

Computing (Containers and Serverless)

SERVERLESS · AWS LAMBDA · APIS AND REST · API GATEWAY · CONTAINERS (ECS · FARGATE · EKS · ECR) · STEP FUNCTIONS · STUDY & REVIEW

01 MUST KNOW

if you remember five things, remember these

- 01 Serverless = no servers to manage:** AWS Lambda runs your code without provisioning or managing servers. It manages the infrastructure, provides built-in fault tolerance across **multiple Availability Zones** in each Region, and **scales automatically** to the request rate. You pay only for requests and compute time, metered in **100 ms** increments.
- 02 Know the Lambda limits:** **3 GB** maximum memory · **15-minute** maximum runtime · **250 MB** deployment package (function plus up to **five layers**, unzipped) · **1,000** concurrent executions per Region. A **layer** is a ZIP archive of libraries or a custom runtime, extracted to /opt.
- 03 Containers virtualize the OS:** VMs virtualize hardware and carry a whole OS; **containers** share a virtualized OS and package only what the software needs — libraries, system tools, code, and the runtime. **Docker** packages software into standardized container units, and a single application can span **multiple containers**.
- 04 Four container services:** **Amazon ECS** manages Docker containers on a cluster of EC2 instances (via JSON **task definitions**) · **AWS Fargate** is an ECS launch type that runs containers with **no servers or clusters** to manage · **Amazon EKS** is a managed **Kubernetes** control plane · **Amazon ECR** is a managed Docker image registry (stores images in Amazon S3).
- 05 APIs, gateways, and workflows:** **REST** APIs communicate over **HTTP** and expose resources at URLs (GET POST PUT DELETE). **Amazon API Gateway** is a fully managed front door for APIs — pay per call, with tight **Lambda** integration for serverless APIs. **AWS Step Functions** coordinates services into serverless workflows: a state machine where each step's output feeds the next.

02 KEY TERMS

TERM	DEFINITION
Serverless computing	Building and running applications and services without provisioning, scaling, or managing any servers — AWS handles the infrastructure
AWS Lambda	AWS serverless compute service; runs your code without provisioning or managing servers, triggered by AWS services or invoked directly
Lambda layer	A ZIP archive of libraries, a custom runtime, or other dependencies; keeps the deployment package small; up to 5 per function, extracted to /opt
Container	A method of OS virtualization; an application and its dependencies run as resource-isolated processes that share a virtualized operating system
Docker	A software platform that packages software into standardized units called containers, allowing quick build, test, and deployment
Task definition	A declarative JSON template for an ECS task specifying its containers — Docker image, CPU and memory, volumes, and container links
Amazon ECS	Highly scalable, high-performance container management service that supports Docker and runs on a managed cluster of EC2 instances
AWS Fargate	A technology (launch type) for Amazon ECS that runs containers without having to manage servers or clusters
Amazon EKS	Elastic Container Service for Kubernetes — a managed Kubernetes control plane run across multiple AZs; certified Kubernetes conformant

TERM	DEFINITION
Amazon ECR	Elastic Container Registry — a fully managed Docker container registry to store, manage, and deploy images; stores images in Amazon S3
REST	Representational state transfer — a design for loosely coupled network applications that communicate over HTTP and expose resources at URLs
AWS Step Functions	Coordinates AWS services into serverless workflows expressed as a state machine; the output of one step is the input to the next

03 DEPLOY AND RUN A LAMBDA FUNCTION

handler to CloudWatch monitoring

01 Define a Lambda **handler** in your code → **02 Create the function** — code, configuration, and resource requirements → **03** Configure access with an **IAM role** (security groups, network ACLs) → **04 Upload** your code → **05 Test**, verify results, and review logs → **06 Monitor** with Amazon CloudWatch — requests, latency, errors

04 CONTAINERS VS. SERVERLESS

who runs the servers?

CONTAINERS (ECS / EKS / FARGATE)	SERVERLESS (AWS LAMBDA)
<i>you keep more control and portability</i>	<i>AWS runs the servers</i>
— Package an app and its dependencies; run the same image in test and production — Portable and vendor-agnostic — deploy on any compute, avoid lock-in — You manage resources, policies, and security; EC2 launch type gives granular control — Fit microservices, Windows containers, EC2 Spot or Reserved Instances	— Upload code — no servers to provision, scale, or manage — Automatic scaling and built-in fault tolerance across multiple AZs — Pay only for requests and compute time (100 ms increments) — Fit event-driven tasks, scheduled jobs, glue between services

05 AWS CONTAINER SERVICES AT A GLANCE

one-line job of each — exam gold

SERVICE	WHAT IT IS	REMEMBER
Amazon ECS	Scalable Docker container management service running on a managed cluster of EC2 instances	Uses task definitions (JSON); launches in your VPC
AWS Fargate	A technology (launch type) for ECS that runs containers with no servers or clusters to manage	Pay by task size, per second, one-minute minimum
Amazon EKS	Managed Kubernetes control plane run across multiple AZs; certified Kubernetes conformant	You add worker nodes; use existing Kubernetes tooling
Amazon ECR	Fully managed Docker container registry to store, manage, and deploy images	Stores images in Amazon S3; IAM access; encrypted at rest

06 REST METHODS AND COMMON HTTP STATUS CODES

verbs create/read; codes report outcome

ITEM	MEANING
GET	Read (retrieve) a resource; rarely includes a body
POST	Create a new resource; typically carries a body
PUT	Update (edit) an existing resource; typically carries a body
DELETE	Delete a resource
200	Success — the request was received and accepted

ITEM	MEANING
401	Unauthorized — authentication required, or credentials not accepted
403	Forbidden — the request was valid, but the server will not allow it
404	Not Found — the resource is unavailable or could not be accessed
500	Internal Server Error — an unspecified server-side error
503	Service Unavailable — the service is temporarily unavailable

07 EXAM TRAPS

where the knowledge check gets you

- × “Serverless means there are no servers.” — There are servers; you just do not provision, scale, or manage them. AWS runs and maintains the compute infrastructure on your behalf.
- × “You have to configure Lambda scaling and cap the request rate.” — Lambda scales automatically to the incoming request rate with no configuration and no limit on the number of requests (though there is a per-Region limit of **1,000** concurrent executions).
- × “Lambda can run a multi-hour batch job.” — The maximum runtime is **15 minutes** and the maximum memory is **3 GB**; long-running work needs a different approach, or Step Functions to coordinate multiple functions.
- × “A container is basically a small virtual machine with its own OS.” — A VM virtualizes hardware and carries a full guest OS; a container virtualizes and **shares** the OS, holding only libraries, tools, code, and runtime — which is why it is smaller and boots fast.
- × “AWS Fargate is a separate service with its own APIs.” — Fargate is a **launch type within ECS** (the alternative to the EC2 launch type); its capabilities are native to ECS, and one cluster can run both Fargate and EC2 tasks.
- × “Amazon EKS runs the Kubernetes control plane on your EC2 instances.” — EKS runs the **managed** control plane for you across multiple AZs; you add worker nodes. Running Kubernetes yourself on EC2 is the self-managed alternative.
- × “A GET request usually carries a body.” — GET reads a resource and rarely includes a body; the body (data) belongs to POST and PUT requests, which create or update resources.
- × “API Gateway charges minimum or upfront fees.” — There are no minimums and no upfront commitments; you pay only for the API calls received and the data transferred out.

08 SELF-QUIZ

answers are upside-down below

- Q1** What is the maximum runtime for a single AWS Lambda function?

Q2 What is the maximum memory that can be allocated to a single Lambda function?

Q3 What is the per-Region limit on concurrent Lambda executions cited in the module?

Q4 A Lambda layer is what kind of file, and to which directory is it extracted at runtime?

Q5 Which ECS launch type removes the need to provision and manage servers and clusters?
- Q6** Which service is a fully managed Docker image registry, and where does it store images?

Q7 Which service provides a managed Kubernetes control plane run across multiple Availability Zones?

Q8 Which fully managed service lets developers create, publish, and secure APIs and integrates tightly with Lambda for serverless APIs?

Q9 In AWS Step Functions, what is the relationship between one step's output and the next step?

Q10 Name the four common REST methods and the action each performs.

ANSWER KEY (rotate the page)

Q1 15 minutes **Q2** 3 GB **Q3** 1,000 **Q4** A ZIP archive; extracted to /opt **Q5** AWS Fargate **Q6** Amazon ECR; it stores images in Amazon S3 **Q7** Amazon EKS **Q8** Amazon API Gateway **Q9** The output of one step is the input to the next step **Q10** GET reads, POST creates, PUT updates, DELETE deletes