Compute

Đà Nẵng, năm COVID thứ 2

EC2 1

EC2 Instance Types:

- General Purpose: M4 to M6 (fixed perf), T2 to T4 (burstable perf).
- Compute Optimized: C4 to C6.
- Memory Optimized: R4 to R6, X1, Z1.
- Storage Optimized: D2 to D3, i3.
- Accelerated: P2 to P4, G3 to G4.

EC2 2

Instance Lifecycle:

- Launch:
  - Creation of an instance from an AMI.
- Stop:
  - The instance is stopped.
- Start:
  - Status goes from Stopped to Pending to Running.
- Reboot:
  - The instance keeps its public DNS name (IPv4), private IPv4 address, IPv6 address (if applicable), and any data on its instance store volumes.
- Hibernate:
  - Signal the operating system to perform hibernation (suspend-to-disk).
- Terminate:
  - Status goes to Shutting-down and then to Terminated.

EC2 3a

EC2 Storage:

- Boot volume (disk) can be based on either the Instance Store or EBS.
- Instance Store volume:
  - Volume on the local disk of the host server.
  - Lower latency
  - No data consistency after the VM is terminated (but is consistent on reboot).
  - Not resizeable.
  - AWS does not encrypt these volumes. You need to use OS-level or Filesystem-level encryption.
  - No data redundancy.

EC2 3b

- EBS volume:
  - Volume mapped to an EBS volume.
  - Consistent even when the EC2 instance is stopped/terminated.
  - You can change size and type.
  - For both root and data volumes on EBS, to attach the volumes to an instance, these need to be in the same AZ as the instance.
  - Volumes can be encrypted by EBS.
  - Data is written to multiple copies in the same AZ.

EC2 4

EC2 AMIs:

- Amazon Machine Image.
- Contains the image of the OS and all other software and configurations that you would like when launching new instances.
- Two types of AMIs: EBS-backed or instance-store-backed.
- EBS-backed AMI:
  - Includes one or more EBS snapshots.
- Instance-store-backed AMI:
  - The root device is an instance store volume created from a template stored in Amazon S3.
- You can copy an AMI within the same AWS Region or to different AWS Regions.

EC2 5

EC2 Instance Billing Types:

- On-demand Instances.
- Savings Plans.
- Reserved Instance (RI).
- Spot instances.
- Dedicated Host.
- Dedicated Instance.
- Capacity Reservation.

EC2 6

On-Demand Instance:

- pay by the second.
- vCPU-based limits per account and per region.
- Instance usages are billed for any time your instances are in a "running" state.

Saving Plans:

- commitment to a consistent amount of usage, in USD per hour, for a term of 1 or 3 years.

Reserved Instance (RI):

- Standard RI:
  - A 1- or 3-yr commitment for a consistent instance configuration, including instance type and Region.
  - Can be sold in the « Reserved Instance Marketplace ».
- Convertible RI:
  - Like Standard RI but you can exchange the reservation for an equal or higher reservation.

EC2 7

Spot instances:

- AWS unused capacity.
- up to a 90% discount compared to On-Demand prices.
- You specify the maximum price you are willing to pay and AWS will allocate Spot instances to you whenever the Spot price is lower than this maximum price your specified.
- If the Spot price exceeds your maximum willingness to pay for a given instance or when capacity is no longer available, your instance will be removed automatically.
- Removal options: Stop, Hibernate, Terminate.
- Termination notices are issued 2 minutes prior to removal (except for hibernation).
- When requesting a spot instance you can make a "Persistent Request" so that AWS keeps trying to spin your instance after it is removed.

EC2 8

Dedicated Hosts:

- A physical server with EC2 instance capacity fully dedicated to your use.
- Allow you to use your existing per-socket, per-core, or per-VM software licenses (BYOL).

Dedicated Instance:

- For instances that need to run on single-tenant hardware.
- May share the host with other instances from the same AWS account.
- You have no visibility on the host.
- You have no guarantee that when you run the instance it will always start on the same host.
- BYOL not supported.

EC2 9

On-Demand Capacity Reservations:

- Enable you to reserve capacity for your Amazon EC2 instances in a specific Availability Zone for any duration.
- Billing starts as soon as the capacity is provisioned.
- When you no longer need it, cancel the Capacity Reservation to stop incurring charges.
- You specify: AZ, number of instances, instance attributes.
- Can be combined with Savings Plans or Regional Reserved Instances to receive a discount.

EC2 10

EC2 Auto Scaling:

- Ensures that you have the correct number of Amazon EC2 instances available to handle the load for your application.

Auto Scaling with ELB:

- You can attach the load balancer to your Auto Scaling group to register the group with the load balancer.

EC2 11

Placement Groups:

- Placement strategies: Cluster, Partition, Spread.
- Cluster Placement Groups:
  - Packs instances close together inside an Availability Zone (same rack).
  - Enables low network latency or high network throughput.
  - Enhanced Networking is recommended.
  - Limited to some instance types. Recommended to use the same instance type in the placement group.
  - Can span peered VPCs but you will not get the full bandwidth.
- Partition Placement Groups:
  - Spreads your instances across logical partitions such that groups of instances in one partition do not share the underlying hardware with groups of instances in different partitions.
  - Can have partitions in multiple Availability Zones in the same Region.
  - Max 7 partitions per AZ.
  - A partition placement group with Dedicated Instances can have a maximum of two partitions.

EC2 12

Data Transfer Costs:

- Data transferred "in" to and "out" from EC2 and ElastiCache instances or Elastic Network Interfaces across Availability Zones or VPC Peering connections in the same AWS Region is charged at $0.01/GB in each direction.
- Same AZ: free
- Data transferred between S3/Glacier/DynamoDB/SES/SQS/Kinesis/ECR/SNS/SimpleDB and Amazon EC2 instances in the same AWS Region is free.

EC2 Instance metadata and user data:

- Instance metadata:
  - AWS metadata about your instance.
  - Divided into categories, for example, host name, events, and security groups.
- User data:
  - Data that you specified when launching your instance.

Lambda

Lambda:

- a serverless compute service that runs your code in response to events and automatically manages the underlying compute resources for you.

Lambda Scaling:

- The first time you invoke your function, AWS Lambda creates an instance of the function and runs its handler method to process the event.
- When the function returns a response, it stays active and waits to process additional events.
- If you invoke the function again while the first event is being processed, Lambda initializes another instance, and the function processes the two events concurrently.

Lambda Concurrency:

- Concurrency is the number of requests that your function is serving at any given time.
- Concurrency is subject to a Regional quota that is shared by all functions in a Region.

Lambda RDS Proxy:

- You can use RDS Proxy: manages a pool of database connections and relays queries from a function.
- This enables a function to reach high concurrency levels without exhausting database connections.
- Supported on MySQL and Aurora.

Lambda@Edge:

- is a feature of Amazon CloudFront that lets you run code closer to users of your application, which improves performance and reduces latency.
- runs your code in response to events generated by the Amazon CloudFront content delivery network (CDN).
- Refer to the CloudFront notes for more details.

ECS:

- A fully managed container orchestration service for Docker.

ECS Network Modes:

- awsvpc (recommended): The task is allocated its own elastic network interface (ENI) and a primary private IPv4 address. This gives the task the same networking properties as Amazon EC2 instances.
- bridge: The task utilizes Docker's built-in virtual network which runs inside each Amazon EC2 instance hosting the task.
- host: The task bypasses Docker's built-in virtual network and maps container ports directly to the ENI of the Amazon EC2 instance hosting the task. As a result, you can't run multiple instantiations of the same task on a single Amazon EC2 instance when port mappings are used.
- none: The task has no external network connectivity.

EKS

Amazon Elastic Kubernetes Service (EKS):

- Gives you the flexibility to start, run, and scale Kubernetes clusters in the AWS cloud or on-premises.
- automates key tasks such as patching, node provisioning, and updates.
- Can run on EC2 or AWS Fargate.
- EKS runs the Kubernetes control plane across three Availability Zones in order to ensure high availability, and it automatically detects and replaces unhealthy control plane nodes.
- EKS supports running Windows worker nodes alongside Linux worker nodes in the same cluster.

AWS Fargate:

- Fargate is a serverless compute engine that you can use to run containers for ECS, EKS and AWS Batch.
- Fargate removes the need to provision, scale and manage EC2 instances.
- Fargate allocates the right amount of compute, eliminating the need to choose instances and scale cluster capacity.

AWS Elastic Beanstalk (EB):

- With Elastic Beanstalk, you can quickly deploy and manage applications in the AWS Cloud without having to learn about the infrastructure that runs those applications.
- It keeps the provisioning of building blocks (e.g., EC2, RDS, ELB Auto Scaling, CloudWatch), deployment of applications, and health monitoring abstracted from the user so they can just focus on writing code.
- You simply specify the code or containers to deploy.
- Supports a large list of platforms: .NET, Docker, Java, Node.js, PHP, Python, …etc.
- Elastic Beanstalk will automatically handle all the details such as provisioning an Amazon ECS cluster, balancing load, auto-scaling, monitoring, and placing your containers across your cluster.
- Elastic Beanstalk is ideal if you want to leverage the benefits of containers but just want the simplicity of deploying applications from development to production by uploading a container image.
- You can create versions of your application (stored in S3).
- You can have multiple environments (prod, dev, test, …).
- You can change settings in the infrastructure created by EB. For instance, you can change the load balancing parameters of the ELB.

AWS offers a wide-range of website hosting options:

- Amazon Lightsail: for Simple Website hosting.
- AWS Amplify Console: for Single Page Web App hosting.
- Amazon S3: for Static Website hosting.
- Amazon EC2: for Enterprise Web Hosting.

Amazon Lightsail:

- Provides ready to use compute, web server software, Database, DNS, Load Balancer and Storage for websites hosting.
- Use cases:
  - Popular website stacks like LAMP, LEMP, MEAN, Node.Js.
  - Websites built on common applications like WordPress, Joomla, Drupal, Magento.
- Targeted at:
  - Websites that are unlikely to scale beyond 5 servers.
  - Customers who want to manage their own web server and resources.
- Simplified management console.
- Simplified billing with predefined packages.

AWS Amplify:

- Websites built with Single page app frameworks such as React JS, Vue JS, Angular JS, and Nuxt.
- Websites built with static site generators such as Gatsby JS, React-static, Jekyll, and Hugo.
- Progressive web apps or PWAs
- Websites that do not contain server-side scripting, like PHP or ASP.NET
- Websites that have serverless backends.