Serverless computing: ---

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

cloud computing: where ever they deploying we can use those sites that is called cloud computing..

Serverless technology: --- not meant by no server present.. server should be present.. maintenance not be take care by human..\

Example : Ec2 machine, os , hardware, image , storgae, security group.

If you install any application like python ,first need to login, then python environment should be set up, libraries need to be installed , version should be matched...etc...

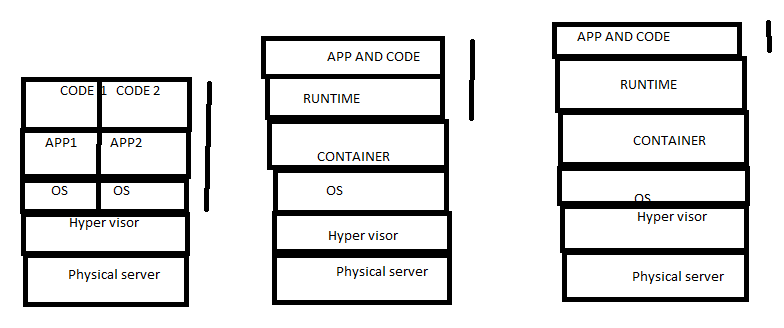
need full dependencies install on the ec2 mechaine.. if they need to access we need to host in perticular host..these many are very difficulty.

To overcome this we have containers .. container as a service... we can resolve few .. still we have few issues by using serverless computing we can reslove most of the cases..

Virtual machines all - IAAS, CASS , SAAS..

Container as a service

| IAAS | CAAS | SAAS |
| --- | --- | --- |
| Need one server to deploy , physical server must be present | Physical server must be present | physical server must be present |
| to host vertual mechine need Hypervisor | to host vertual mechine need Hypervisor | to host vertual mechine need Hypervisor |
| if two VM hosted --guest OS, Runtime -JDk,JRE on top of this applications, on top of applications we have code | os, on top of here we have container engine(K8S, docker swam, EKS) | OS and container system, Run time |
| all this maintence should be take care by users | container split into different pods | Application and code |
| Eg: Ec2 | runtime, app1 ,app2 | exclude application and code everything maintain by cloud |
|  | Till containere splits cloud will be taken care | only we have to taken care of application and code. |
|  | here only runtime and application code wiil be taken care by user |  |
|  | Eg: EKS, ECS | Eg: Lmbda |



Serverless computing called as function as a service ALSO…(FAAS)

Lambda :-- high availability, scalability… since less down time..

Pricing very low comparing with other clouds.. Pay as you use..

When ever hit the particular sit at that time code will be convert into container and run and provide the response.. How much time that is up and run till that time only they charge ..

AWS LAMBDA:-----

—------------------------------------------------------------------

Advantages: No maintenance, low down time, high availability, scalability, pay as you use…

Scalability : – 100 user use at once, containers will create 100 and give the response.. How many request will come that many containers will create, 1000 concurrent request per region…

Aws support will give more options like 1000 to 10000 or more..

First container exception will not be dependent on any other.. No ambiguity , no dependency.. Iso lated environment…

Monitoring feature also provided by lambda…

Logging…if any error happened to reason we have to look into logs,, provided by cloud –cloudwatch…

AWS lambda mostly used in event based triggers …

S3 bucket having data(excel, json, css)--> while uploading need to verify whether that should be having correct extension or correct format…write one event notification and trigger lambda..

Code will whether particular extension came or not —> then pass or fail ..

We know service EC2 (start and stop)..

Suppose night time that shold be stop and remaining time should be start..

Create one lambda function and we write the condition based on our requirement..

We can integrate this lambda in cloud watch,, cloudwatch will inform to lambda…

Particular runtime need to select,

Aws lambda supports , python, java, nodejs and Rubby…