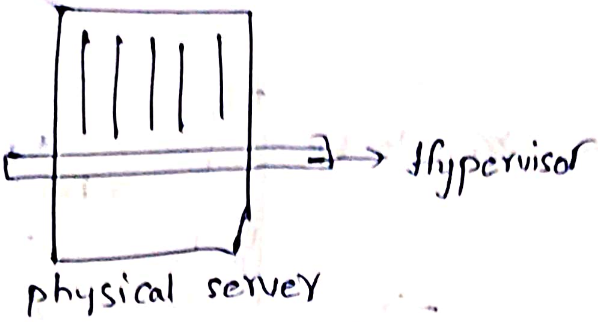
DAY - 4

Azure virtual machines

1) create a virtual machine in Azure → use Relevant i/p's & create
2) connect to the VM → copy public address
3) Deploy your first app on an Azure vm
4) virtual machine scalesets for Autoscaling.

virtualization :-

let's say these are physical servers which come with 100 Cpu, 100 TB mem.. Assigning one physical server to one developer would be definately waste. He can't use total space .. To use these physical servers, there is a concept called virtualization.

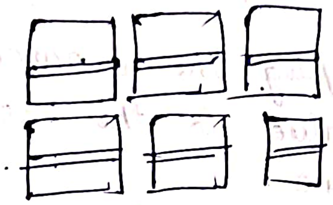


* A software installed on a physical server

* Hypervisor is responsible for ~~isolating~~ dividing this physical machine logically. so that multiple people can use this One physical server.

Eg:- Azure purchases the physical servers from IBM,

Zone-1 – East-US



Hypervisors installed on all physical servers.

when we request for one Azure vm, this is how vm assigns to us on one of the physical servers.

------- Inside portal, we create VM --------

1. Subscription – free trial
2. Resource Group – create one & give that
3. Region – Always select the nearest Region.
4. Availability zone – Zone 1 or 2 or 3
5. image – select free trial or select acording to the Requirement.
6. VM architecture – Arm64 or X64
7. Run with Azuere spot discount :- ~~Select~~ ~~yes~~
   It will give for special discount if there are any unuse Resources

* we shouldn't use this for production. Because Azure can't guarantee.. These machines can go down any time.

8. Size – Very important. Because charge will be according

there are
1) General purpose VMs
2) memory optimized VMs
3) cpu optimized VMs etc there are So many

which one should we select :-

1. what kind of developer is asking ?

A series :- Entry-level VMs for dev/test - we can avoid retired on August 31st 2024

Bs-series :- strictly for Read/write usecases. only for learning the platform

- very less cpu, very less RAM - 1 cpu, 1 RAM less than mobile

* Economical burstable VMS

- In organizations, we mostly not use Bs series unless some emergency

D-series :- General purpose compute

If a developer wants to deploy a sample application with not much configurations, then we can give D-series VM to him.

But if the developer came to you & he have an application with the memory intensed, some thing like reddis, then create

E-series : ↑

F-series :- compute optimised VMS

multiple processors running in an application - for those kind of applications use F-series.

- Gaming
- crown Jobs
- multi threading apps

H-series :- High performance computing virtual machines

G-series :- memory & storage optimised VMS

LS-series - storage optimised VMS

- DB instances can be installed

we have multiple types of series - we have to select accordingly.

N-series :- GPU enabled VMS

- LLMS or running ML.

① Authentication type :-

SSH or password
↓
selected one

username : ~~root~~ azureuser

SSH public key source : Generate new key pair

Key pair name : first-VM-Key

once after selecting all the feilds, click 'Review + create' then we will get an option " Generate new keypair" - Download private key & create Resourse.

→ In downloads folder, we see first-VM-key downloaded. & Azure VM creates.

→ "first-VM" VM creates.

using public ip address we will login to the Azure vm.

2 ways to connect → 1. Azure shell - not recommended.
2. from terminal / gitbash from windows

```
ssh -i /users/xfmbr22/downloads/first-VM-Key.pem azureuser@
~~publicip address~~ 172.178.84.132
```

```
ssh -i path-to-pem-file username@ipadress
    ↑
 identity
```

Warning : unprotected private key file ~~es~~

permissions 0664 for pem file is too open. It is required that your private key files are not accessible by others. permission denied

→ we will see this error

→ chmod 600 pem file.

(this will set the file permissions to read and write for the owner of the file, and no permissions for group mem & others).

Now, ssh command works fine.

now, we connected to a VM. now, we have to deploy a Jenkins app.

→ After installing Jenkins, it runs on a particular ~~host~~ port

there can be multiple applications deployed on a VM, to access those applications, we need to access ipadress@port every application has unique port

```
$ps -ef | grep Jenkins
Jenkins   17449   1 46 14:10 ?   -
                              --httpport =8080
```

→ we can see jenkins is running in 8080 port.

Now, go to browser →

http://172.178.84.132:8080 → initially it won't work, because in Azure, all the ports are blocked by default. we have to open the port. That's for security retesons.

To do that →

Azure vm in portal → go to vm → network settings → inbound port rules..., we have to open 8080 port

create port rule → ~~So~~

1. source : Any (from where the app should be accessible)
2. ~~Source~~ destination : Ip address, for now Any.
3. service : custom
4. port : 8080
5. protocol : Any
6. Action : Allow
7. priority : 400

now, In browser, we can access jenkins application.