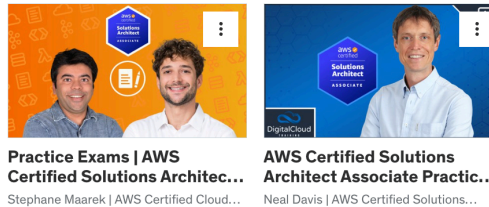


AWS Solution Architect Associate| SAA-C03 - Cheatsheet

I would like to create this page documentation for everyone, who wants to know how much knowledge you have to learn and note some knowledge that you have to remember or distinguish between some services such as ElasticCache for Memcached or Redis, EFS vs EBS...

As a practitioner, and take example exam courses from UdeMy from Neal Davis and Stephane Maarek, helped me a lot



This exam will focus on multiple topics but mainly will ask you about the *high availability architect*, *optimization cost/cost-effectiveness*, *disaster recovery*, *migration data from on-premise to Cloud*, *hybrid cloud model*, *storage durable for cost optimization*, etc.

To be familiar with AWS questions for SAA-03, the question then usually asks to you select the answer (an AWS service, feature, option, etc.) that optimizes a variable such as:

- Lowest cost (the question could be calling for serverless options such as Lambda or Athena. Don't forget build costs, not just operational costs!)
- Scalability (again serverless services such as Lambda come into play because of the ability to handle bursts of incoming traffic)
- Easiest / least effort
- Lowest operational complexity (this may be calling for managed services such as RDS)

These are key words that you will face while read the question from AWS SAA-03 exam.

EC2 Instance purchasing Option:

Instance Type	Usecase	Description
On-Demand Instances	the default option, for short-term ad-hoc requirements where the job <i>can't be interrupted</i>	Pay attention to the question that require can't be interrupted work, that is the hint for you to choose On-Demand Instance
On-Demand Capacity Reservations	the only way to reserve capacity for blocks of time such as 9am-5pm daily	Please remember in the question that ask for purchase or choosing the instance that need to do a job for period of time such as 9AM - 5PM.
Spot instance	highest discount potential (50-90%) but no commitment from AWS, could be terminated with 2min notice. Could use	

	for grid and high-performance computing.	
Spot Block Instance	guaranteed to be available for a finite duration (1-6 hours) and are provisioned based on the available capacity in the Spot instance market.	For saving cost or choosing the work that operate from 1-6 hours you can choose spot block instance
Reserved Instances	for long-term workloads, 1 or 3 year commitment in exchange for 40-60% discount	
Dedicated Instances	run on hardware dedicated to 1 customer (more \$\$)	
Dedicated Host	fully dedicated and physically isolated server. Allows you to use your server-bound software licenses (e.g. IBM, Oracle) and addresses compliance and regulatory requirements and potentially reduce cost (note: billing is per-hour not per-instance)	When question asking for software license it is a hint that you should to choose Dedicated host.

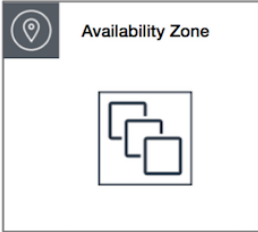
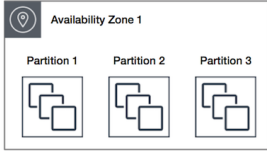

Dedicated Instance vs Dedicated Host

It is some questions that will ask about different between Dedicated instance and Dedicated host with multiple choose options, that you need to select two or three option for correct answer, here is table to compare between them:

	Dedicated Host	Dedicated Instance
Dedicated physical server	Physical server with instance capacity fully dedicated to your use.	Physical server that's dedicated to a single customer account.
Instance capacity sharing	Can share instance capacity with other accounts.	Not supported
Billing	Per-host billing	Per-instance billing
Visibility of sockets, cores, and host ID	Provides visibility of the number of sockets and physical cores	No visibility
Host and instance affinity	Allows you to consistently deploy your instances to the same physical server over time	Not supported
Targeted instance placement	Provides additional visibility and control over how instances are placed on a physical server	Not supported
Automatic instance recovery	Supported. For more information, see Host recovery .	Supported
Bring Your Own License (BYOL)	Supported	Partial support *
Capacity Reservations	Not supported	Supported

Dedicated Host vs Dedicated Instance

Placement Group - AWS Compute

Cluster Placement Group	Partition Placement Group	Spread Placement Group
<p>packs instances close together inside an AZ to achieve low latency, high throughput</p> <p>- use for HPC</p> <p>This is use for <i>low-latency network performance necessary for tightly-coupled node-to-node communication</i></p> 	<p>separate instances into logical partitions such that instances in one partition do not share hardware with instances in another partition. Gives you control and visibility into instance placement, but not great for performance. Used by large distributed workloads such as <i>Hadoop, Kafka</i>.</p>  <p>Note: A partition placement group can have a maximum of seven partitions per Availability Zone</p>	<p>place 1 or few instances each in distinct hardware to reduce correlated failures. Not great for performance</p> <p>Rack level spread placement groups</p> <p>The following image shows seven instances in a single Availability Zone that are placed into a spread placement group. The seven instances are placed on seven different racks, each rack has its own network and power source.</p>  <p>Note: In a Region, a rack level spread placement group can have a maximum of seven running instances per Availability Zone per group</p>
<p>Ref:</p> <p>https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-strategies.html#placement-groups-cluster</p>	<p>Ref:</p> <p>https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-strategies.html#placement-groups-partition</p>	<p>Ref:</p> <p>https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-strategies.html#placement-groups-spread</p>

Scaling instance:

In high-availability contexts you use an Auto-Scaling Group (ASG) to automatically launch and stop instances, and an Elastic Load Balancer (ELB) to distribute traffic among the instances

Auto Scaling Group policies:

Simple	Scheduled	Dynamic	Step	Target Tracking
Uses a metric like CPU utilisation to scale.	scale based on a scheduled event or recurring schedule (e.g. if you know that you have traffic spike every morning at 9am)	Scaling-in and scaling-out happens based on configured rules called policies.	Like Simple Scaling but has multiple stepsto scale. These steps are called Step Adjustments.	Keeps instance at a fixed level using, for example, CPU utilization.

VPC, SUBNET, NETWORKING

- A Virtual Private Cloud (VPC) mirrors the structure of a conventional network typically found in an on-premises data center.
- Subnets within a VPC represent specific ranges of IP addresses. Deploying AWS resources within these subnets is enabled after their creation.
- Route tables are utilized to dictate the path of network traffic originating from subnets or gateways.
- Gateways establish connections between a VPC and external networks. For instance, an internet gateway facilitates VPC connectivity to the internet.
- VPC endpoints provide private connections to AWS services, bypassing the need for internet gateways or NAT devices.
- VPC peering connections facilitate traffic routing between resources within two separate VPCs.
- Employ a Transit Gateway as a centralized hub for routing traffic between VPCs, VPN connections, and AWS Direct Connect connections.
- Establish connections between VPCs and on-premises networks via AWS Virtual Private Network (VPN) solutions.

Subnet (Public & Private)

A VPC is located within a Region, and each subnet aligns with a specific Availability Zone (AZ). To ensure high availability, you should have at least two subnets in your VPC spanning across two AZs.

When you create a new subnet, it automatically links to the main route table.

AWS recommends different VPC/subnet setups:

- VPC with a single public subnet: Suitable for simple, public-facing web applications like blogs or basic websites.
- VPC with both public and private subnets: Ideal for multi-tier web applications, with web servers in the public subnet and databases in the private subnet.

Public vs Private

Public Subnet	Private Subnet
<u>Has a route table that routes to an Internet Gateway</u> (note the Internet Gateway is attached to the VPC, not directly to the subnet)	<u>Outbound traffic is routed to a NAT device. The NAT device is installed in the Public Subnet and connected to an Internet Gateway for outbound access to the internet.</u>
When EC2 instances launched in a Public Subnet, they are auto-assigned a public IP address or ENI	NAT Gateway vs. NAT Instance = NAT Gateway is managed for you by AWS and highly available, whereas NAT Instance is a lot more manual work but can be used as a bastion host / jump box
Security groups and network ACLs on Public Subnet must allow SSH traffic (on port 22) for admin config.	EC2 instances <u>don't</u> have public IP or ENI You have to use a bastion host ("jump box") to access instances in the Private Subnet over SSH (port 22)

VPC Endpoints

Interface Endpoints	Gateway Endpoint
<i>Privately connect your VPC to AWS services, services hosted by other AWS accounts, and supported AWS Marketplace</i>	Direct traffic to S3 or DynamoDB only, using private IP addresses:

services as if they were in your VPC

- powered by **AWS Privatelink**
 - applies to many AWS services (API Gateway, CloudFormation, CloudWatch, S3)
 - does not go over the internet
 - no need to use an internet gateway, NAT device, DX connection, or VPN
 - Is an ENI with a private IP address, in the subnet that you specify, directing traffic to the service that you specify. Uses DNS to direct traffic to the service. Protected by a Security Group.
- Does not enable AWS Privatelink
 - You route traffic from your VPC to the gateway endpoint using route tables. Protected by VPC endpoint policies rather than Security Groups.

Security Group vs Network ACL

Security Group	Network ACL
<ul style="list-style-type: none">• <u>Security Group is at the instance level, Network ACL is at the subnet level</u> and applies to all instances within that subnet• Security Groups don't have deny rules, Network ACL have accept and deny• Security Groups are stateful, Network ACL stateless• Security Groups evaluate all rules together, Network ACL processes rules in order• Neither can block traffic by country• Security Groups have inbound allow rules allowing traffic from within the group, whereas custom security groups don't allow any inbound traffic by default. All outbound traffic is allowed by default.• Security Group default state: outbound rule allows all traffic to all IPs, but inbound has no rules and traffic therefore denied by default	<ul style="list-style-type: none">• NACLs function at the subnet level with separate allow/deny rules for inbound and allow/deny rules for outbound. They are stateless so it's all about what the rules say each time. Don't apply -within- the subnet, only in/out of the subnet.• Default security groups have inbound allow rules (from within the group). Custom security groups do not allow any inbound traffic. All outbound traffic is allowed.• VPC automatically comes with a default NACL which allows all inbound/outbound traffic. A custom NACL denies all inbound/outbound traffic by default.

Route 53

- Best practice is to use DNS names/URLs whenever possible rather than IP addresses. Some exceptions include pointing ELBs directly to the IP address of a peered VPC, or an on-prem resource linked via DX or VPN connection.

Alias Record	Cname Records
provide a Route 53-specific extension to DNS functionality	(canonical name records) redirect DNS queries to any DNS record. For example, you can create a CNAME record that

<ul style="list-style-type: none"> • They let you route traffic to selected AWS resources: ELBs, APIs, CloudFront distributions, S3 buckets, Elastic Beanstalk, VPC interface endpoints, etc. • Unlike a CNAME record, they also let you route traffic from one record in a hosted zone (usually the <i>zone apex</i> / naked domain name, such as "☞ Example Domain") to another record (e.g. "☞ Example Domain") • When Route 53 receives a DNS query for an alias record, it responds with 1 or more IP addresses that the record maps to 	<p>redirects queries from acme.example.com to zenith.example.com or acme.example.org.</p> <ul style="list-style-type: none"> • You don't need to use Route 53. • Unlike Alias records, they can't be used for resolving apex domain names
---	---

PTR records = reverse lookup where you map an IP address to a DNS name

AWS Direct Connect (DX) Gateway

- Utilize Direct Connect (DX) to establish connections between an on-premises data center and one or multiple VPCs.
- Setting up DX can take more than a month.
- For increased resilience, consider adding a second DX connection. Since this setup process is time-consuming and expensive, in the short term, also think about incorporating an IPSec VPN connection (with the same BGP prefix) for additional resilience.
- To start using DX, you must create one of the following virtual interfaces:
 - Private virtual interface (private VIF): Access a VPC using private IP addresses.
 - Public virtual interface (public VIF): Access all AWS public services using public IP addresses.
 - Transit virtual interface (transit VIF): Access one or more VPC Transit Gateways associated with DX gateways within a Region.
 - Hosted virtual interface (hosted VIF): Allow another AWS account to access your DX.
- Employ AWS DataSync to efficiently transfer large amounts of data from on-premises sources to S3, EFS, FSx, NFS shares, or SMB shares, including AWS Snowcone (via Direct Connect). For database migration, use AWS Database Migration Service (DMS).

AWS Transit Gateway

- Central Hub connecting on-prem networks and VPCs.
 - Reduces operational complexity as you can easily add more VPCs, VPN capacity, Direct Connect gateways, without complex routing tables.
 - Provides additional features over-and-above VPC peering
- A **transit virtual interface** is used to access VPC Transit Gateways
- Pattern for connecting 1 DX to multiple VPCs in the same Region is to associate the DX with a transit gateway
 - on-prem -> DX -> DX location -> transit virtual interface -> transit gateway association -> Transit Gateway -> multiple VPCs

VPC Connection

- VPN connections go over the internet
- **AWS Managed site-to-site VPN Connection** is connected between a **Customer Gateway** on the customer side and **Virtual Private Gateway (VPG, or VPN gateway)** that you create at the edge of your VPC.

AWS Cloudformation

- provision infrastructure using a text-based template that describes exactly what resources are provisioned and their settings. Can use scripts to automate the creation of member accounts and VPCs.
- manages the template history similar to how code is managed in source control
- 2 methods of updating a stack
 - a. *direct update* - CloudFormation immediately deploys your changes
 - b. *change sets* - preview your changes first, then decide if you want to deploy
- **AWS SAM (Serverless Application Model)** is an extension of CloudFormation for packaging, testing and deploying serverless applications

Disaster Recovery (DR)

- DR approaches
 - Backup and restore = lowest cost, just create backups
 - Pilot Light = small part of core services that is running and syncing data or documents
 - Warm Standby = scaled down version of a fully functional environment that is actively running
 - Multi-site = on-prem and in AWS in an active-active configuration
- For disaster recovery in a different region, create a AMI from your EC2 instance and copy it into a 2nd region.