instance for purpose backup.

### \* TWO WAY AMI →

- ① directly from AMI dashboard
- ② Launch EC2 instance
  - ↳ select AMI owned by me.

### \* How to create →

Your main instance  
Reboot all config change that why.

\* Enable

No

Reboot ← NAME ← Create Image

① EC2 instance select



Action



Create Image

# ASG → Auto Scaling group.

- \* Amazon Ec2 auto scaling helps you ensure that you have the correct number of Amazon Ec2 instances available to handle the load for your application
- \* You create collection of Ec2 instance, called Auto scaling group.
- \* It is specify the minimum number of instance running and ensure your application never goes to down.
- \* same for maximum number and desired capacity
- \* If you specify the policies, then Auto scaling can launch or terminate instance as demand on your application increases or decreases

Imp → Auto scaling group required template or configuration template

Vertical scaling → Manually work

Horizontal scaling → Work on automation

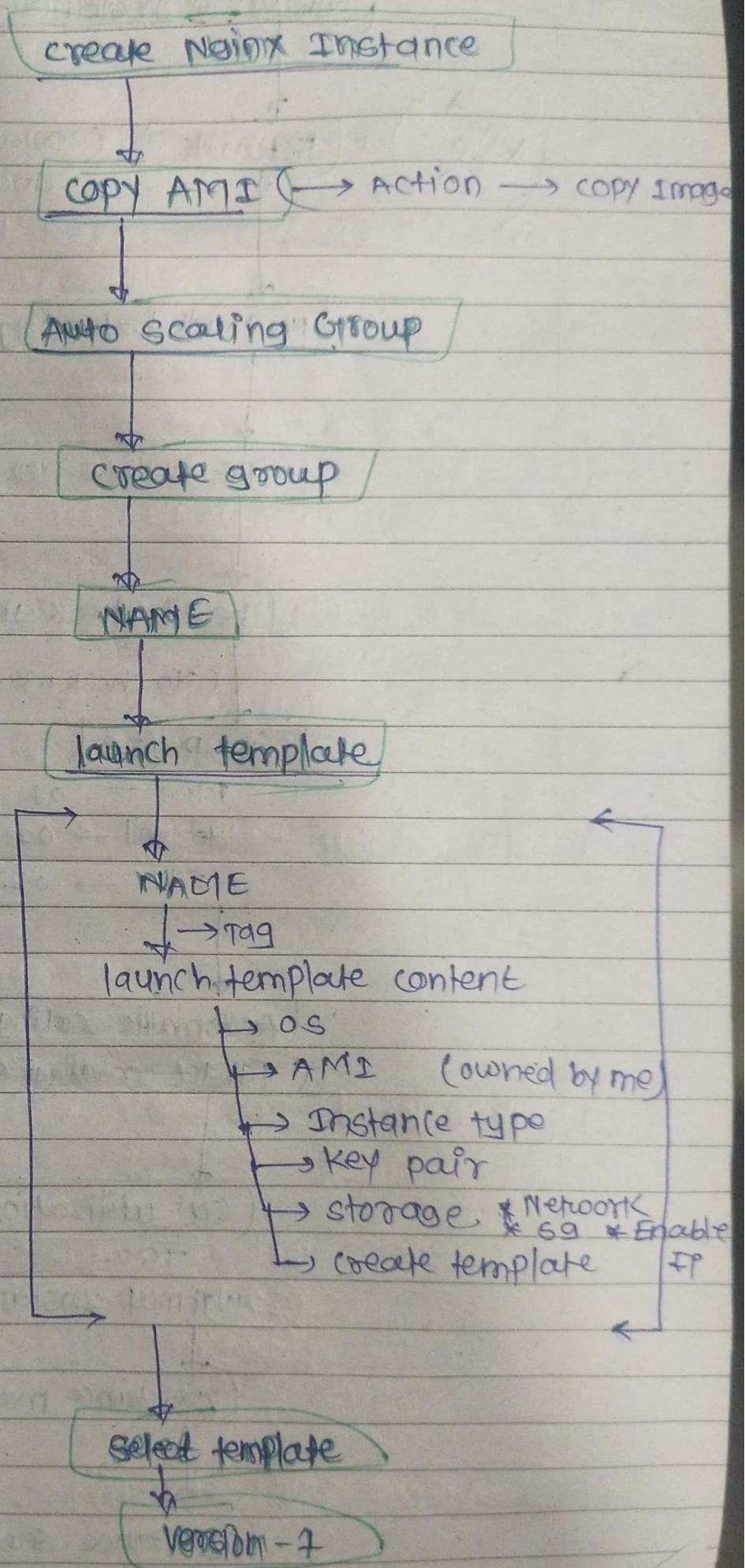
## \* Features →

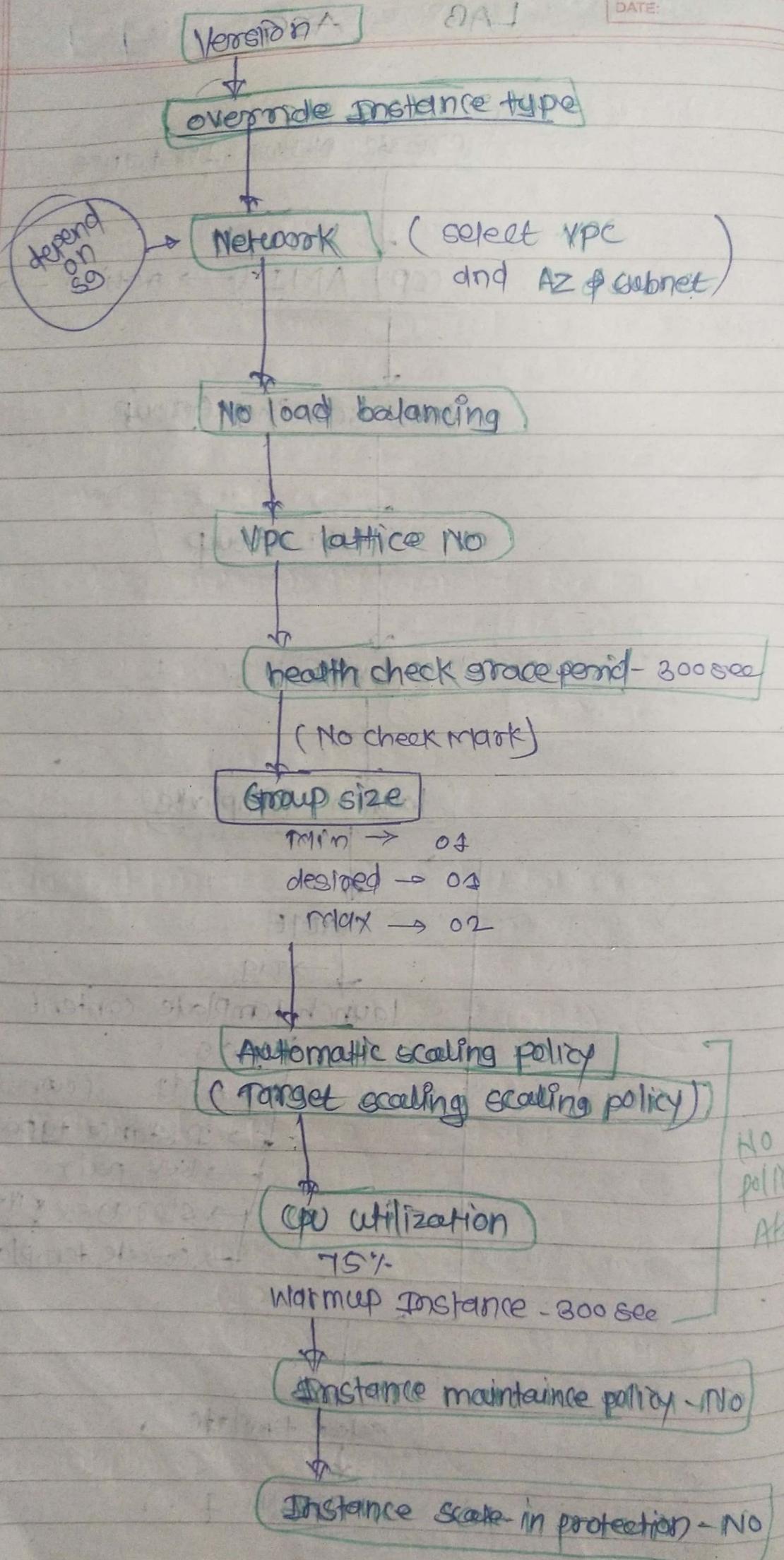
- ① Monitoring ~~to~~ health
- ② Multi Availability zone (Working)
- ③ Multi Instance type & purchase option
- ④ Automated spot instance  
On-demand by AWS
- ⑤ Scalability
- ⑥ Load balancing
- ⑦ Instance refresh.

# LAB - ASG

PAGE NO.

DATE





## Add Notification

Tag

- \* NOW, When load increases / CPU utilization above TS1, another instance create

- \* To check →

① Connect instance with putty

apt-get install stress

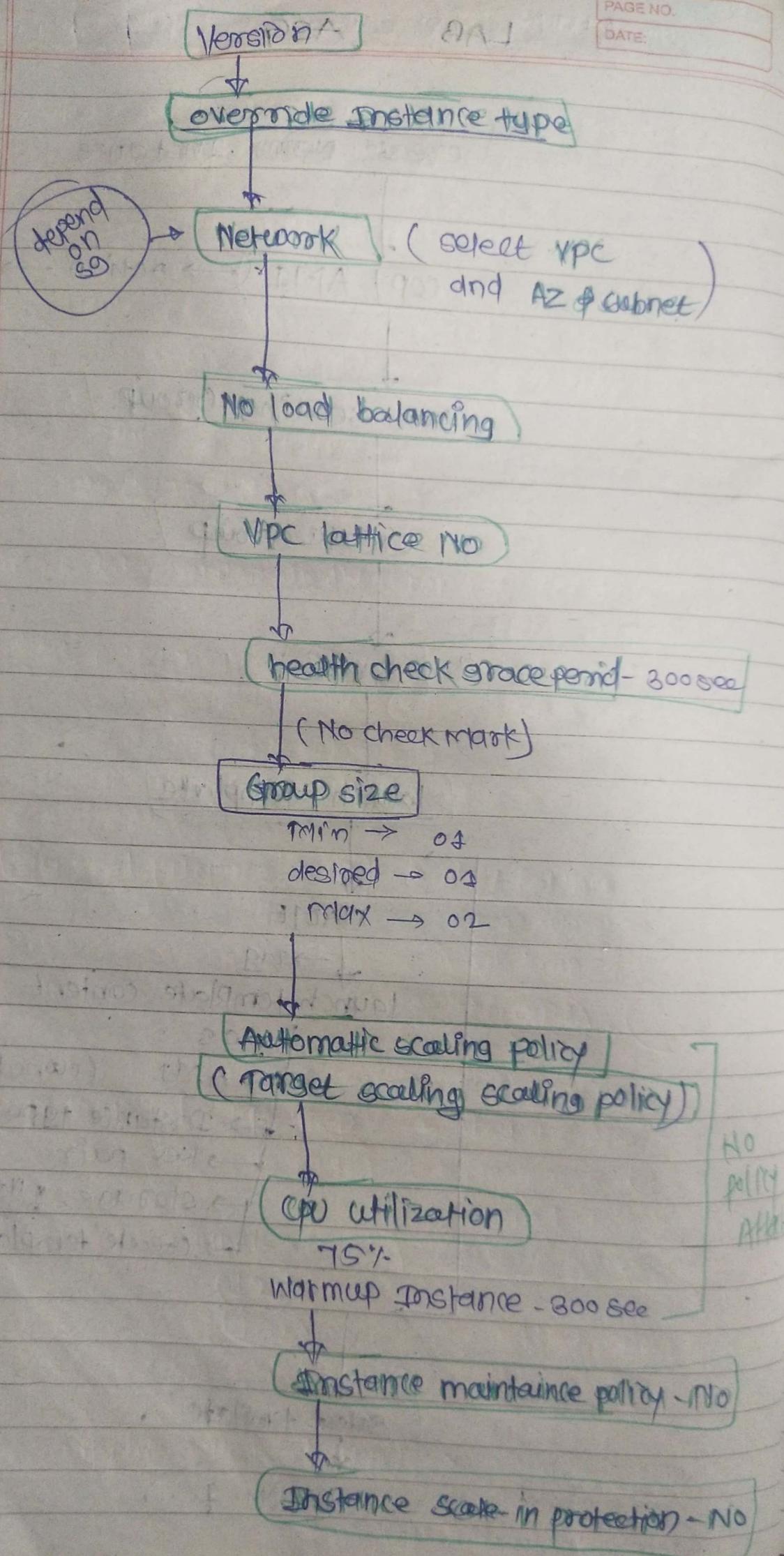
\* stress -c 5 (command)

\* top (check utilization)

\* Health grace period

\* Warmup → for instance scale in

\* cooldown → for instance scale out



## Add Notification

↓  
**Tag**

- \* NOW, When load increases / CPU utilization above TSI, another instance create

- \* TO check →

① Connect instance with putty

apt-get install stress

\* stress -c 5 (command)

\* TOP (check utilization)

- \* Health grace period

\* Warmup → for instance scale in

\* cooldown → for instance scale out.

# EBS → Elastic Block Storage

- \* It is a Network drive to help us to storage our data.
- \* EBS connect with multiple instance
- \* multiple EBS volume connected to single instance
  - ① default EBS - root volume
- \* EBS volume and running instance are in same AZ's
- \* EBS take automatically snapshot for backup.
- \* key management service → use for ~~store~~ Encrypt the data

## EBS TYPE

- ① General purpose SSD → they balance price and performance for a variety of transactional workload

TYPE → GP2, GP3
- ② provisioned IOPS SSD → They are the highest performance EBS volume designed for critical, IOPS-incentive, throughput intensive workloads that require low latency

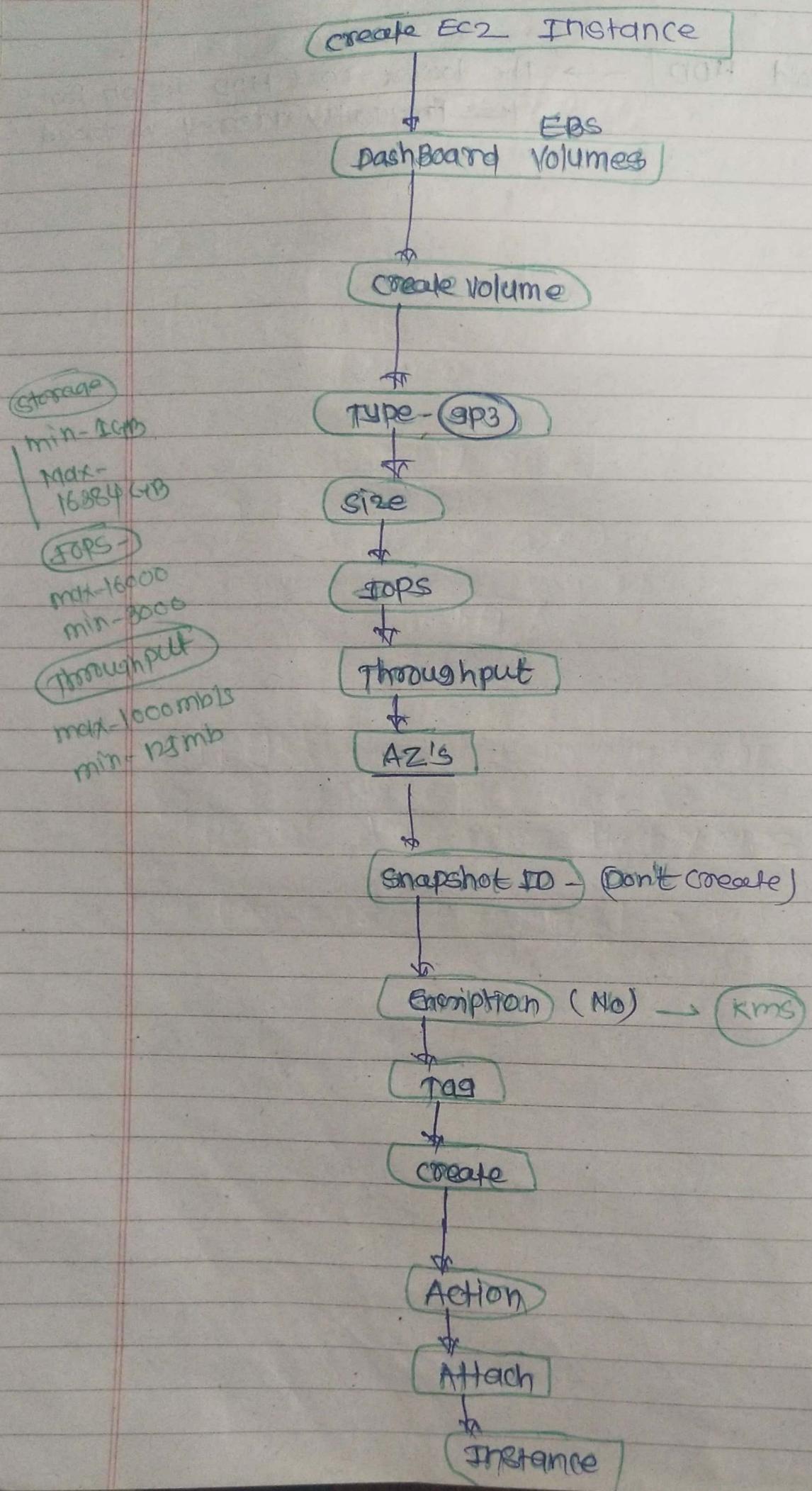
TYPE → IOPS1, IOPS2
- ③ throughput optimized HDD → A low cost HDD designed for frequently accessed, throughput-intensive workloads

- ① SSD - solid state drive  
② HDD - Hard disk drive  
③ previous generation volume.

PAGE NO. \_\_\_\_\_

DATE: \_\_\_\_\_

- ④ [cold HDD] → The lowest-cost HDD design for less frequently accessed workload



Instance

device Name

Select Form  
(SDF - TO: SDP)

Attach Volume

Now,

Go to instance connection

use following command step by step.

① `lsblk` → To check all block / volumes

② `df -h` → To check volume attach or Not

③ `file -s deviceName` → To check data available or not  
Ex /dev/xvdp  
(data) - mens no data

④ ~~mkfs -t ext4~~

`mkfs -t ext4 deviceName` → To make file system

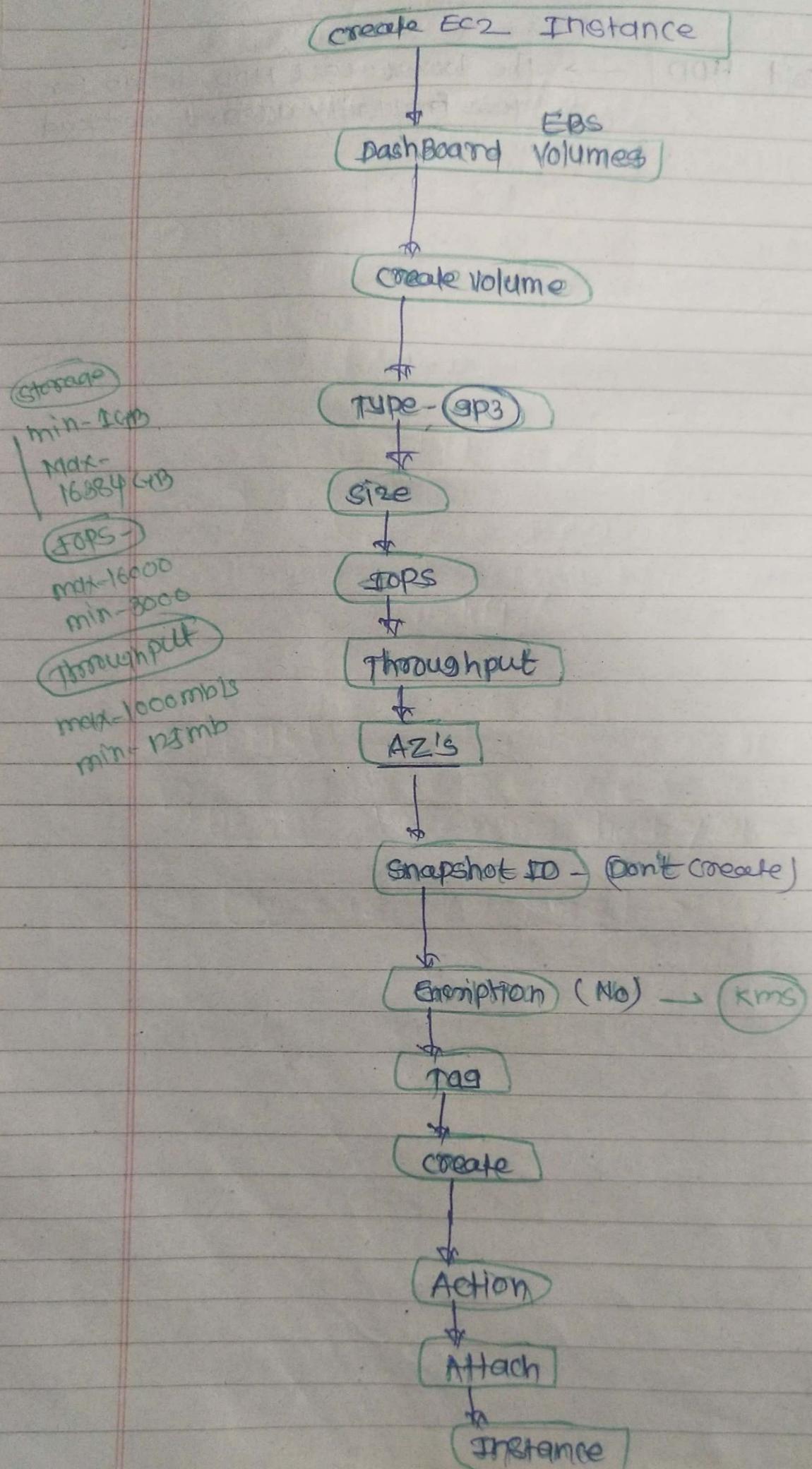
⑤ `mkdir directory` → Make directory  
Ex /mnt/EBSvolume1 for mount EBS

⑥ `mount deviceName directory` → mount EBS volume

⑦ `df -h` → check mount or not

For permanent Attach

only after ⑧ `nano /etc/fstab` → open write  
termination  
EBS volume  
detach.  
Ex. `device name` `directory` `File system` `defaults, o o  
/mnt/ebsvolume1 ext4 nofail`



Instance

device Name

Select Form  
(SDF - IO, SDP)

Attach Volume

Now,

Go to instance connection

Use following command step by step.

① `lsblk` → To check all block / volumes

② `df -h` → To check volume attach or not

③ `file -s deviceName` → To check data available or not  
Ex /dev/xvdp

(data) - mens no data

④ ~~mkfs -t ext4~~

`mkfs -t ext4 deviceName` → To make File system

⑤ `mkdir directory` → Make directory  
Ex /mnt/EBSvolume1 for mount EBS

⑥ `mount deviceName directory` → mount EBS Volume

⑦ `df -h` → check mount or not

For permanent Attach

only after termination  
EBS volume detach. ⑧ `nano /etc/fstab` → open write

device name	directory	File System	defaults	No fail
Ex. /dev/xvdp	/mnt/ebsvolume1	ext4	0 0	

(a) `mount -a`) → For permanent volume attach

(10) `umount directory Name` → For detach the  
ex-`/mnt/ebsvolume` volume

(A) Now,

Go to volume

Action

modify

increase size

modify

Not possible to  
decrease

(11) `lsblk` → check increase size of block or not

(12) `mount device name directory Name` → mount

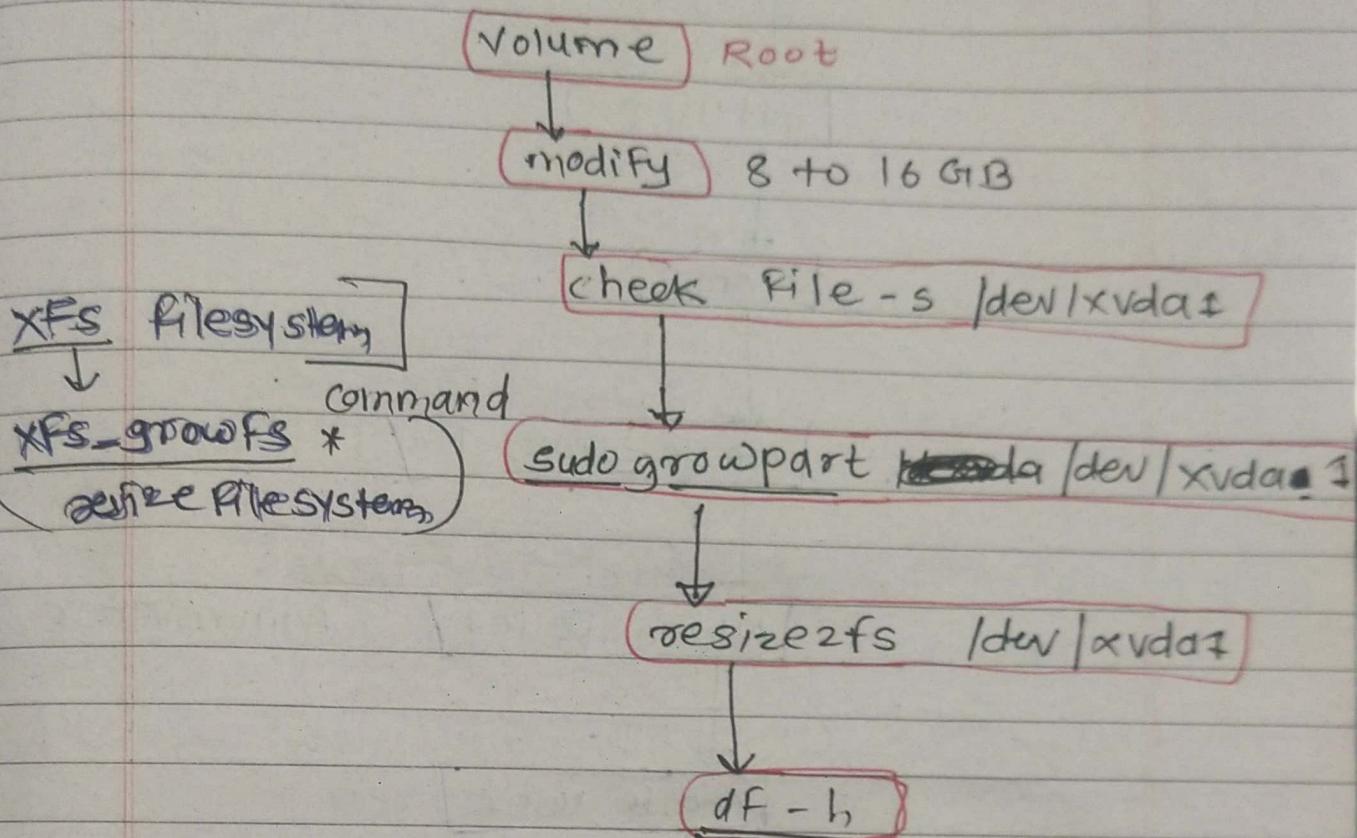
(13) `resize2fs device name` → Resize your volume

## Systemctl daemon-reload

PAGE NO.

DATE

### \* Root Volume \*



### \* Multi-Attach \*

single EBS to  
multiple instances

- \* it is possible but not in general purpose SSD
- \* it is possible only provisioned IOPS SSD  
only IOP

- \* Two Hypervisors in AWS - xen hypervisor  
Nitro hypervisor

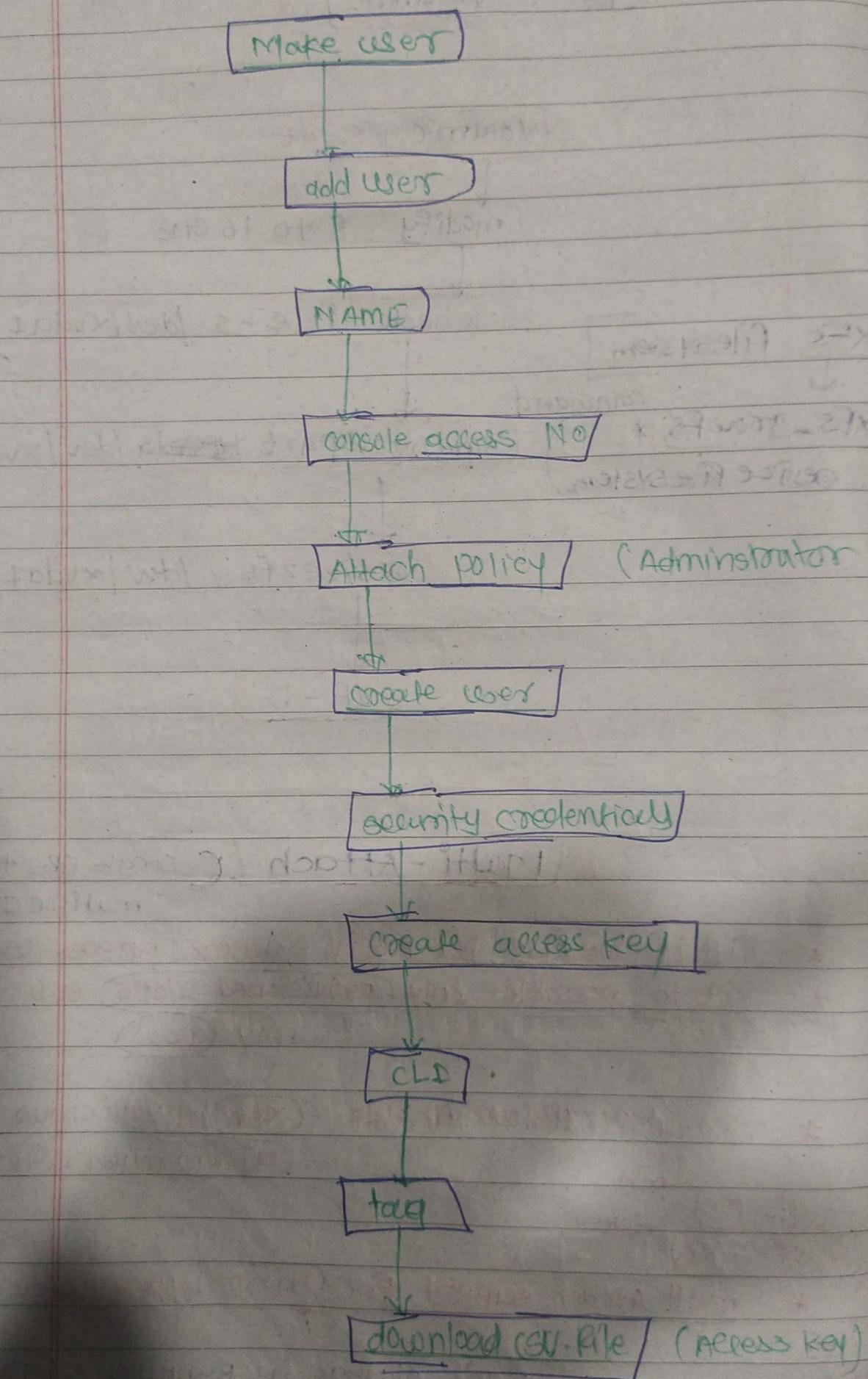
Instance type -  
 Action  
 Instance setting  
 ① xen  
 ② Nitro

\* multi Attach support for Nitro hypervisor.

- \* Instance type can't change in running condition  
You first stop than we can change instance type.

# AWS - CLI

PAGE NO. \_\_\_\_\_  
DATE \_\_\_\_\_  
Command Line Interface

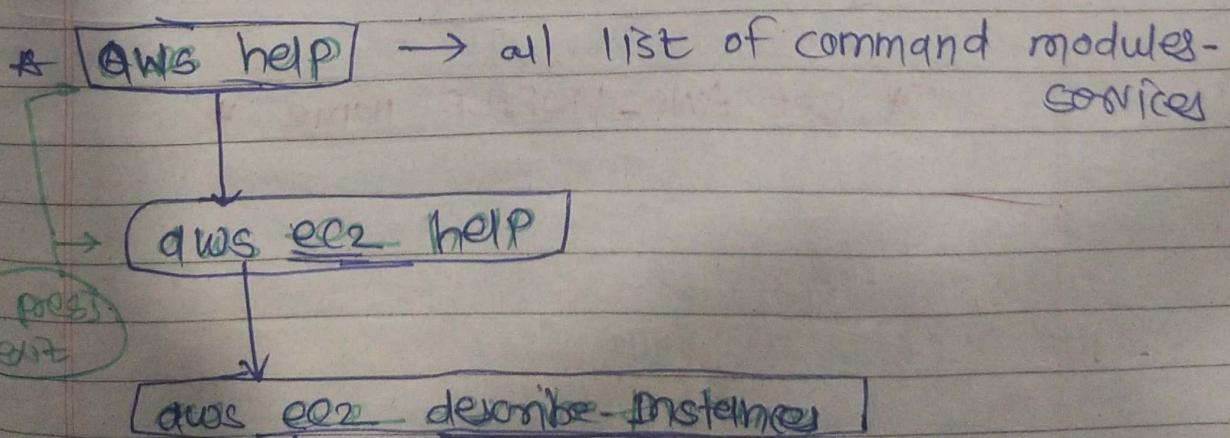
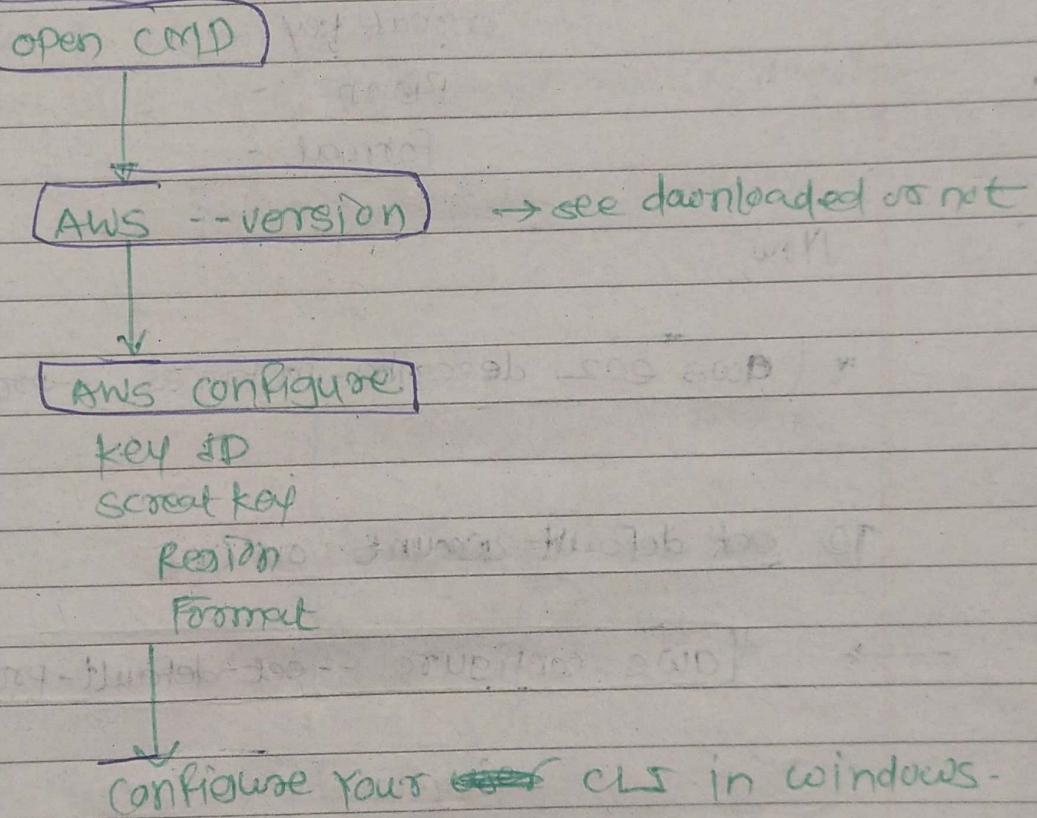
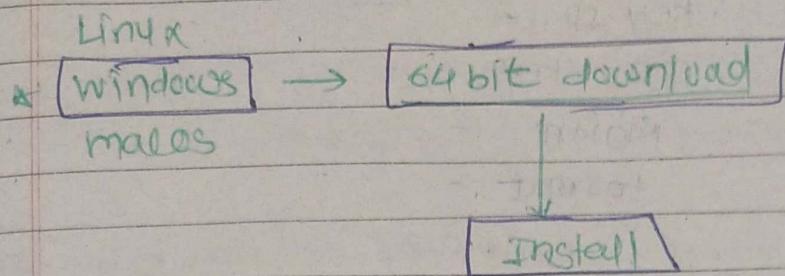


ctrl+shift+b - linux  
page

PAGE NO.

DATE:

## AWS CLI documentation



## \* AWS CLI Handle multiple AWS Account

→ **aws configure --profile Account 1**

key ID -

secret key -

Region -

format -

→ **aws configure --profile Account 2**

key ID -

secret key -

Region -

format -

Now

\* **aws ec2 describe-instances --profile=Account**

To set default account on CLI

→ **aws configure --set-default-profile profile name**

OR.

**AWS DEFAULT PROFILE = account name**

\* **Set AWS\_PROFILE Name \***

S3

# Simple storage service

\* Infinite data store

↳ Max 5GB File/Object

\* Storage the file, image, document, video from a machine and access them over the internet.

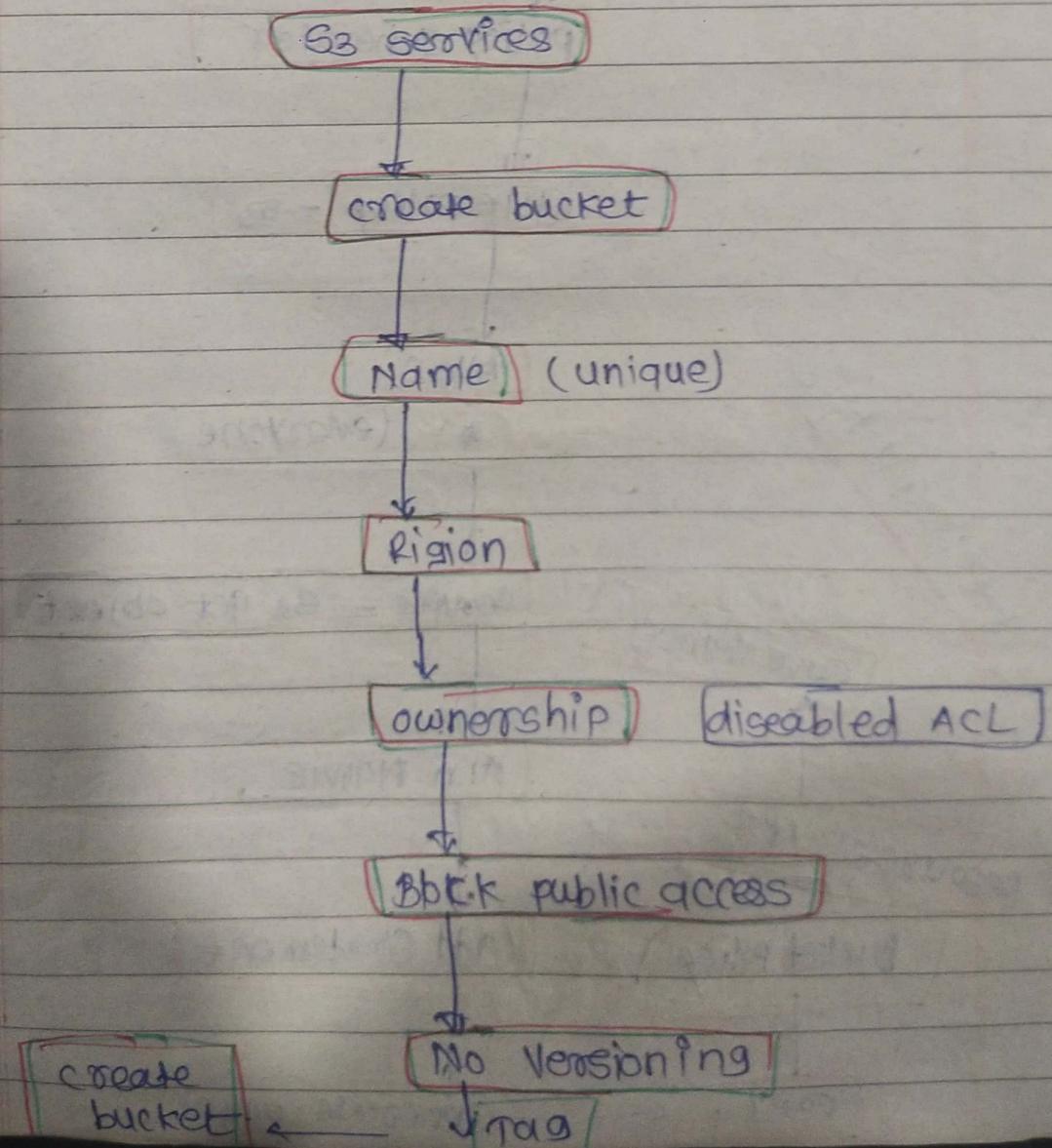
\* also use as backup

\* bucket is created in Region / S3 is Global specific

\* data is Region specific

\* bucket type → ① General purpose ② directory  
(onezone storage class)

## How to create bucket

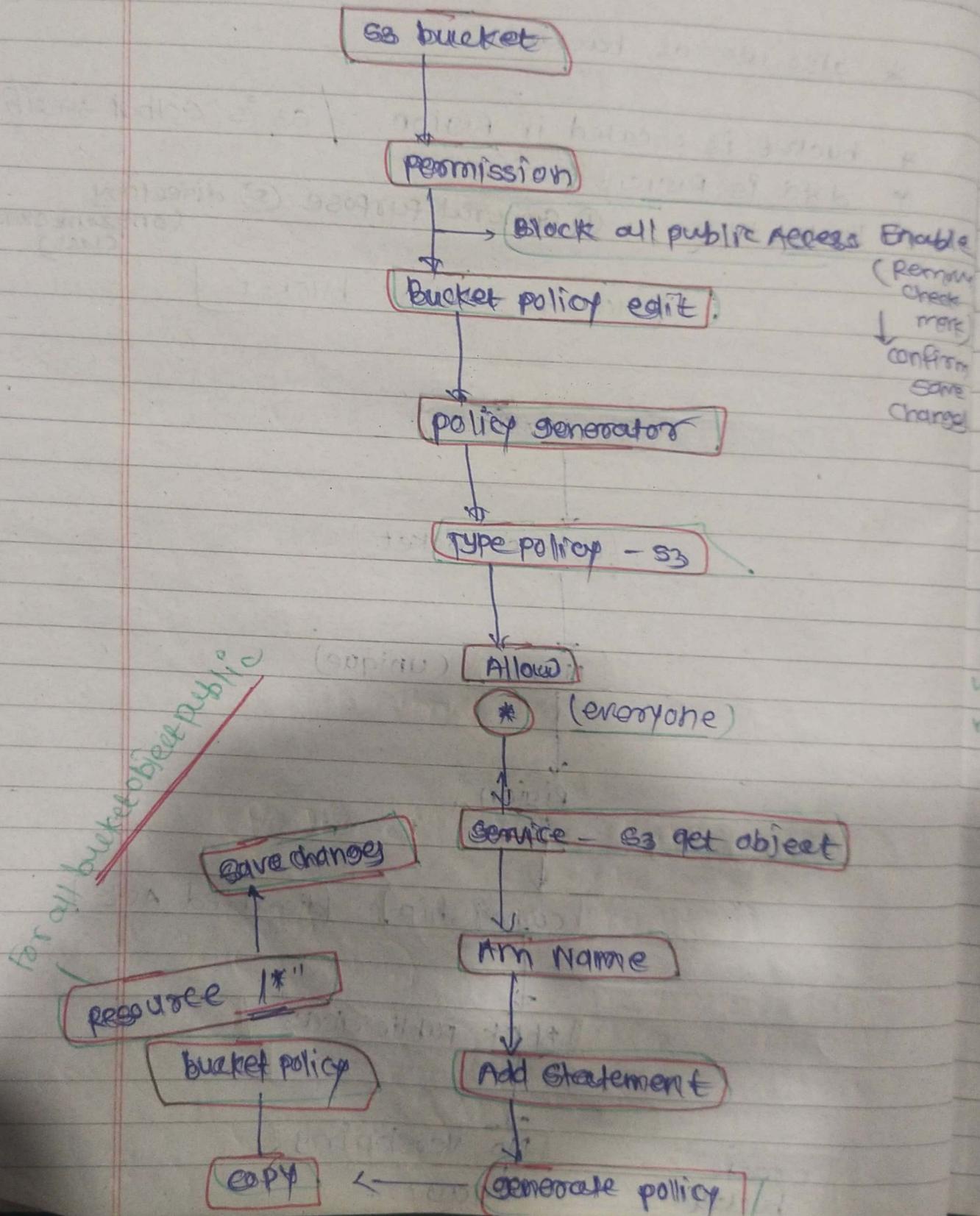


Name → 3 to 63 letter  
• No IP  
• First letters

PAGE NO.  
DATE

\* File is called as object

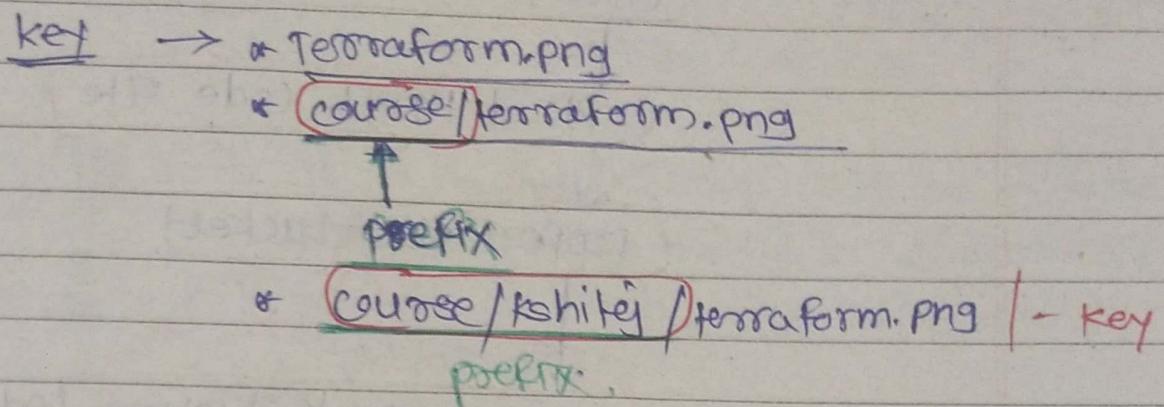
[ public Access ]  
public your Bucket object



## (S3 object properties)

- \* Give tag
- \* Give metadata

When you upload data



\* We can't create folder in S3 bucket

## \* (Versioning) \*

(Versioning suspended)  
when future version  
not created)

S3 bucket

create folder

file upload - f.png

properties

Bucket versioning

Enable

Save

delete

f.png

↓

only delete mark add

↓

show versions

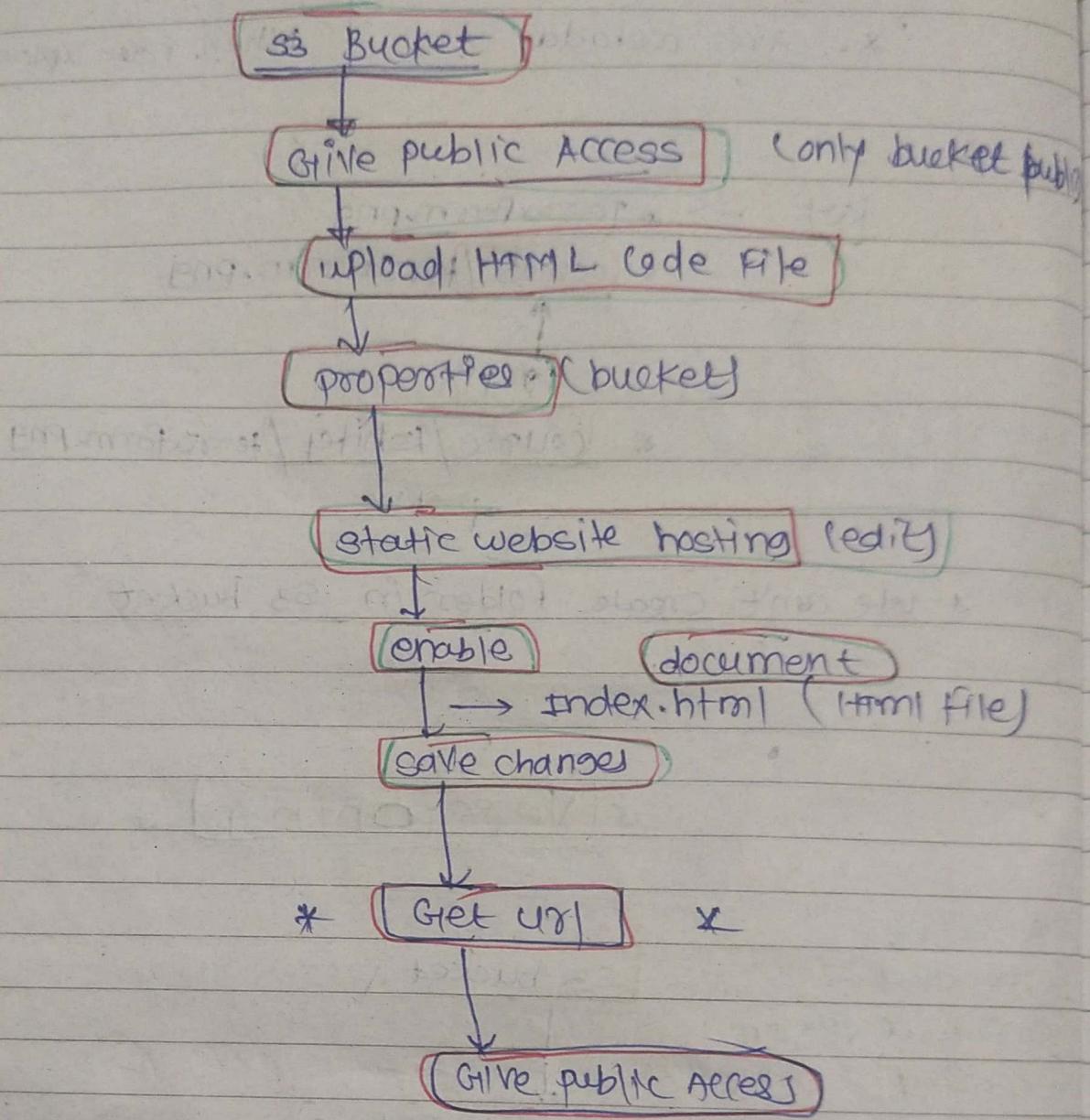
↓ delete (permanently delete)

version ID

f.png → \*\*\*\*\*

f.png → null

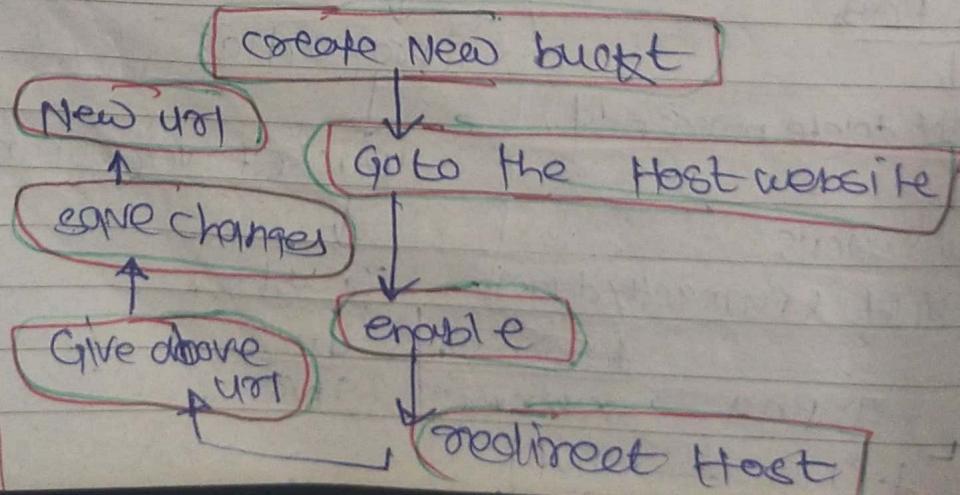
## (Host Static Website)



## Redirect Host static website

Form  
above  
Step  
after

HTTP  
Remove  
only  
Host  
id



# VPC → peering connection

PAGE NO.

DATE:

- Instance connection types -
- ① ssh via public IP
  - ② SSM (Session manager)
  - ③ NAT Gateway
  - ④ endpoint

\* **endpoint** → Using endpoint whole subnet Allow access to go over the internet.

\* **peering** → it is a channel that use by different instances which belong to different VPC's to talk to each other using their private IP.

- \* Transit not Allow overlapping
- \* Not allow overlapping VPC
- \* end-to-end connection

- ① **VPC** create (two) in different
- ② create subnet (two in different VPC)

- ③ Create two instance using different VPC & their subnet

- \* security group create ~~not possible~~
- \* add rule - ICMP (0.0.0.0/0)

- ④ **Create endpoint** for connection

(Create endpoint)

Name

ECS Instance connection

Security group (of instance)

VPC

Subnet

→ (Create)

## ⑥ peering connection

Create peering connection

Name

Add requester VPC

my Account / other account

Region / same / other Region

acceptor VPC

Create peering connection

Click on Peering

Accept request

from Auction

button

⑥ Both subnet Route table Entry we

Both subnet  
Peer connection  
Entry RT

Console Engine

Rule - IP Network CIDR Entry other  
subnet Route table

Custom Fwmp (0.0.0.0/0) - (same for both)

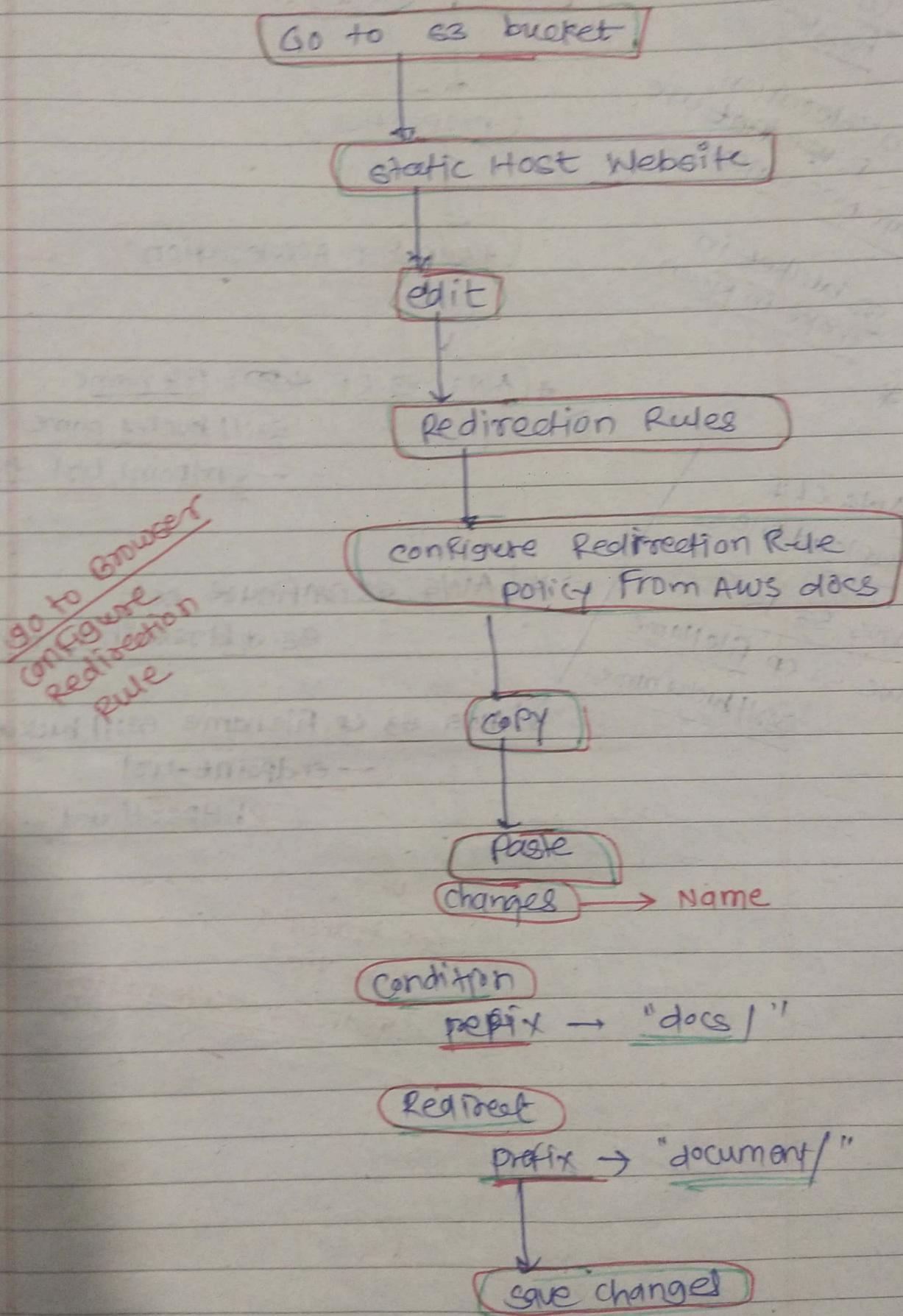
⑦ Both instance security group rule add

⑧ Now check connection established

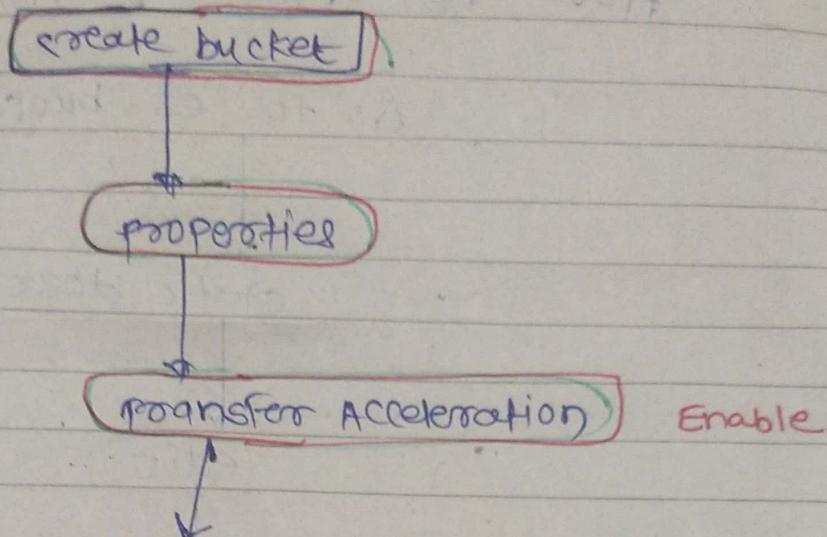
(ping acceptor instance IP) PRIVATE

53

## How to change prefix in static website



## How to enable Accelerated transfer | How to



- \* download/upload faster
- \* acceleration is not use AWS console
- \* S3 bucket in make in Ribbon
- \*

**AWS S3 CP ~~HTTP~~ File name**

s3://bucket name  
-- endpoint URL Accelerate endpoint

**AWS S3 Configure set**  
S3 addressing style virtual

**AWS S3 CP filename s3://bucketname**  
-- endpoint URL  
https://url

\* For transfer file via Transfer acceleration

AWS CLI  
upload file s3

\* AWS S3 ls

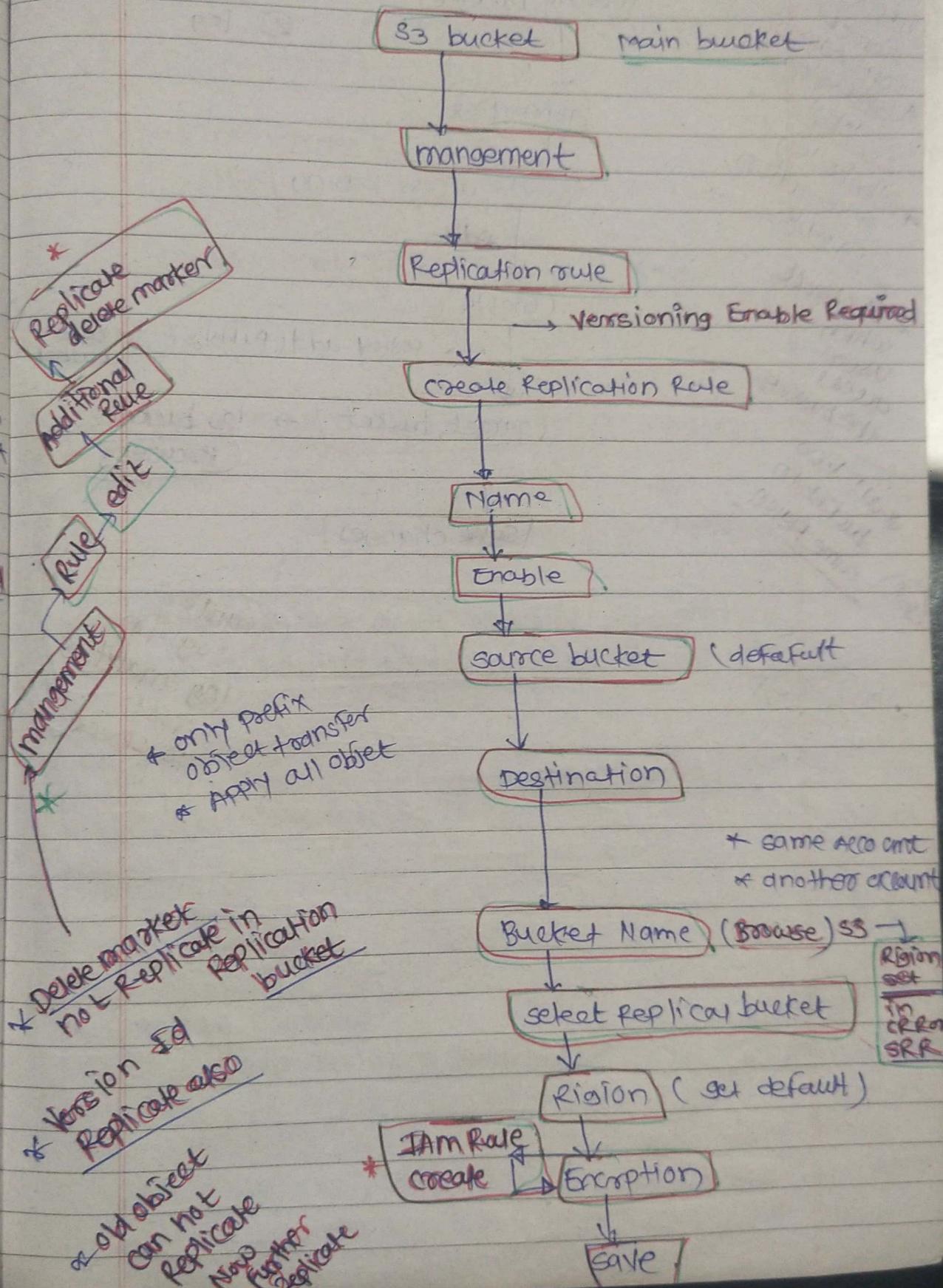
\* AWS S3 CP File Name

\* AWS S3 CP File Name

\* s3://bucket name

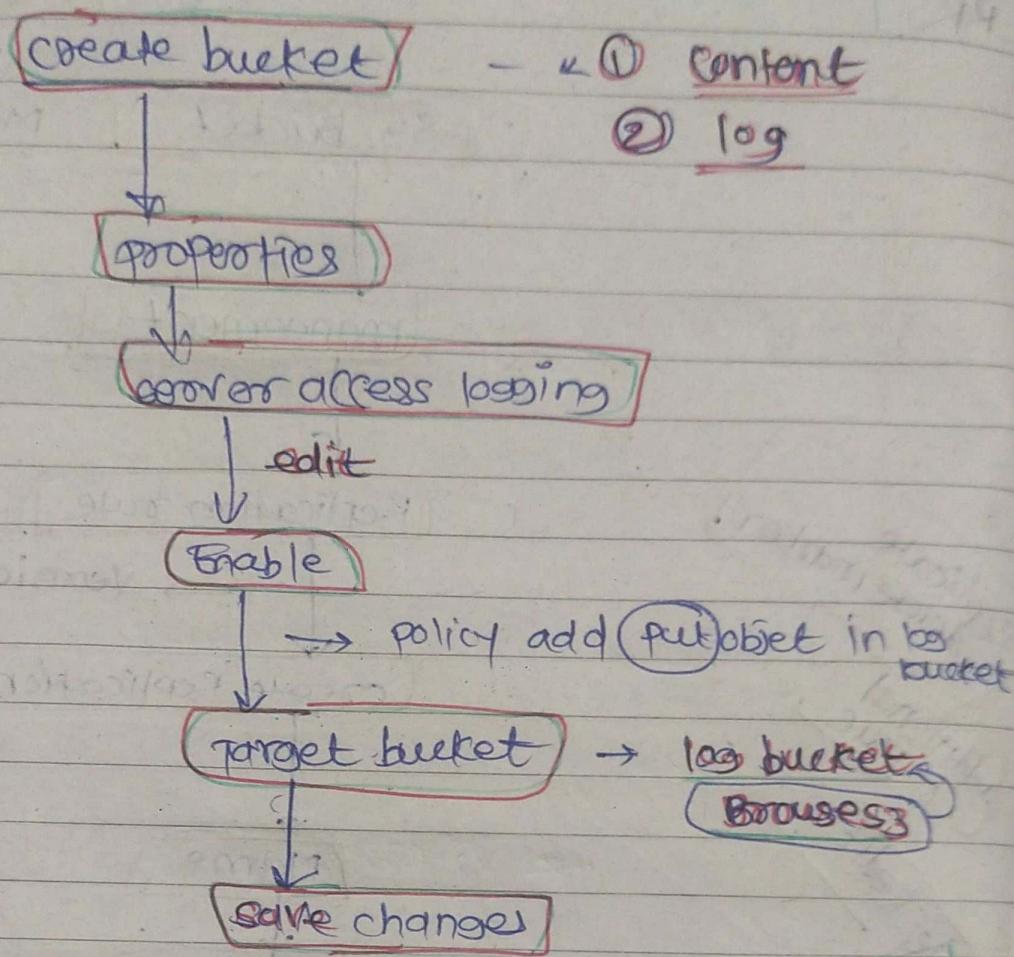
SRR  
CRR

## Same / cross Region Replication



## Configure logging in S3 Bucket

- \* user S3 object operation perform notebook
- \* their log generation enable
- \* to check which user access the bucket
- \* all two bucket in same region



Transfer log in log analytics tool to check

class depends on  
Availability  
durability

## \* S3 storage classes \*

frequent  
Access  
data  
use

S3 Standard

Availability	Durability	ARZ	change
99.9%	99.9%	$\geq 3$	N/A

S3 intelligent  
Tiering

Storage duration	NA
NA	NA

S3 Standard - FA

(Infrequent Access)

Storage duration	30 days
$\geq 3$	128 KB

S3 one zone - IA

Storage duration	30 days
1	128 KB

cost  
minimum  
per 1000  
reqs

S3 glacier

Instant Retrieval

Storage - go	30 days
day	128 KB

\*

S3 glacier

Flexible Retrieval

99.9

99.9%

Storage - go	30 days
--------------	---------

\* Latency  
= min/hour

\*

S3 glacier

Deep archive

99.9

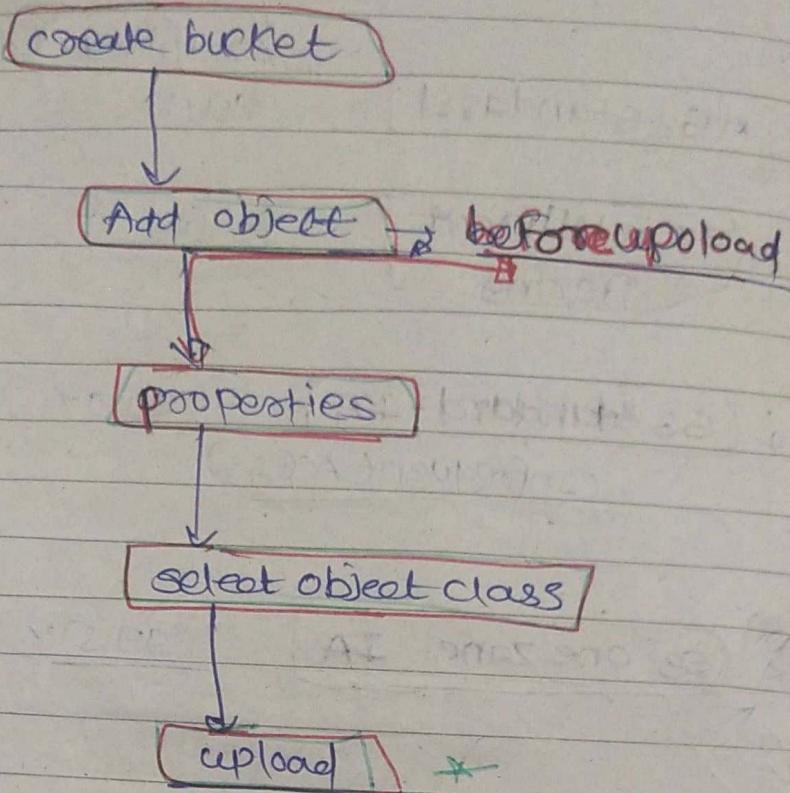
99.9

Storage - 180 days	NA
--------------------	----

\* Latency -  
hour

## Configure storage

\* at bucket level we can't select storage classes  
 \* at object level select storage class



\* move automatically other storage base class on increase

cost efficiency and performance with AWS intelligent Tiering for S3

standard

default setting AWS

standard IA

30 days

No Retrieval

90 days

glacier S3 instant retrieval

monthly

120 days

glacier flexible Retrieval

create

bucket

Properties

deep Achieve access

180 day

Achieve Access tier

90 days

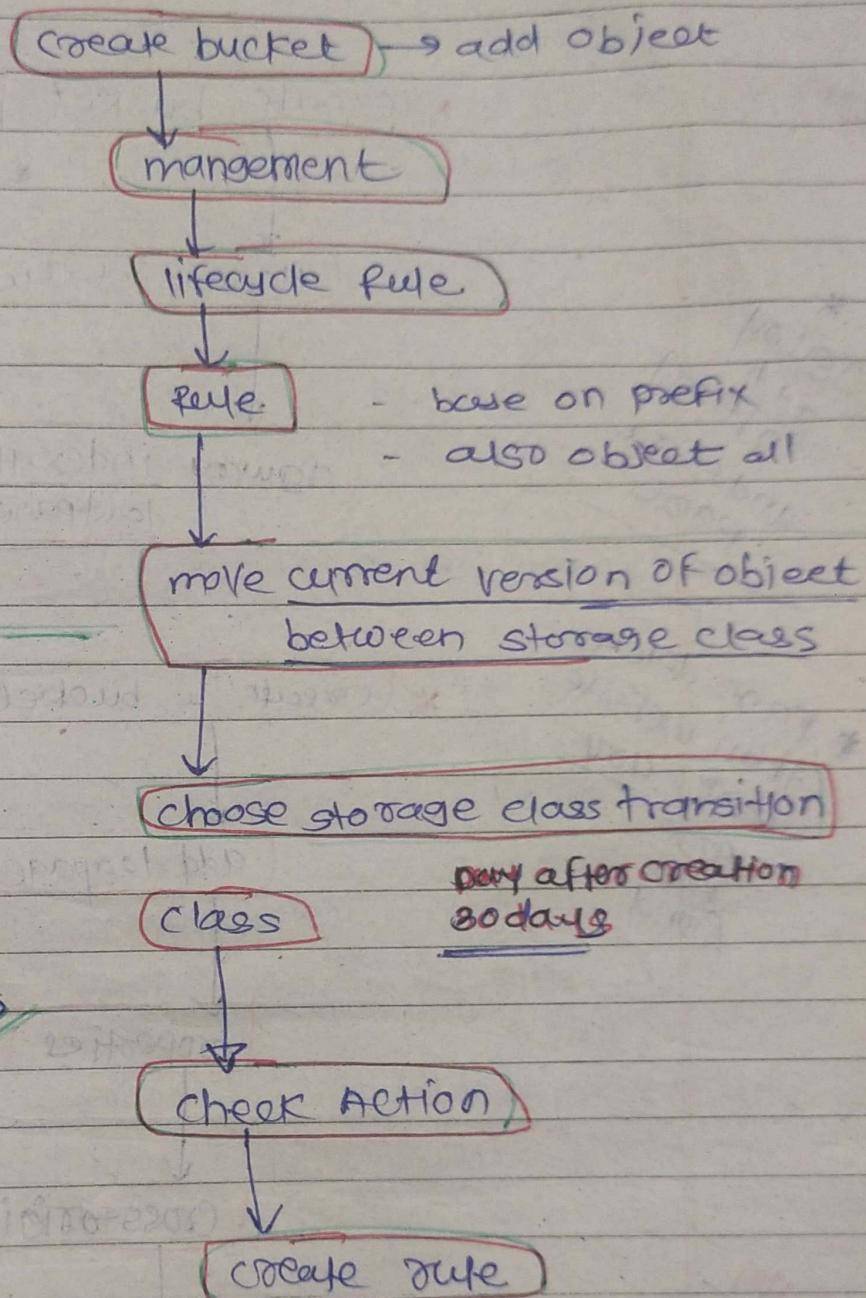
Intelligent-timing Archive

Name

configuration scope

# [ Data ]

## simplify lifecycle management with AWS



~~single operation~~  
 1 at a time  
 many  
 action perform

\* Valid days based on class

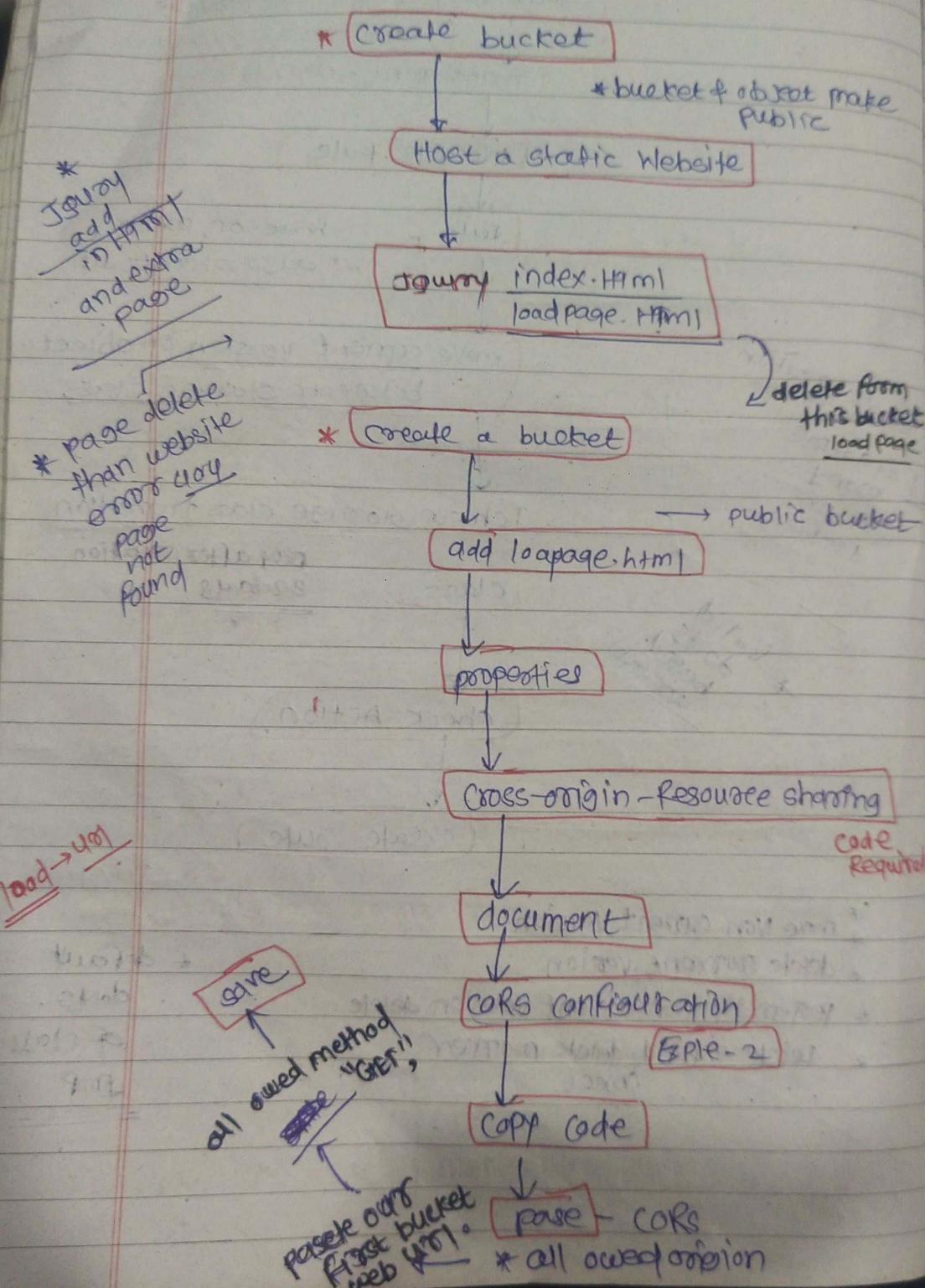
day after creation  
30 days

- \* move Non current version
- \* delete current version
- \* permanently Non current version delete
- \* delete expired, delete marker object

← default days of class 30mp

## \* cross-origin-resource-sharing \*

### \* What is CORS and How to enable \*



- ① Compute
- ② Storage
- ③ Data

\* Based on

Compound

on hr base

## \* AWS Pricing \*

① Compute

② Storage

③ Data transfer (Download or upload from outside)

+ Based on above three types to charge.

~~on hr base~~ Compound - on demand Instance → ~~roung basecharge~~  
1 min first charge  
after sec charge

[ Ex + 20sec we

charge - 1min

\* 80sec we

charge - 80sec

- reverse instance → upto 70% discount

(OS  
Type  
Region)

based convertible  
max discount.

- spot instance → depend on bid

upto 90% discount

\* Reliability of instances are too less.

- dedicated Host → particular to us

## \* Command Line Interface \*

AWS → CLI → Windows - 64 bit download

\* AWS → File created

c:/users/Marko/.AWS

① config →

All default & other account detail

② credential →

All user key present

Kshitij  
User name - Kshitij  
Password - KshitijSS@

\* [For user create & Attach policy] \*

① aws iam create-user --user-name kshitij

② aws iam list-policies Policy Arn copy

③ aws iam attach-user-policy --user-name \_\_\_\_\_ --policy-arm \_\_\_\_\_

④ aws iam create-login-profile --user-name \_\_\_\_\_ --password \_\_\_\_\_

\* [For check profile list] \*

① aws configure list-profiles

## \* For create New profile \*

- ① aws iam create-user --user-name —
- ② aws iam list-policies
- ③ aws iam attach-user-policy --user-name —  
— policyarn —
- ④ aws iam create-access-key --user-name —
- ⑤ aws configure --profile username  
provide access key & secrete accesskey

## \* For default profile set \*

- ① set AWS\_PROFILE Name

## \* Presigned URL in S3 \*

- S3 object share with user
- generate URL valid for some days / some hour
- particular time is completed URL is not work.
- Without public object We can share object to the client

Create object

↓  
Upload object

↓  
Select object

↓  
Object action

↓  
Share with presigned URL

↓  
Time interval for presigned URL  
expire

↓  
min / hour

↓  
Create presigned URL

↓  
Copy - presigned URL



PAGE NO.  
DATE:

PAGE NO.  
DATE:

## \* Securing your AWS S3 Data \*

At rest, server-side, client-side Encryption

\* HTTPS/  
HTTP

lays) some hours  
not cost.  
bject to the Client

(Type of encryption)

\* (Version Enable)

Amazon S3 managed key

default

SSE-S3

(server-side

encryption)

\* AWS Key management

Service Key (SSE-KMS)

\* Dual-layer, server-side Encryption  
with KMS

\* Bucket key - Enable

not supported

Create bucket

(Bucket Encryption)

Name

Properties

Encryption

S3 managed key

Bucket key - enable

Create bucket

When KMS encryption is used to encrypt objects  
in the this bucket, the Bucket key reduces  
encryption costs by lowering calls to AWS key.

ed URL

presigned URL  
expire

URL

URL

# CloudFront

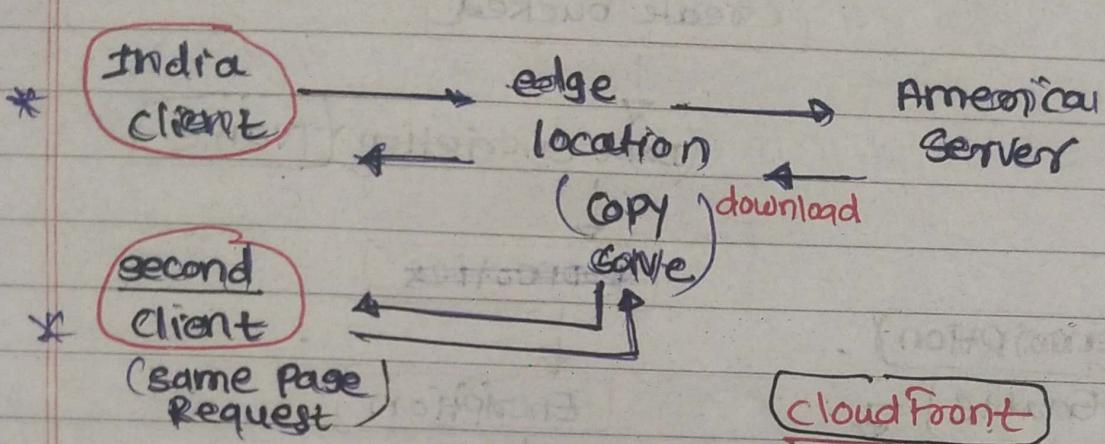
PAGE NO.

DATE:

Content delivery Network

It's a  
cache

- \* It's an edge delivery network.
- \* Server is located in America and Host a website.
- \* From India a client access the website they face some latency to access the webpage.

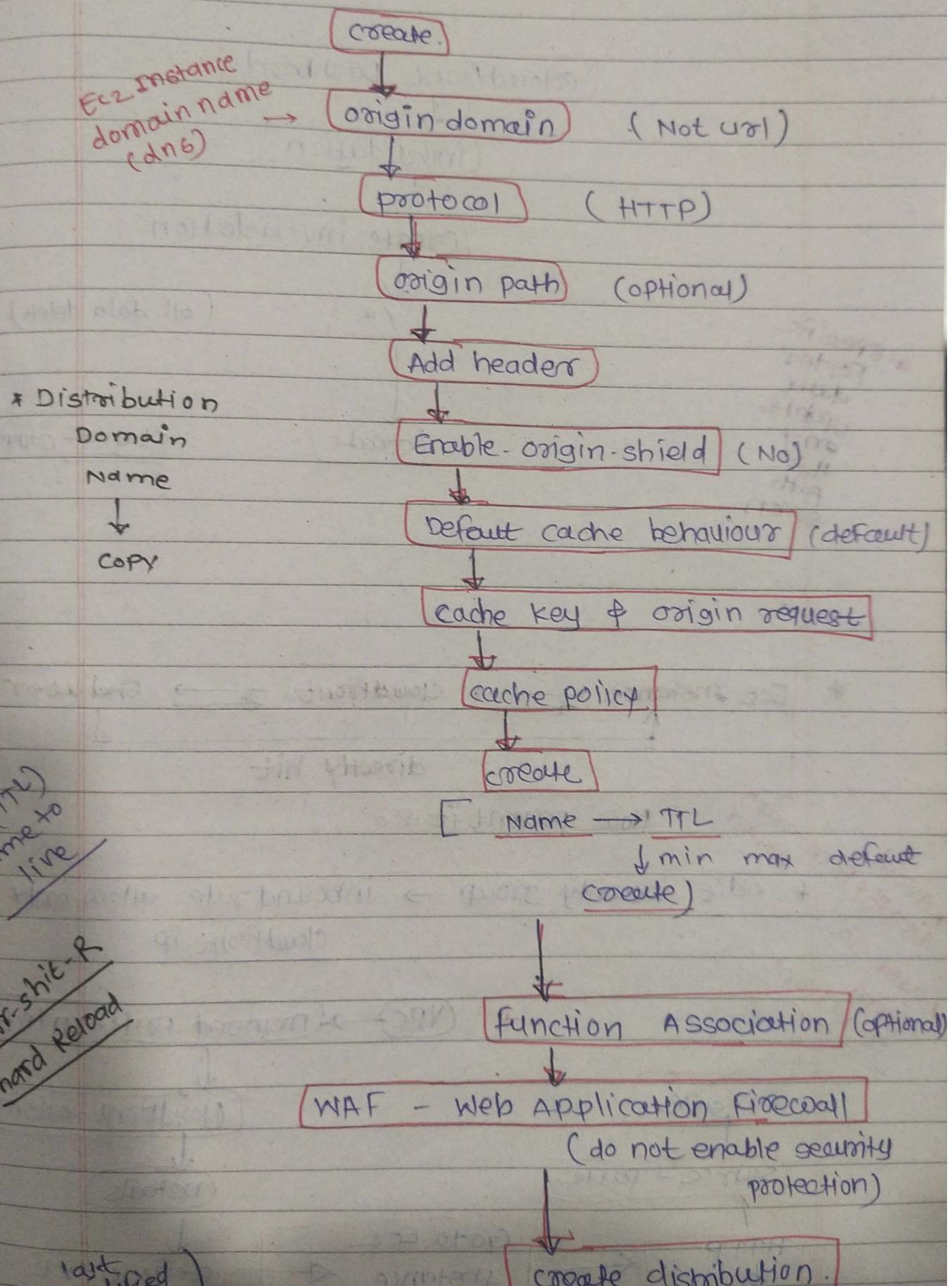


## CloudFront

- Form edge location
- First Byte come form main server cloudfront
- transfer the data at a same time

(make nginx server with EC2)

## \* Create cloudfront distribution with EC2 \*



\* miss from CloudFront  
\* hit from CloudFront

DATE:

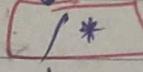
(Nov 10) + 30

- \* Main server update but CloudFront cache are old in this condition we delete our CloudFront Cache.

### CloudFront Dashboard

#### invalidation

##### create invalidation

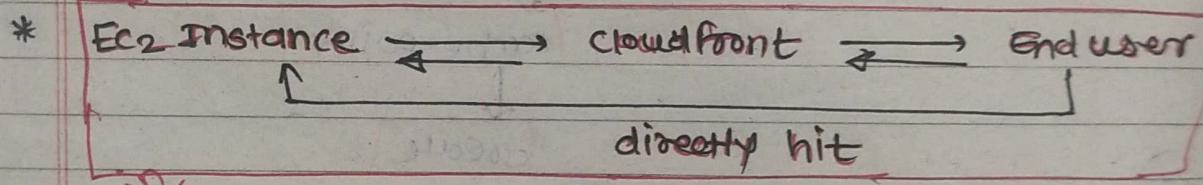


(all data deleted)

##### create

Status - Complete

\* Specific folder data update only this path given



How prevent it ??

→ avoid directly from main server  
→ user go from CloudFront.

\* edit security group → inbound rule allow only CloudFront IP

VPC → managed prefix list

Source - Paste  
save

HTTP  
Post - 80

Goto EC2 instance  
security group

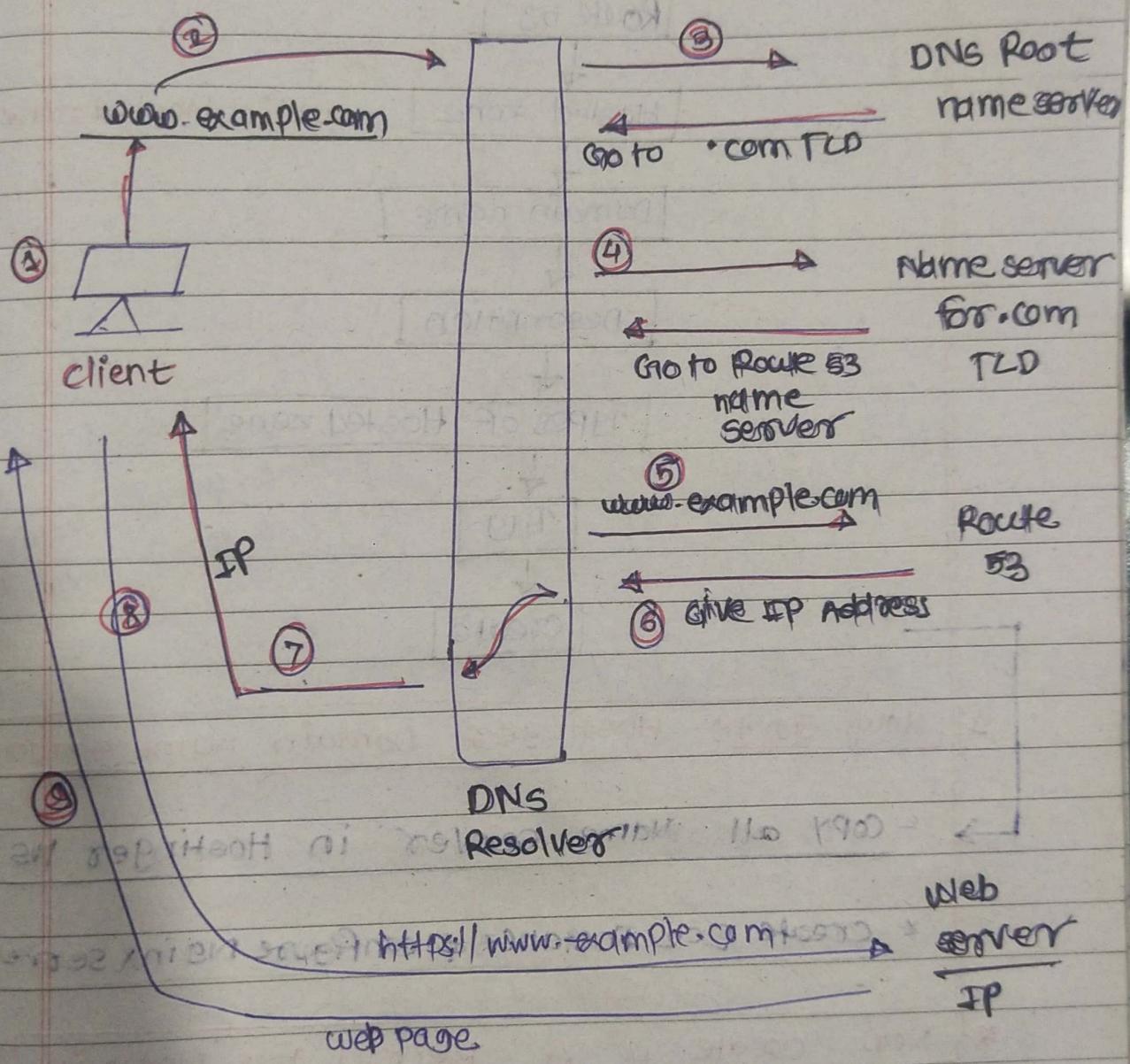
CloudFront-origin

details

prefix list  
(copy)

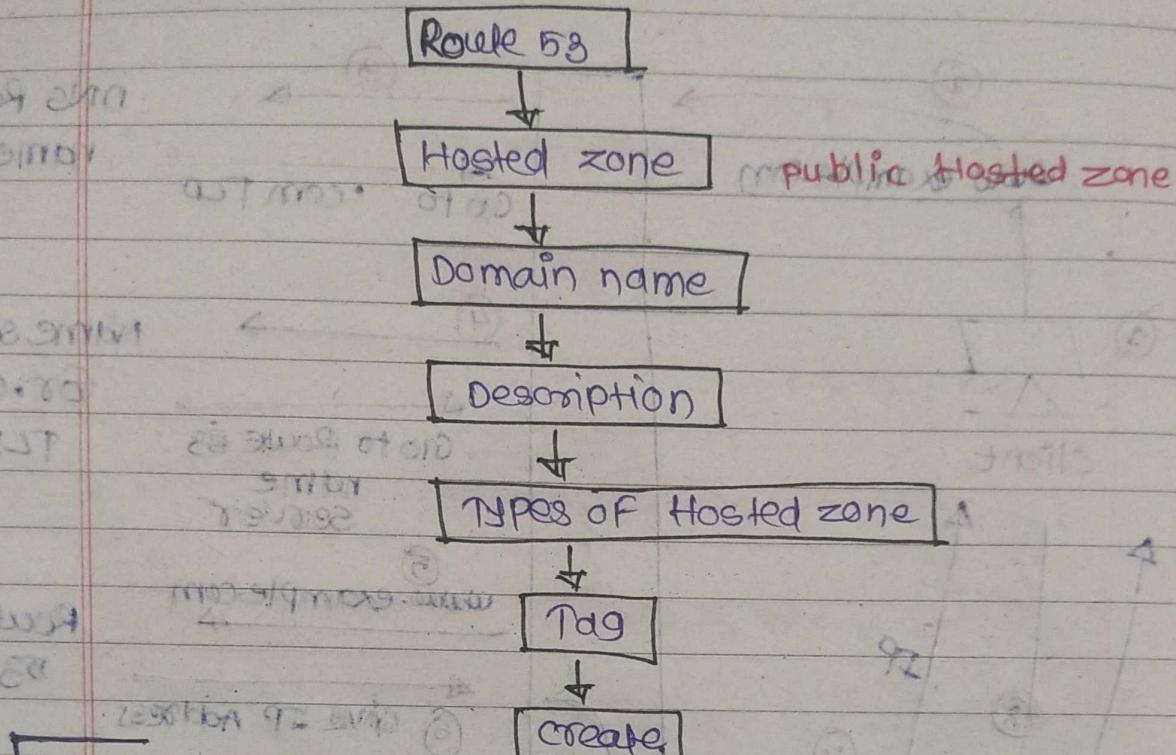
# Route 53

- \* It work on DNS.
- \* Route 53 function → ① Register ② Routing ③ health check.



\* NS Record → Traffic from Domain Provider to AINS

\* In Route 53 : Creating Hosted zone

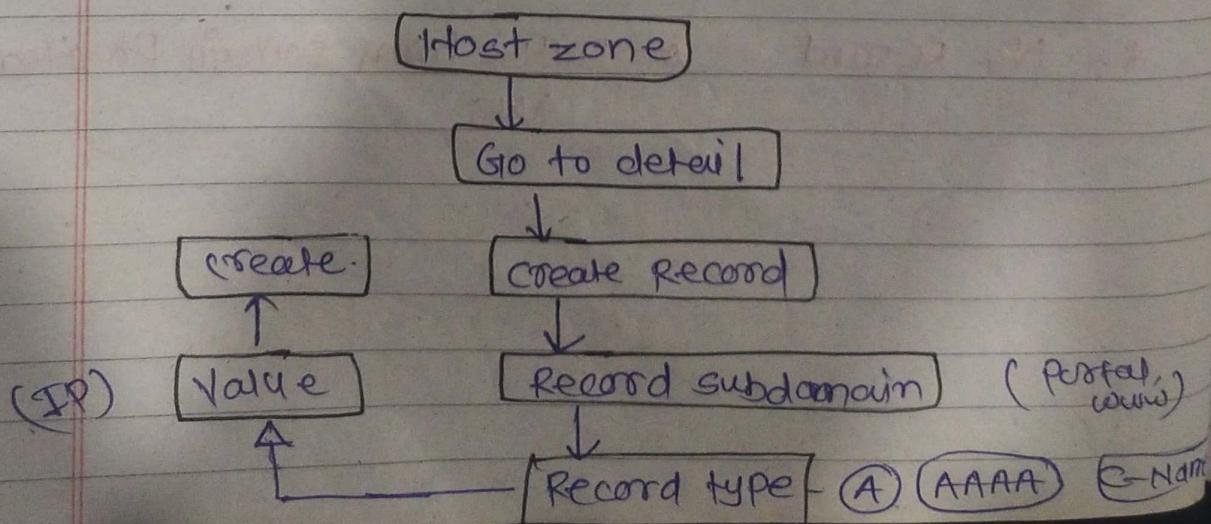


\* Now, Go to Hostinger Domain name servers

→ \* COPY all Name servers in Hostinger NS

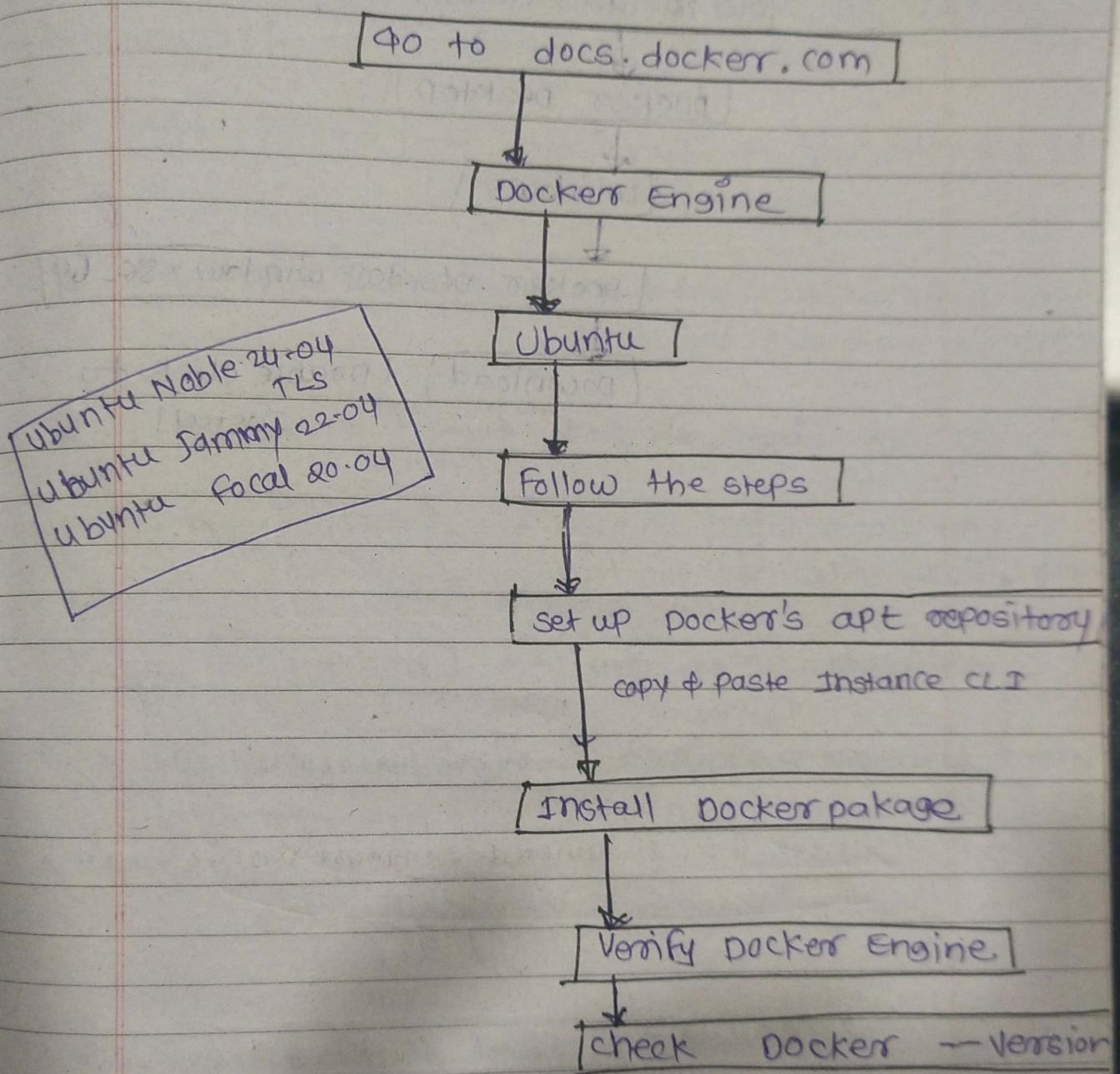
\* Create EC2 instance & configure Nginx server

\* Now create a A Record for Domain



# Now Docker Installation

For Linux (Ubuntu) Instance.



## For Windows Docker installation

Go to docs.docker.com

Docker Desktop

windows

Docker Desktop windows x86\_64

download

(Double click to  
Install)

## installation/

TTL time to live

PAGE NO.  
DATE:

### A-Name & AAAA- Name Record in action

\* **A-Record** → Route traffic to an IPv4 address and  
Domain converted in IP Some AWS Resources.

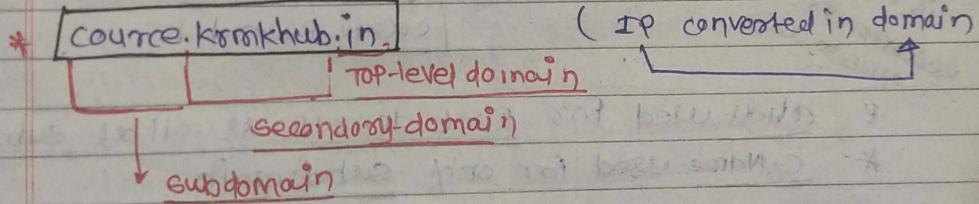
\* **AAAA-Record** → Route traffic to IPv6 addresses and  
Some AWS Resources

\* **C-NAME Record** → Route traffic to another domain  
name and AWS other resources

\* **MX Record** → specifies mail servers

\* **TXT Record** → used to verify email senders  
and for application specific-values

\* **PTR Record** → maps an IP address for a  
domain name



\* S3 static website traffic access from your domain  
using C-Name. (Your bucket name & your subdomain  
name is same)

\* C-Name  
used for  
subdomain  
not for  
root domain..

Go to Record Create

↓  
Subdomain

↓  
C-Name

↓  
Given S3 static url

nslookup domain name → give info

nslookup -type=CNAME domain name → give info

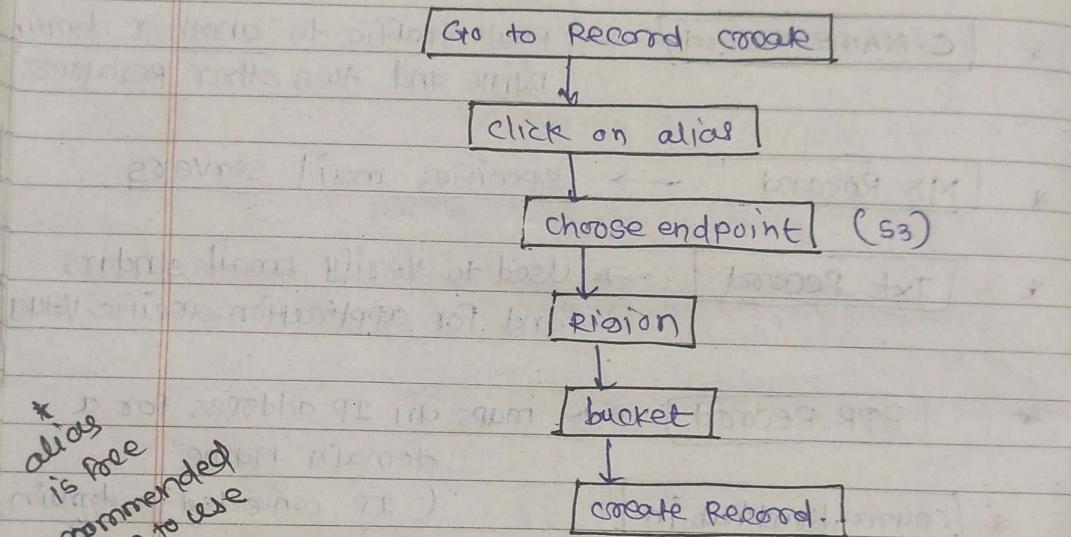
PAGE NO.:  
DATE:

\* SSL certif

### \* Route-53 - alias in Action alias v/s C-Name

\* without subdomain for s3 static Host website access using domain (Root) (not possible via CName)

\* it is possible via alias. (You create bucket with name Root domain)



- \* alias used for Root domain as well as Subdomain
- \* C-Name used for only Subdomain

### \* Route-53 - TXT Record in Action

- \* u can not access the TXT Record not browser.
  - \* For this access only command (Programmatically)
  - \* nslookup -type=txt domain
- \* Show value value u set.
- 
- ```
graph TD; A[Go to Record create] --> B[subdomain]; B --> C[TXT Record]; C --> D[Value]; D --> E[Create]
```
- The flowchart illustrates the process of creating a TXT record in Route-53. It starts with 'Go to Record create', followed by 'subdomain', 'TXT Record', 'Value', and finally 'Create'.
- Annotations on the right side of the flowchart include:  
\* At least one  
\* health check  
\* Report

give info  
PAGE NO.  
DATE:

\* SSL certificate → To check authorised person of domain  
use TXT record. (TXT Report)

alias v/s C-name

Host website accessible via C-name

Create bucket with name Root domain

\* Route 53 - PTR Record

- \* In this record it is totally against the A-Record
- \* IP address converted in domain
- \* in A Record domain converted in IP

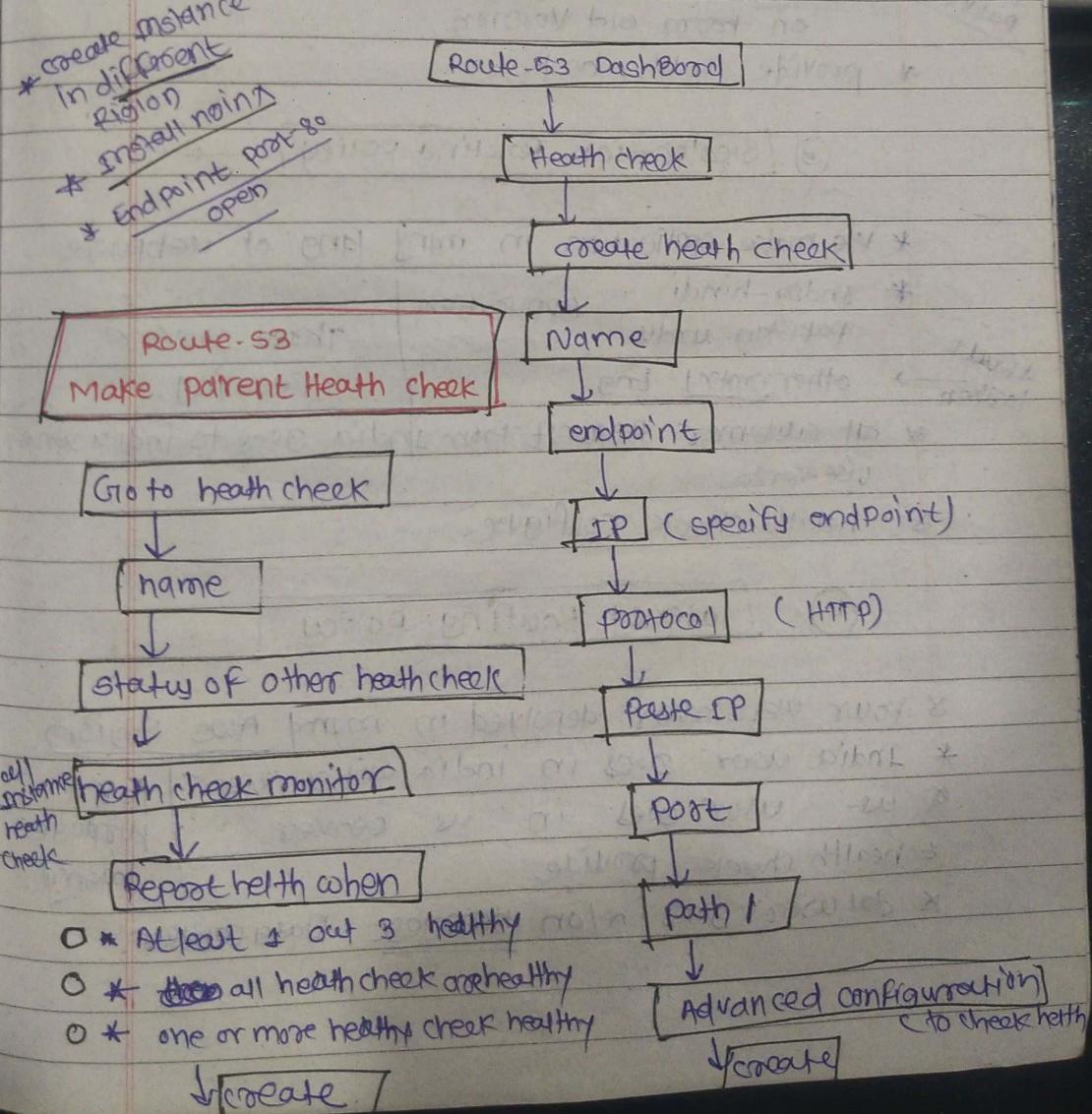
\* use for Monitoring

\* Create instance in different Region

\* Instal point

\* Endpoint port 80 open

Create Health Check in Route-53



## Route-53 → Routing policy

### ① [simple Routing policy] →

- \* In this policy we can define many ip addresses.
- \* No and health check.

### ② [weightage Routing policy] →

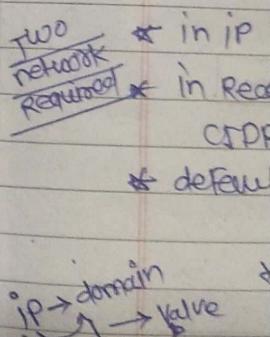
- No compare with local balancing*
- \* We can't distribute our traffic.
  - \* this policy used for old Version + New Version slowly we distribute our traffic to new Version from old Version
  - \* provide health check.

### ③ [Geolocation Routing policy] →

- default location*
- |                     |             |                       |
|---------------------|-------------|-----------------------|
| * India-hindi       | Application | Three server running. |
| Pakistan- urdu      | mpn         |                       |
| other country - Eng |             |                       |
- \* all customer request from India goes to India server vice versa.
  - \* health check configure.

### ④ [latency Routing policy]

- \* Your webserver deployed in many AWS Region.
- \* India user goes in India webserver for purpose.
- \* US user goes in US server for purpose latency.
- \* health check provide
- \* do user goes in/on nearest Datacenter.



PAGE NO.  
DATE:

search  
brain  
optimization SEO

dig @ name server  
dig @ 8.8.8.8 domain

### ⑤ failover Routing policy

- \* This is based on health check.
- \* Primary server health check fail it goes to secondary server.
- \* Only primary server down than all traffic goes to secondary servers.

### ⑥ multiValue Answer Routing policy

- \* It is same with simple policy
- \* But in this policy we can add health check for every IP address.
- \* When any server down the traffic goes to another server.

### ⑦ IP based Routing policy

- \* In IP based Routing we make CIDR collection.
- \* In Record set a value which resolve with your CIDR collection.
- \* Default value resolve make compulsory.

ip → domain → value

\* An organization which at same geolocation having multiple departments and each departments are in different CIDR block hence to redirect them to different servers we can use IP based - Routing.

## \* [Route-53 - Traffic policy] \*

\* when complex scenario B create than we use traffic policy

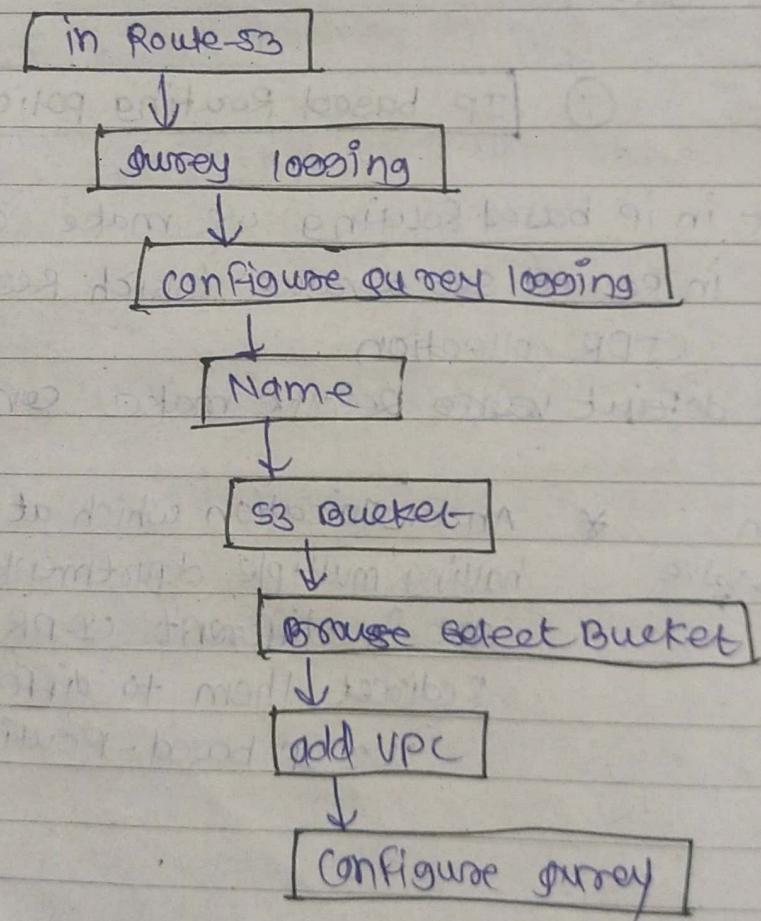
*too costly we used for industry level*

\* when we create weighted policy and in this weighted policy Big traffic on server and this server we also add failure policy for no any server down scenario occur.

\* it is part of VPC

## \* [Route 53 - logging] \*

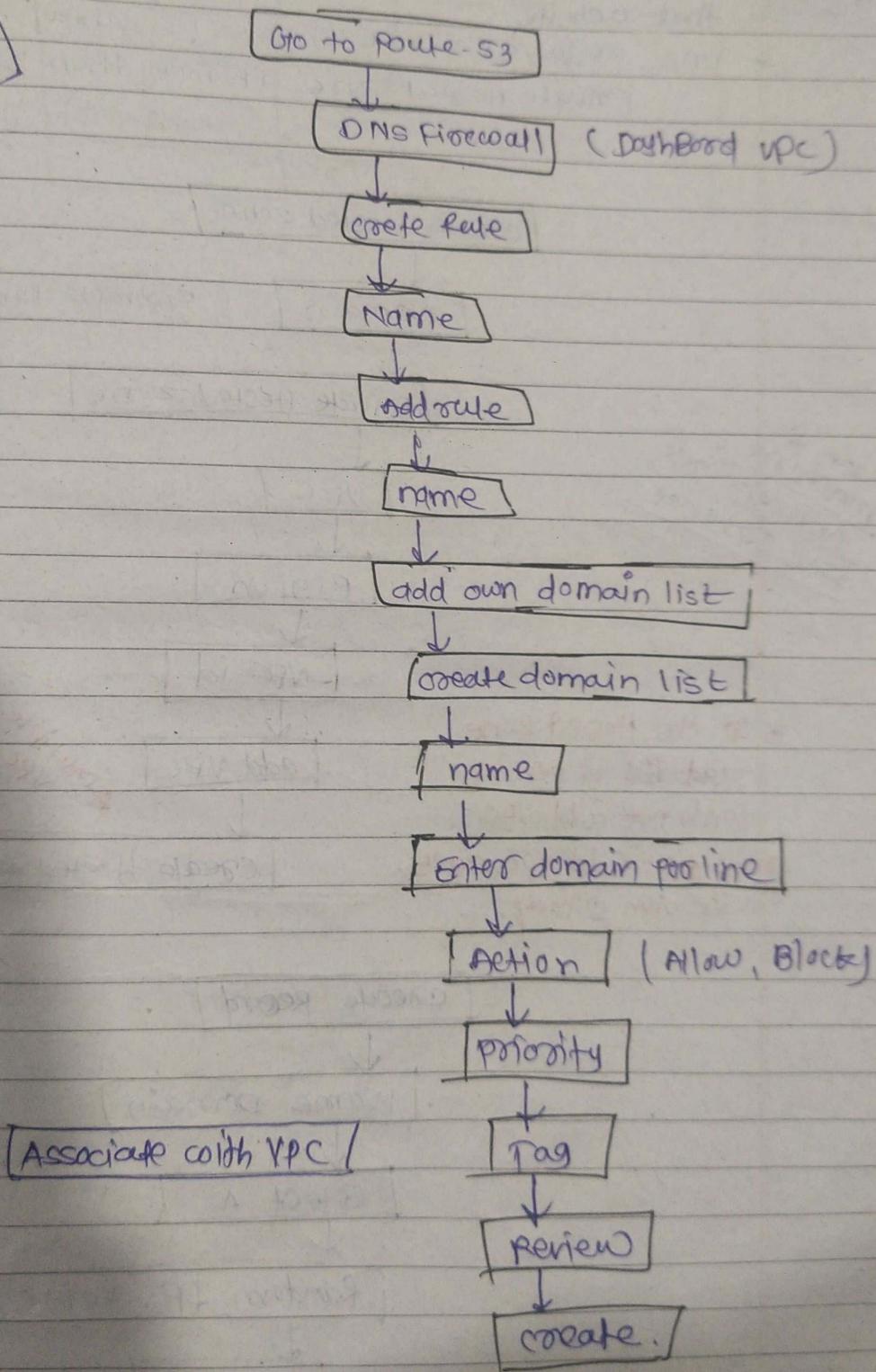
\* all log of DNS log save in S3 Bucket



## \* [ Route-53 - DNS Firewall ] \*

- \* we use for this website dieable or Not allow.
- \* Your instance not allow to specific website.

\*  
IT IS  
part of  
VPC



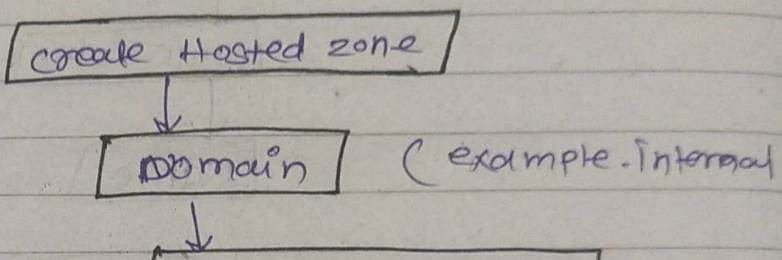
\* Route-53 → private Hosted zone

inside  
VPC

- \* some website work on only internally
- \* when outside form company we can't access that website.
- \* DNS Resolver first check is query on private Hosted zone if not then he check public.

\*\* machine  
Amazon Linux  
EC2 instance

- \* In the Hosted zone add VPC is peering with not add UPC
- \* so not add VPC can't resolve query.



private Hosted zone

VPC

Region

VPC-id

add VPC

Create Hosted zone

\* multiple  
VPC can add

Create Record

Name domain

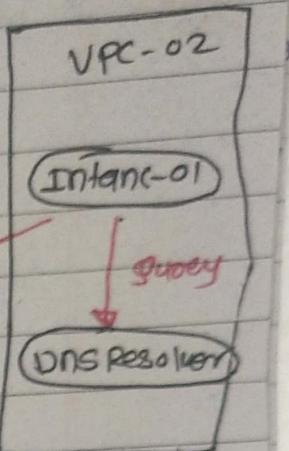
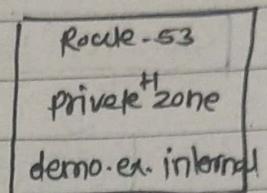
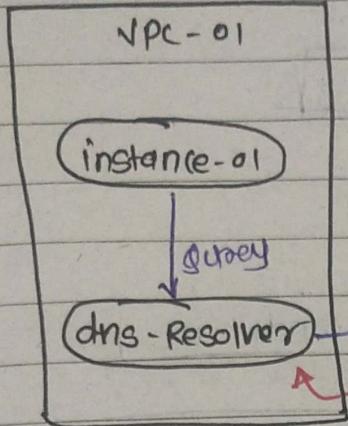
Select A

Random IP. Value

Create

\* VPC-01 and VPC-02 are peering connection established

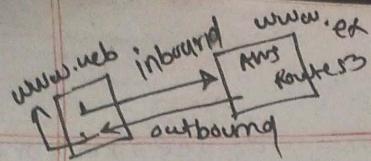
\* in private host zone  
VPC-02  
disassociate



\* it's not possible

\* nslookup domain server

→ Go  
to  
check  
on  
public  
No ans.



PAGE NO.

DATE:

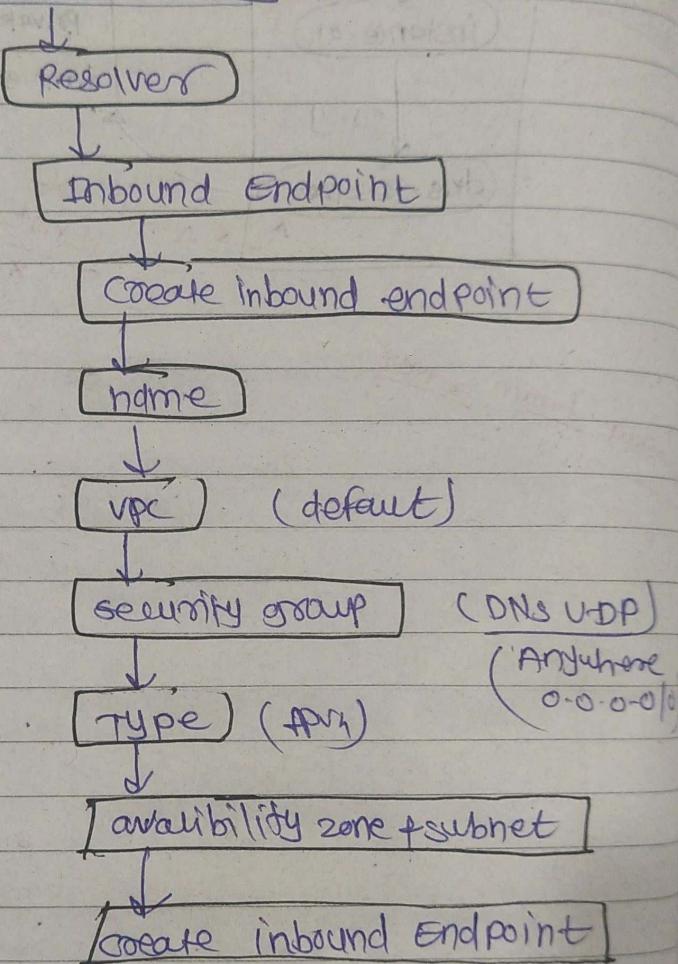
\* Client  
Go for  
AWS

## Route-53 - hybrid DNS - DNS inbound Endpoint

\* To resolve previous problem we use inbound Endpoint

\* AWS Go for outside \* Route

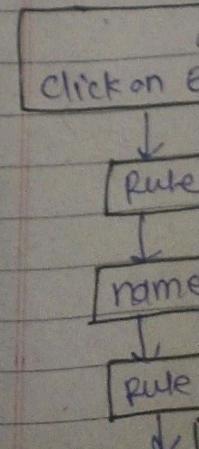
[Go to Route 53]



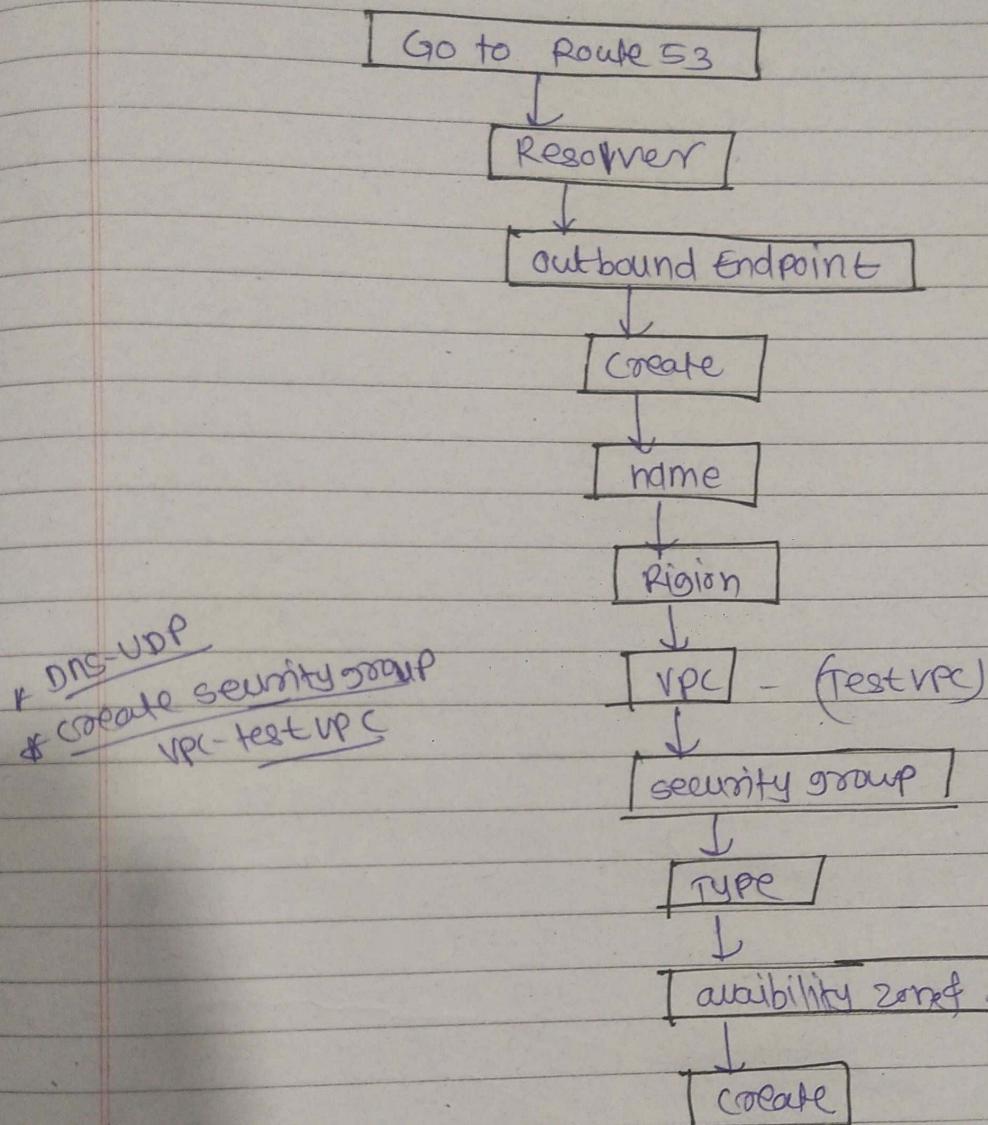
\* DNS-UDP  
\$ Create security  
VPC-test

nslookup domain name

\* We get two IP \*



\* AWS Go for outside \* [Route 53 - hybrid dns - outbound Endpoint]



we get two ip

