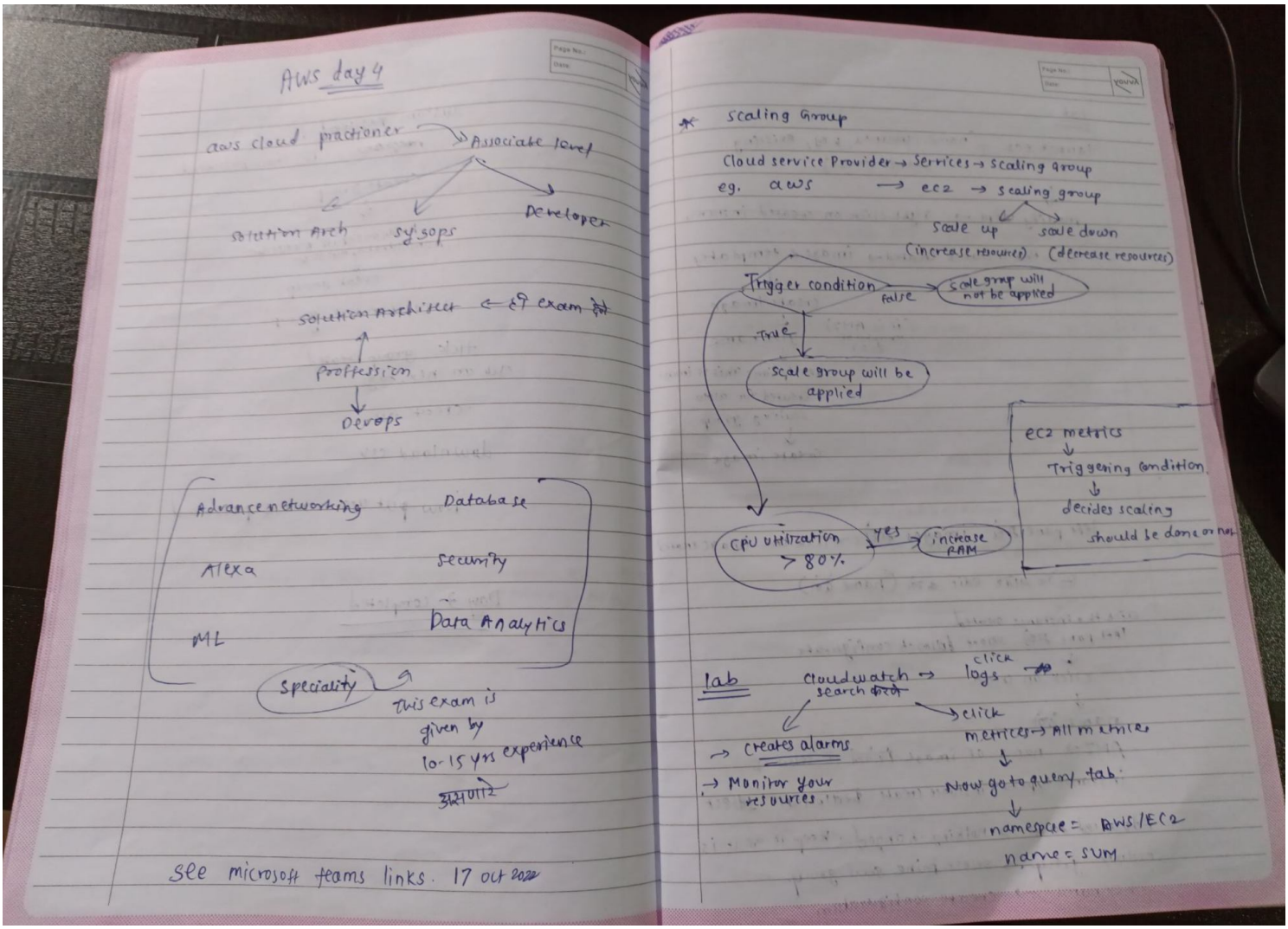

AWS day 4
aws cloud practioner
Associate level
Developer
sysops
Solution Arch
Solution Architect ← exam
Profession
Devops
Advance networking
Database
Alexa
Security
ML
Data Analytics
Speciality
this exam is given by 10-15 yrs experience
see microsoft teams links. 17 oct 2022
Scaling Group
Cloud service Provider → Services → Scaling Group
eg. aws → ec2 → scaling group
scale up (increase resources)
scale down (decrease resources)
Trigger condition
false
scale group will not be applied
True
scale group will be applied
ec2 metrics
Triggering condition
decides scaling should be done or not
CPU utilization > 80%
yes
increase RAM
lab
cloudwatch
search
logs
click
creates alarms
Monitor your resources
click metrices → All metrics
Now go to query tab
namespace = AWS/EC2
name = SUM

Lab

launch ec2 (name, ubuntu, key, existing. subnet choose)

↓

modify IAM → right click on created instance

Tick the instance created → image & templates

↓

create image

↓

(it is AMI) (snapshot of ec2) give name

↓

description: This is image created for auto scaling group

↓

Create image

left pane मध्ये images मध्ये → Tick the image created

(-) < यावर edit करणे (name देणे)

Click the instance created

left pane मध्ये select launch configuration

↓

click on create

↓

name देणे

AMI → name of image दिलेले ते select करणे

instance type → instance create करतानाचा select

Advanced details → nothing changed. keep it as it is

security group → select mine secu group

key name → create configuration

Imp steps → ec2 → AMI → launch config → attach launch config to auto scaling group

go to search box & search ec2 auto scaling

↓

select AWS auto scaling

ec2 → left page मध्ये → Auto scaling Groups

↓

create

↓

name देणे, click on switch to launch configuration

select the launch config created

↓

click on next

Click on vpc select it

↓

No load balancer

Health check grace period → 60 seconds

↓

next

group size — desired capacity 1

min 1

max 4

जर desired : 2
min : 1
max : 3
} असं असलं तर

So imp → min < desired < max

whenever initializing it will use desired capacity.

click Target tracking scaling policy

Average CPU utilization ✓

Target value : ~~50~~ ~~15~~ ~~20~~ 5

Instances need 60 ✓

↓

click on next

↓

Skip to review

↓

37:17 create autoscaling group

Tick the instance created. ~~Check the~~.

↓

monitoring मध्ये

Check CPU Utilization

Connect the instance not by session manager ~~via session manager~~

(tick the instance, ~~session manager~~ → connect)

Now on black screen,

(~~bash program for infinite loop ← Google वर search~~)

sudo apt update. यानंतर

sudo nano file2.sh

~~chmod 777 file.sh~~

47:00 ↓ 51:02 ~~49:33~~ पर्यंत बघणे recording

```
#!/bin/bash
# Infinite for loop
for ((;;))
do
    echo "CTRL+C to exit"
d
```

```
while true
do
    echo "hello world"
    sleep 1
done
```

see ~~1:04:44~~ 1:02:41

1:06:44

---

ec2 → click the instance created

↓

CPU utilization मध्ये

~~ctrl+C~~ ~~Ctrl+X~~ [illegible] read

2:07 PM at chats (MS teams)

run commands one by one

sent in outlook but written

⟹ increases CPU utilization

These commands are:

After "sudo apt update", type:

```
sudo apt install python3.8
enter
sudo add-apt-repository ppa:deadsnakes/ppa
enter

sudo apt install build-essential zlib1g-dev libncurses5-dev libnss3-dev libssl-dev libreadline-dev libffi-dev wget.
enter
```

Now check CPU utilization of created instance.

असं येतं म्हणजे जास्त होते

Also see in all instances created, max=4 so keys (meghna name) can be 4 times seen in the dashboard.

lab 2:

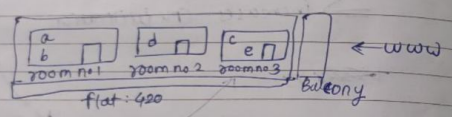
on google: relentless.com ← domain घेतले

2 domains can be attached on 1 IP

Route

Theory:

VPC: virtual Private cloud.



Hall is common for 3 rooms

each room has separate doors

a, b, c, d, e → person name

a, b → on same subnet

c, e → same subnet

d → service which does not depend on any other service

a, b, c, d, e → ec2 ~~झालेत~~

First component of VPC → I/G

& opens door → routing table

Isolation layer (private subnet)

a & d if are friends then can connect ⇒ VPC Peering

VPC Peering: not possible to form the connection without mutual Aknowledgement

router { ~~a~~ b & d like panipuri then connection ✓ But a, c, e don't know about it }

c & e are closer to balcony. c & e have their own private subnet ↓ used for testing

Que: If we stop/re-launch ec2, then the public IP will change.

dynamic || static

Private IP → VPC, FTP

Notes:

① VPC is virtual Network dedicated to your AWS account

② It is logically isolated from other virtual network in AWS cloud.

③ It provides complete control over the virtual networking environment including selection of IP range, creation of subnet, configuration of route tables & gateways

④ You can launch your AWS resources to your VPC

⑤ When you create a VPC you must specify a range of IPv4 addresses for the VPC in the form of a classless inter domain ~~routing~~ routing. (CIDR) 10.0.0.0/16

⑥ network ID is the AND operation between
1928.168.0.1
255.0.0.0

⑦ VPC spans all the availability zone of the region
⑧ By default you can create 5 VPC per region

* Components of VPC

① Network interface: a point of connection between public & private network.

② Route table: defines how traffic is routed between each subnet

③ Internet gateway / NAT (Network Address Translation)
: It is logically enabled routing of a traffic in the public network

④ Elastic IP: It is the static IPV_4 address
⑤ VPC endpoint: private connection betn VPC and other AWS services without using Internet

* Difference betn

Private subnet	Public subnet
- Resources are not exposed to outer world	- ~~they~~ are exposed to outer world through IG
- They make use of only private IP	- public IP & private IP
- used for backend, database & application server	- mainly used for frontend application

IP address → 172.32.0.0/16
192.168.100.1

192 binary 11000000
168 binary 10101000
100 binary 1100100
1

32102001
network → 15
host → 05

* lab:

Conclusion: ① create VPC ② public subnet 1b ③ private subnet 1a ④ create public rtr ⑤ create private rtr ⑥ attach respectively ⑦ create internet gateway ⑧ Connect ig to public rtr

① search for VPC
↓
~~create~~
click on your VPCs on left pane
Tick it that is the default VPC - 22f last तो - ID
create VPC → name दो (vpconly) → (10.0.0.0/16) ← IPV_4 ✓

No IPv6 CIDR block ✓

~~keys~~ create ✓

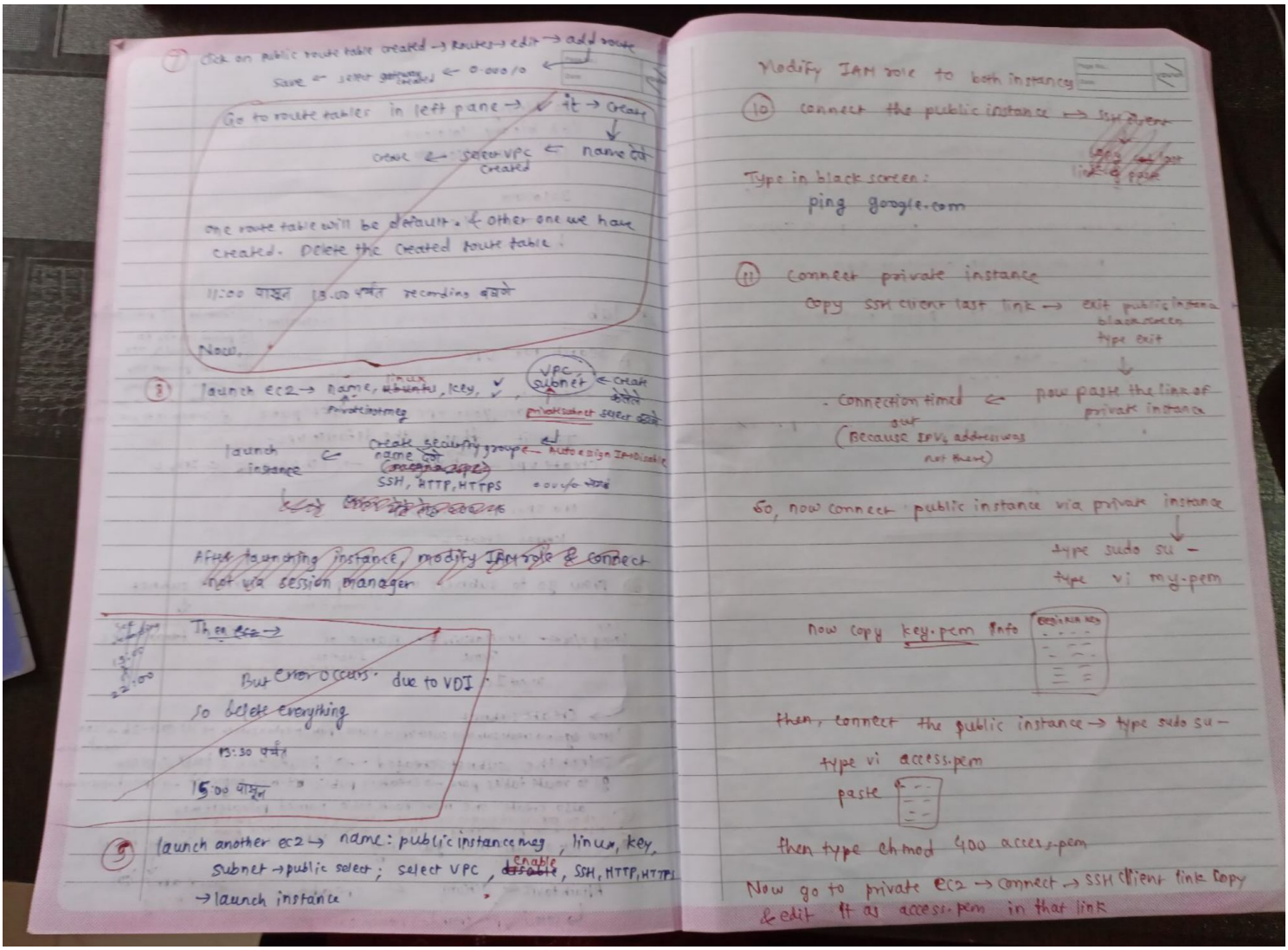
② Now go to subnets in left pane (~~default subnet~~ is already there) → create subnet
↓
select ~~terana~~ created VPC → name of subnet (privatemeghnasubnet) → availability zone (us-east-2 a (ohio)) → CIDR 10.0.1.0/24
→ Create subnet.

③ Now again → create subnet → select VPC → name: publicmeghnasubnet → us-east-2b → CIDR → create subnet

④ ~~select the subnet created → Route table edit~~
go to route tables pane → create → public-rt-meg name दो → VPC choose → create
also create one more route table named privatertmeg.

⑤ click on public-rt-meg → subnet associations → edit → select public subnet → save.
Also click on privatertmeg → sub association → edit → select private subnet → save

⑥ → Click on Internet gateway in left pane → create new → name दो (megig) → create ✓ → actions → Attach to VPC
↳ select VPC created → Attach gateway


7 Click on public route table created → Routes → edit → add route
Save ← select gateway created ← 0.0.0.0/0
Go to route tables in left pane → it → create
name दो ← select VPC created ← create
one route table will be default & other one we have created. Delete the created route table
11:00 पासून 13.00 पर्यंत recording केली
Now,
8 launch ec2 → name, linux, key, vpc subnet ← create केली
launch instance ← create security groups ← Auto assign IP Disable
SSH, HTTP, HTTPS 0.0.0.0/0 वाले
After launching instance, modify IAM role & connect that via session manager
Then ec2 →
But error occurs due to VDI
So delete everything
13:30 पर्यंत
15:00 पासून
9 launch another ec2 → name: public instance meg, linux, key,
subnet → public select; select VPC, Enable, SSH, HTTP, HTTPS
→ launch instance
Modify IAM role to both instances
10 connect the public instance → ssh client
Type in black screen:
ping google.com
11 connect private instance
copy SSH client last link → exit public instance black screen type exit
. Connection timed ← now paste the link of private instance
(Because IPv4 addresses was not there)
So, now connect public instance via private instance
type sudo su -
type vi my.pem
now copy key.pem info
then, connect the public instance → type sudo su -
type vi access.pem
paste
then type chmod 400 access.pem
Now go to private ec2 → connect → SSH client link copy & edit it as access.pem in that link

paste that in black screen

type ping google.com

=> private instance cannot connect internet

(12) NAT gateway → create name, public subnet ✓,

Allocate elastic IP → create NAT

(13) Update route table for private route table

↓

0.0.0.0/0 ← nat search ← save route

select NAT

private subnet can access internet via NAT gateway

Day 4 completed