For more information on AWS, visit aws.amazon.com

Description

Topic	Answer
Exam Time:	80 Minutes
No. Questions:	60 Questions
Question Types:	Scenario and Multiple Choice
Passing Score:	~ 70%
Validity Period:	2 years

PreRequiste Assumptions Note:

StudyGuide Note:
This study guide builds upon the AWS Solutions Architect Study Guide under the Notes section. You should reference that study guide and use this studyguide for additional information required for the AWS Developer Associate Exam.

General

> Amazon Web Services SDK's:

- Android, IOS, JavaScript (Browser)
- Java
- .NET
- Node.js
- PHP

- Python

- C++

> Default Regions:

- US-EAST-1
- Java has default region
- Some languages such as Node.js do not have a default region

Service Limits:

Each service has the default limits defined, to see the official AWS documentation on service limits, check here

Networking:

> **VPC (Virtual Private Cloud):**

- This topic is covered in AWS Solutions Architect Study Guide
- Virtual data center in the cloud
- Allowed up to 5 VPCs in each AWS region by default
- All subnets in default VPC have an Internet gateway attached
- Multiple IGW's can be created, but only a single IGW can be attached to a VPC
- Each EC2 instance has both a public and private IP address
- If you delete the default VPC, the only way to get it back is to submit a support ticket
- By default when you create a VPC, a default main routing table automatically gets created as well.
- Subnets are always mapped to a single AZ's
- Subnets can not be mapped to multiple AZ's
- /16 is the largest CIDR block available when provisioning an IP space for a VPC

    - x.x.x.1 - Reserved by AWS for the VPC router
    - x.x.x.2 - Reserved by AWS for subnet DNS
    - x.x.x.3 - Reserved by AWS for future use
    - x.x.x.255 - Always subnet broadcast address and is never usable.
- 169.254.169.253 - Amazon DNS
- By default all traffic between subnets is allowed
- By default not all subnets have access to the Internet. Either an Internet Gateway or NAT gateway is required for private subnets
- You can only have 1 Internet gateway per VPC
- A security group can stretch across different AZ's
- You can also create Hardware Virtual Private Network (VPN) connection between your corporate data center and your VPC and leverage the AWS cloud as an extension of your corporate data

could encounter issues
    - NAT instances must be in a public subnet
    - There must be a route out of the private subnet to the NAT instance in order for it to work
    - The amount of traffic that NAT instances support depend on the size of the NAT instance
    - If you are experiencing any sort of bottleneck issues with a NAT instance, then increase the instance size
    - HA can be achieved by using Auto-scaling groups, or multiple subnets in different AZ's with a scripted fail-over procedure
    - NAT instances are always behind a security group
- Network Address Translation (NAT) Gateway:
    - NAT Gateways scale automatically up to 10Gbps
    - There is no need to patch NAT gateways as the AMI is handled by AWS

    - Preferred in the Enterprise
    - No need to disable Source/Destination checks
- Network Access Control Lists (NACLS):
    - Numbered list of rules that are evaluated in order starting at the lowest numbered rule first to determine what traffic is allowed in or out depending on what subnet is associated with the rule
    - The highest rule number is 32766
    - Start with rules starting at 100 so you can insert rules if needed
    - Default NACL will allow ALL traffic in and out by default
    - You must assign a NACL to each subnet, if a subnet is not associated with a NACL, it will allow no traffic in or out
    - NACL rules are stateless, established in does not create outbound rule automatically

private IP addresses via a direct network route
    - Instances in either VPC can communicate with each other as if they are within the same network
    - You can create VPC peering connections between your own VPCs or with a VPC in another account within a SINGLE REGION
    - AWS uses existing infrastructure of a VPC to create a VPC peering connection. It is not a gateway nor a VPN, and does not rely on separate hardware
    - There is NO single point of failure for communication nor any bandwidth bottleneck
    - There is no transitive peering between VPC peers (Can't go through 1 VPC to get to another)
    - Hub and spoke configuration model (1 to 1)
    - Be mindful of IPs in each VPC. if multiple VPCs have the same IP blocks, they will not be

Resource or Operation	Default Limit	Comments
VPCs per region:	5	The limit for Internet gateways per region is directly correlated to this one. Increasing this limit will increase the limit on Internet gateways per region by the same amount.
Subnets per VPC:	200	

		attached to a VPC at a time.
Customer gateways	50	

per region:		
VPN connections per region:	50	
VPN connections	10	

gateway):		
Route tables per VPC:	5	Including the main route table. You can associate one route table to one or more subnets in a VPC.
Routes per route table (non-propagated routes):	50	This is the limit for the number of non-propagated entries per route table. You can submit a request for an increase of up to a maximum of 100; however, network performance may be impacted.
BGP advertised	5	You can have up to 100 propagated routes per route table; however, the total number of propagated and non-propagated entries per route table

routes):		advertise a default route.
Elastic IP addresses per region for each AWS account:	5	This is the limit for the number of VPC Elastic IP addresses you can allocate within a region. This is a separate limit from the Amazon EC2 Elastic IP address limit.
Security groups per VPC:	500	

rules per security group:		or decrease this limit, you can contact AWS Support — a limit change applies to both inbound and outbound rules. However, the multiple of the limit for inbound or outbound rules per security group and the limit for security groups per network interface cannot exceed 250. For example, if you want to increase the limit to 100, we decrease your number of security groups per network interface to 2.
Security groups per network interface:	5	If you need to increase or decrease this limit, you can contact AWS Support. The maximum is 16. The multiple of the limit for security groups per network interface and the limit for rules per security group cannot exceed 250. For example, if you want 10 security groups per network interface, we decrease your number of rules per security group to 25.

per instance:		
Network interfaces per region:	350	This limit is the greater of either the default limit (350) or your On-Demand instance limit multiplied by 5. The default limit for On-Demand instances is 20. If your On-Demand instance limit is below 70, the default limit of 350 applies. You can increase the number of network interfaces per region by contacting AWS Support, or by increasing your On-Demand instance limit.

Network ACLs per VPC:	200	You can associate one network ACL to one or more subnets in a VPC. This limit is not the same as the number of rules per network ACL.

		performance may be impacted due to the increased workload to process the additional rules.
Active VPC peering connections per VPC:	50	If you need to increase this limit, contact AWS Support . The maximum limit is 125 peering connections per VPC. The number of entries per route table should be increased accordingly; however, network performance may be impacted.
Outstanding VPC peering connection requests:	25	This is the limit for the number of outstanding VPC peering connection requests that you've requested from your account.

unaccepted VPC peering connection request:	nrs)	
VPC endpoints per region:	20	The maximum limit is 255 endpoints per VPC, regardless of your endpoint limit per region.
Flow logs per single eni, single subnet, or	2	You can effectively have 6 flow logs per network interface if you create 2 flow logs for the subnet, and 2 flow logs for the VPC in which your network interface resides. This limit cannot be increased.

NAT gateways per Availability Zone:	5	A NAT gateway in the pending, active, or deleting state counts against your limit.

For additional information about VPC Limits, see Limits in Amazon VPC

Compute:

EC2 (Elastic Compute Cloud):

Elastic Compute Cloud - Backbone of AWS, provides re-sizable compute capacity in the cloud. Reduces the time required to obtain and boot new server instances to minutes allowing you to quickly scale capacity, both up and down, as your computing requirements change.

- This topic is covered in AWS Solutions Architect Study Guide
- Once an Instance has been launched with instance store storage, you can not attach additional instance store volumes after the instance is launched, only EBS volumes
- When using an instance store volume, you can not stop the instance (the option to do so will not be available, as the instance moves to another host and and would cause complete data loss)

- By default both Root volumes will be deleted on termination, however you can tell AWS to keep the root device volume on a new instance during launch
- You can poll an instances meta-data by using curl http://169.254.169.254/latest/meta-data/
- You can get an instance's IP address by using curl http://169.254.169.254/latest/meta-data/public-ipv4
- No such thing as user-data, remember its always meta-data not user-data
- Can not encrypt root volumes, but you can encrypt any additional volumes that are added and attached to an EC2 instance.
- You can have up to 10 tags per EC2 instance
- AWS does not recommend ever putting RAID 5's on EBS
- When configuring a launch configuration for an auto-scaling group, the Health Check Grace Period is the period of time to ignore health checks while instances or auto-scaled instances are

    - You can only assign an EC2 role to an instance on create. You can not assign a role after the instance has been created and/or is running
    - You can change the permissions on a role post creation, but can NOT assign a new role to an existing instance
    - Role permissions can be changed, but not swapped
    - Roles are more secure then storing your access key and secret key on individual EC2 instances
    - Roles are easier to manager, You can assign a role, and change permissions on that role at any time which take effect immediately
    - Roles can only be assigned when that EC2 instance is being provisioned
    - Roles are universal, you can use them in any region
- Instance sizing:

    - C4 - Compute Optimized - CPU Intensive Apps/DBs
    - C3 - Compute Optimized - CPU Intensive Apps/DBs
    - R3 - Memory Optimized - Memory Intensive Apps/DBs
    - G2 - Graphics / General Purpose - Video Encoding/Machine Learning/3D App Streaming
    - I2 - High Speed Storage - NoSQL DBs, Data Warehousing
    - D2 - Dense Storage - Fileservers/Data Warehousing/Hadoop
    - D - Density
    - I - IOPS
    - R - RAM
    - T - Cheap General Purpose
    - M - Main General Purpose
    - C - Compute

    - Instance store (Ephemeral):
        - Also referred to as ephemeral storage and is not persistent
        - Instances using instance store storage can not be stopped. If they are, data loss would result
        - If there is an issue with the underlying host and your instance needs to be moved, or is lost, Data is also lost
        - Instance store volumes cannot be detached and reattached to other instances; They exist only for the life of that instance
        - Best used for scratch storage, storage that can be lost at any time with no bad ramifications, such as a cache store
    - EBS (Elastic Block Storage):
        - Elastic Block Storage is persistent storage that can be used to procure storage to EC2 instances

        - Default action for EBS volumes is for the root EBS volume to be deleted when the

instance is terminated
        - By default, ROOT volumes will be deleted on termination, however with EBS volumes only, you can tell AWS to keep the root device volume
        - EBS backed instances can be stopped, you will NOT lose any data
        - EBS volumes can be detached and reattached to other EC2 instances 3 Types of available EBS volumes can be provisioned and attached to an EC2 instance:
            - General Purpose SSD (GP2):
                - General Purpose up to 10K IOPS
                - 99.999% availability
                - Ratio of 3 IOPS per GB with up to 10K IOPS and ability to burst
                - Up to 3K IOPS for short periods for volumes under 1GB

                - Use if need more than 10K IOPS
            - Magnetic (Standard)
                - Lowest cost per GB
                - Ideal for workloads where data is accessed infrequently and apps where the lowest cost storage is important.
                - Ideal for fileservers
        - Encryption:
            - Root Volumes cannot be encrypted by default, you need a 3rd party utility
            - Other volumes added to an instance can be encrypted.
- AMIs:
    - AMI's are simply snapshots of a root volume and is stored in S3
    - AMI's are regional. You can only launch an AMI from the region in which it was stored

    - Permissions that control which AWS accounts can use the AMI to launch instances
    - When you create an AMI, by default its marked private. You have to manually change the permissions to make the image public or share images with individual accounts
    - Block device mapping that specifies volumes to attach to the instance when its launched
    - Hardware Virtual Machines (HVM) AMI's Available
    - Paravirtual (PV) AMI's Available
    - You can select an AMI based on:
        - Region
        - OS
        - Architecture (32 vs. 64 bit)
        - Launch Permissions
        - Storage for the root device (Instance Store Vs. EBS)

    - You can not set any deny rules in security groups, you can only set allow rules
    - There is an implicit deny any any at the end of the security group rules
    - You don't need outbound rules for any inbound request. Rules are stateful meaning that any request allowed in, is automatically allowed out
    - You can have any number of EC2 instances associated with a security group
- Snapshots:
    - You can take a snapshot of a volume, this will store that volumes snapshot on S3
    - Snapshots are point in time copies of volumes
    - The first snapshot will be a full snapshot of the volume and can take a little time to create
    - Snapshots are incremental, which means that only the blocks that have changes since your last snapshot are moved to S3
    - Snapshots of encrypted volumes are encrypted automatically

    - Snapshots can be shared with other AWS accounts or made public in the market place again as long as they are NOT encrypted
    - If you are making a snapshot of a root volume, you should stop the instance before taking the snapshot
- RAID Volumes:

- If you take a snapshot, the snapshot excludes data held in the cache by applications or OS. This tends to not be an issue on a single volume, however multiple volumes in a RAID array, can cause a problem due to interdependencies of the array
- Take an application consistent snapshot
  - Stop the application from writing to disk
  - Flush all caches to the disk
- Snapshot of RAID array --> 3 Methods:
  - Freeze the file system

- Placement Groups:
  - A logical group of instance in a single AZ
  - Using placement groups enables applications to participate in a low latency, 10Gbps network
  - Placement groups are recommended for applications that benefit from low network latency, high network throughput or both
  - A placement group can't span multiple AZ's so it is a SPoF.
  - Then name you specify for a placement group must be unique within your AWS account
  - Only certain types of instances can be launched in a placement group. Computer Optimized, GPU, Memory Optimized, and Storage Optimized.
  - AWS recommends that you use the same instance family and same instance size within the instance group.
  - Your [illegible] [illegible]

AMI into a placement group

- Pricing Models:
  - On Demand:
    - Pay fixed rate by the hour with no commitment
    - Users that want the low cost and flexibility of EC2
    - Apps with short term, spiky or unpredictable workloads that cannot be interrupted
    - Apps being developed or tested on EC2 for the first time
  - Reserved:
    - Provide capacity reservation and offer significant discount on the hourly charge for an instance (1-3 year terms)
    - Applications have steady state, or predictable usage
    - Apps that require reserved capacity

    - Bid whatever price you want for instance capacity by the hour
    - When your bid price is greater than or equal to the spot price, your instance will boot
    - When the spot price is greater than your bid price, your instance will terminate with an hours notice.
    - Applications have flexible start and end times
    - Apps that are only feasible at very low compute prices
    - Users with urgent computing needs for large amounts of additional capacity
    - If the spot instance is terminated by Amazon EC2, you will not be changed for a partial hour of usage
    - If you terminate the instance yourself you WILL be charged for any partial hours of usage.

- Install the AWSCLI tools or use the Amazon AMI to have access to the Amazon Command line tools
- Create a user in IAM, download the access key/secret access key
- Use the **aws configure** command to configure the CLI tools to interface with your amazon account using the IAM user access key/secret access key, and default region (Default output format can be left blank)
- Configured credentials can be found in ~/.aws/credentials

- Region and other configuration parameters can be found in ~/.aws/config
- Common CLI commands
    - aws configure: Use to configure the command line tools to access your amazon account
    - aws s3 ls - List all buckets that are associated with your AWS account

    - PHP:
        - From the instance that you want to install the SDK install composer (curl -sS https://getcomposer.org/installer | php)
        - Install the SDK using composer in the web directory which is usually /var/www/html (php composer.phar require aws/aws-sdk-php)

Resource or Operation	Default Limit
Elastic IP addresses for EC2-Classic:	5

Rules per security group for EC2-Classic:	100
Key pairs:	5000
On-Demand instances:	Varies based on instance type
Spot Instances:	Varies based on instance type
Reserved Instances:	20 instance reservations per Availability Zone, per month
Dedicated Hosts:	Up to 2 Dedicated Hosts per instance family, per region can be allocated

Throttle on the emails that can be sent :	Throttle applied
Tags per EC2 instance:	10

ELB (Elastic Block Storage Limits)

Resource or Operation	Default Limit
Number of EBS volumes:	5000

Total volume storage of General Purpose SSD (gp2) volumes:	20 TiB
Total volume storage of Provisioned IOPS SSD (io1) volumes:	20 TiB
Total volume storage of Throughput Optimized HDD (st1):	20 TiB
Total volume storage of Cold HDD (sc1):	20 TiB
Total volume storage of Magnetic volumes:	20 TiB
Total provisioned IOPS:	40,000

For additional information about EC2 Limits, see Limits in Amazon EC2

Resource or Operation	Default Limit
Number of EBS volumes:	5000
Number of EBS snapshots:	10,000
Total volume storage of General Purpose SSD (gp2) volumes:	20 TiB
Total volume storage of Provisioned IOPS SSD (io1) volumes:	20 TiB
Total volume storage of Throughput Optimized HDD (st1):	20 TiB

Total provisioned IOPS:	40,000

For additional information about EC2 Limits, see Limits in Amazon EC2

ELB (Elastic Load Balancer)

Elastic Load Balancing offers two types of load balancers that both feature high availability, automatic scaling, and robust security. These include the Classic Load Balancer that routes traffic based on either application or network level information, and the Application Load Balancer that routes traffic based on

- This topic is covered in AWS Solutions Architect Study Guide
- When configuring ELB health checks, bear in mind that you may want to create a file like healthcheck.html or point the ping path of the health check to the main index file in your application
- Remember the health check interval is how often a health check will occur
- Your Healthy/Unhealthy thresholds are how many times either will check before marking the origin either healthy or unhealthy
    - Health Check Interval: 10 seconds
    - Unhealthy Threshold: 2
    - Healthy Threshold: 3
    - This means that if the health check interval occurs twice without success, then the source will be marked as unhealthy. This is 2 checks @ 10 seconds per check, so basically after 20

origin will be marked as healthy.

- Enable Cross-Zone Load Balancing will distribute load across all back-end instances, even if they exist in different AZ's
- ELBs are NEVER given public IP Addresses, only a public DNS name
- ELBs can be In Service or Out of Service depending on health check results
- Charged by the hour and on a per GB basis of usage
- Must be configured with at least one listener
- A listener must be configured with a protocol and a port for front end (client to ELB connection), as well as a protocol and port for backed end (ELB to instances connection)
- ELBs support HTTP, HTTPS, TCP, and SSL (Secure TCP)
- ELBs support all ports (1-65535)
- ELBs do not support multiple SSL certificates

    - 443 (HTTPS)

- 465 (SMTPS)
- 587 (SMTPS)
- 1024-65535

- HTTP Error Codes:
  - 200 - The request has succeeded
  - 3xx - Redirection
  - 4xx - Client Error (404 not found)
  - 5xx - Server Error

Application Load Balancer Limit	Default Limit

Target groups per region:	50
Listeners per load balancer:	10
Targets per load balancer:	1000
Subnets per Availability Zone per load balancer:	1
Security groups per load balancer:	5
Rules per load balancer (excluding defaults:	10
No. of times a target can be registered per LB:	100

Targets per target group:	1000

Classic Load Balancer Limit	Default Limit
Load balancers per region:	20
Listeners per load balancer:	100
Subnets per Availability Zone per load balancer:	1
Security groups per load balancer:	5

Load Balancers per Region Limit NOTE:
This limit includes both your Application load balancers and your Classic load balancers. This limit can be increased upon request.

Elastic Beanstalk:

AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as

- Elastic Beanstalk is free, however any resources that are used in conjunction with the service are subject to normal pricing

- Predefined Configuration:
  - IIS
  - Node.js
  - PHP
  - Python
  - Ruby
  - Tomcat

- Python

Resource or Operation	Default Limit
Applications:	1000
Application Versions:	1000
Environments:	500

S3 (Simple Storage Service):

Amazon Simple Storage Service (Amazon S3), provides developers and IT teams with secure, durable, highly-scalable cloud storage. Amazon S3 is easy to use object storage, with a simple web service interface to store and retrieve any amount of data from anywhere on the web.

- This topic is covered in AWS Solutions Architect Study Guide
- Object based storage only for files, can not install OS or applications

- Files are stored flatly in buckets, Folders don't really exist, but are part of the file name
- S3 bucket names have a universal name-space, meaning each bucket name must be globally unique
- S3 Stores data in alphabetical order (lexigraphical order)
- S3 URL structures are region/amazon.aws.com/bucketname (https://s3-eu-west-1.amazonaws.com/myawesomebucket)
- Read after write consistency for PUTS of new objects (As soon as you write an object, it is immediately available)
- Eventual consistency for overwrite PUTS and DELETES. (Updating or deleting an object could take time to propagate)
- S3 is basically a key value store and consists of the following:
  - Key - Name of the object

  - ACLs - Permissions for stored objects
- Amazon guarantees 99.99% availability for the S3 platform
- Amazon guarantees 99.999999999% durability for S3 information (11 x 9's)
- Tiered storage, and life-cycle management available
- Versioning is available but must be enabled. It is off by default
- Offers encryption, and allows you to secure the data using ACLs
- S3 charges for storage, requests, and data transfer
- Bucket names must be all lowercase, however in US-Standard if creating with the CLI tool, it will allow capital letters

- The transfers tab shows uploads, downloads, permission changes, storage class changes, etc..
- When you upload a file to S3, by default it is set private
- You can transfer files up to 5GB using PUT requests

- S3 Events include SNS, or SQS events or Lambda functions. Lambda is location specific, not available in South Korea
- All storage tiers have SSL support, millisecond first byte latency, and support life-cycle management policies.
- Storage Tiers:
  - Standard S3:
    - Stored redundantly across multiple devices in multiple facilities
    - Designed to sustain the loss of 2 facilities concurrently
    - 11-9's durability, 99.99% availability
  - S3-IA (Infrequently Accessed):
    - For data that is accessed less frequently, but requires rapid access when needed
    - Lower fee than S3, but you are charged a retrieval fee

  - Reduced Redundancy Storage (RRS):
    - Use for data such as thumbnails or data that could be regenerated
    - Costs less than Standard S3
    - Designed to provide 99.99% durability and 99.99% availability of objects over a year
    - Designed to sustain the loss of a single facility
  - Glacier:
    - Very cheap, Stores data for as little as $0.01 per gigabyte, per month
    - Optimized for data that is infrequently accessed. Used for archival only
    - It takes 3-5 hours to restore access to files from Glacier
- Versioning and Cross-Region Replication (CRR):
  - Versioning must be enabled in order to take advantage of Cross-Region Replication
  - Versioning resides under Cross Region Replication tab
  - Once Versioning is turned on, it can not be turned off, it can only be suspended

  - When versioning is enabled, you will see a slider tab at the top of the console that will enable you to hide/show all versions of files in the bucket
  - If a file is deleted for example, you need to slide this tab to show in order to see previous versions of the file
  - With versioning enabled, if you delete a file, S3 creates a delete marker for that file, which tells the console to not display the file any longer
  - In order to restore a deleted file you simply delete the delete marker file, and the file will then be displayed again in the bucket
  - To move back to a previous version of a file including a deleted file, simply delete the newest version of the file or the delete marker, and the previous version will be displayed
  - Versioning does store multiple copies of the same file. So in the example of taking a 1MB file, and uploading it. Currently your storage usage would be 1MB. Now if you update the

1MB file, so your total S3 usage is now 2MB not 1MB
  - Versioning does NOT support de-duplication or any similar technology currently
  - For Cross Region Replication (CRR), as long as versioning is enabled, clicking on the tab will now give you the ability to suspend versioning, and enable cross region replication
  - Cross Region Replication (CRR) has to be enabled on both the source and destination buckets in the selected regions
  - Destination bucket must be created and again globally unique (can be created right from the versioning tab, in the CRR configuration section via button)
  - You have the ability to select a separate storage class for any Cross Region Replication destination bucket
  - CRR does NOT replicate existing objects, only future objects meaning that only objects stored post turning the feature on will be replicated

This will use MFA to provide additional security against object deletion

- Life-cycle Management:
  - When clicking on Life-cycle, and adding a rule, a rule can be applied to either the entire bucket or a single 'folder' in a bucket
  - Rules can be set to move objects to either separate storage tiers or delete them all together
  - Can be applied to current version and previous versions
  - If multiple actions are selected for example transition from STD to IA storage 30 days after upload, and then Archive 60 days after upload is also selected, once an object is uploaded, 30 days later the object will be moved to IA storage. 30 days after that the object will be moved to glacier.
  - Calculates based on UPLOAD date not Action data
  - Transition from STD to IA storage class requires MINIMUM of 30 days. You can not select

because the minimum for STD->IA is 30 days, and the transition to glacier then takes an additional 30 days
  - When you enable versioning, there will be 2 sections in life-cycle management tab. 1 for the current version of an object, and another for previous versions
  - Minimum file size for IA storage is 128K for an object
  - Can set policy to permanently delete an object after a given time frame
  - If versioning is enabled, then the object must be set to expire, before it can be permanently deleted
  - Can not move objects to Reduced Redundancy using life-cycle policies
- S3 Transfer Acceleration:
  - Utilizes the CloudFront Edge Network to accelerate your uploads to S3
  - Instead of uploading directly to your S3 bucket, you can use a distinct URL to upload

  - There is a test utility available that will test uploading direct to S3 vs through Transfer Acceleration, which will show the upload speed from different global locations
  - Turning on and using Transfer Acceleration will incur an additional fee
- 2 types of encryption available:
  - In transit:
    - Uses SSL/TLS to encrypt the transfer of the object
  - At Rest (AES 256):
    - Server Side: S3 Manged Keys (SSE-S3)
    - Server Side: AWS Key Management Service, Managed Keys (SSE-KMS)
    - Server Side: Encryption with Customer provided Keys (SSE-C)
    - Client Side Encryption
- Pricing (What your charged for when using S3):

  - Data Transfer

Developer Associate Specific Topics

- Web Hosting:
  - Used for static hosting only; Server side code will not execute
  - Don't need to worry about scaling, ELBs or number of instances, S3 handles all of that for you
  - When you create an S3 bucket or enable hosting, you still need to make sure that either the files or the entire bucket are set to public accesibility
  - Bucket URLs are structured such as http://s3-eu-west-

1.amazonaws.com
  - Hosting sites on S3 does not allow HTTPS support

    - Sites hosted on S3 can be served via HTTPS if distributed by cloudfront; Cloudfront would be configured to terminate a client HTTPS requst, and then talk to the bucket via standard HTTP
    - Can be configured to redirect to another URL
- CORS Configuration:
    - Cross Origin Resource Sharing (CORS)
    - Configured in the permisssions section of the properties tab in a bucket
    - CORS configuration is in XML format and will be pasted directly into the permissions
    - CORS is required if you are calling an asset that resides in another bucket from the bucket that your static site resides in using the hosted URL

Buckets per account:	100
Largest files size you can transfer with PUT request:	5GB
Minimum file size:	1 byte
Maximum file size:	5 TB

For additional information about API Gateway Limits, see Limits in Amazon S3

Amazon CloudFront is a global content delivery network (CDN) service that accelerates delivery of your websites, APIs, video content or other web assets.

- This topic is covered in AWS Solutions Architect Study Guide
- Edge Location is the location where content will be cached, separate from an AWS Region/AZ
- Origin is the origin of all files, can be S3, EC2 instance, a ELB, or Route53
- Distribution is the name given to the CDN which consists a collection of edge locations
- Web Distributions are used for websites
- RTMP - (Real-Time Messaging Protocol) used for streaming media typically around adobe flash files
- Edge locations can be R/W and will accept a PUT request on an edge location, which then will replicate the file back to the origin

- When enabling cloudfront from an S3 origin, you have the option to restrict bucket access; this will disable the direct link to the file in the S3 bucket, and ensure that the content is only served from cloudfront
- The path pattern uses regular expressions
- You can restrict access to your distributions using signed URLS
- You can assign Web Application Firewall rules to your distributions
- Distribution URLs are going to be non-pretty names such as random_characters.cloudfront.com; you can create a CNAME that points to the cloudfront name to make the URL user friendly
- You can restrict content based on geographical locations in the behaviors tab
- You can create custom error pages via the error pages tab
- Purging content is handled in the Invalidations tab

Data transfer rate per distribution:	40 Gbps
Requests per second per distribution:	100,000
Web distributions per account:	200

RTMP distributions per account:	100
Alternate domain names (CNAMEs) per distribution:	100
Origins per distribution:	25

White-listed cookies per cache behavior:	10
SSL certificates per account when serving HTTPS requests using dedicated IP addresses (no limit when serving HTTPS requests using SNI):	2
Custom headers that you can have Amazon CloudFront forward to the origin:	10 name–value pairs

For additional information about CloudFront Limits, see Limits in Amazon CloudFront

DynamoDB (No-SQL):

Fast and flexible NoSQL DB service for all apps that need consistent, single-digit millisecond latency at any scale. It is a fully managed database and supports both document and key-value data models. Its flexible data model and reliable performance make it a great fit for mobile, web, gaming, ad-tech, IoT, and many other applications.

- This topic is covered in AWS Solutions Architect Study Guide
- Non Relational DB (No-SQL), comprised of collections (tables), of documents (rows), with each

- RDS is not so easy, you usually have to use a bigger instance size or add read replicas
- Stored on SSD Storage
- Spread across 3 geographically distinct data centers
- Eventual Consistent Reads (Default)
  - Consistency across all copies of data is usually reached within 1 second
  - Repeating a read after a short time should return updated data
  - Best Read Performance
- Strongly Consistent Reads
  - Returns a result that reflects all writes that received a successful response prior to the read
- Structure:
  - Tables
  - Items (Think rows in a traditional table)

- Read throughput 0.0065 per hour for every 50 units
- First 25 GB of storage is free
- Storage costs of 25 cents per additional GB per Month
- Can be expensive for writes, but really really cheap for reads
- The combined key/value size must not exceed 400 KB for any given document

Developer Associate Specific Topics

- Supports attribute nesting up to 35 levels
- Conditional writes are idempotent, you can send the same conditional write request multiple

of an existing attribute without interfering with other write requests
- Atomic counter updates are not idempotent, the counter will increment each time you call UpdateItem
- If you can have a small margin of error in your data, then use atomic counters
- If your application needs to read multiple items, you can use the BatchGetItem API endpoint; A single request can retrieve up to 1MB of data with as many as 100 items
- A single BatchGetItem request can retrieve items from multiple tables
- All write requests are applied in the order in which they are received
- Pricing (calculate the amount of writes and reads per second):
    - Divide total number of writes per day / 25 (hours) / 60 (minutes) / 60 (seconds) = No. writes per second
    - A write or read capacity unit can handle 1 write/read per second

- 1,000,000 writes per day = 1,000,000/24 = 41,666.67
- 41,666.67 / 60 (minutes) = 694.44
- 694.44 / 60 (seconds) = 11.574 writes per second
- This example would require 12 write capacity units (single capacity unit is 1 write per second)
- Charge for write is $0.0065 per 10 units
- $0.0065 / 10 = $0.00065 per unit
- $0.00065 * 12 (required write units) = $0.0078
- $0.0078 * 24 (hours per day) = $0.1872 per day for writes
- Charge for read is $0.0065 per 50 units
- $0.0065 / 50 = $0.00013 per unit
- $0.00013 * 12 (required read units) = $0.00156

- Indexes:
    - Primary Key types:
        - Single attribute (unique ID):
            - Partition Key (Hash Key composed of one attribute)
            - Partition Key's value is used as input to an internal hash function which output determines the partition (physical location in which the data is stored)
            - No 2 items in a table can have the same partition key value
        - Composite (unique ID and date range):
            - Partition Key & Sort Key (Hash and Range) composed of two attributes
            - Partition Key's value is used as input to an internal hash function which output determines the partition (physical location in which the data is stored)

sort key value
    - Local Secondary Index (LSI):
        - Has the SAME partition key, but different sort key
        - Can ONLY be created when creating a table
        - Can not be removed or modified after creation
        - Can have up to 5 LSI's per table
    - Global Secondary Index (GSI):
        - Has DIFFERENT partition key and different sort key
        - Can be created at table creation or added LATER
        - Can have up to 5 GSI's per table
- Streams:
    - Used to capture any kind of modification of the DynamoDB tables

were modified in the item
    - If an item is deleted from the table, the stream captures an image of the entire item before it was deleted
    - Streams are stored for 24 hours and then is lost
    - Streams can trigger functions with Lambda that will perform actions based on the instantiation of a stream event
- Query's:
    - Operation that finds items in a table using only the primary key attribute value
    - Must provide a partition attribute name and distinct value to search for
    - Optionally can provide a sort key attribute name and value and use comparison operator to refine the search results
    - By default a query returns all of the data attributes for items with the specified primary

    - Results are always sorted by the sort key
    - If the data type of the sort key is a number, the results are returned in numeric order
    - If the data type of the sort key is a string, the results are returned in order of ASCII character code values
    - Sort order is ascending, the ScanIndexForward parameter can be set to false to sort in descending order
    - By default queries are eventually consistent but can be changed to strongly consistent
    - More efficient then a scan operation
    - For quicker response times, design your tables in a way that can use the query, GET, or BatchGetItem API
- Scans:
    - Examines every item in the table

    - Always scans the entire table, then filters out values to provide the desired result (added step of removing data from initial dataset)
    - Should be avoided on a large table with a filter that removes many results
    - As table grows, the scan operation slows
    - Examines every item for the requested values, and can use up provisioned throughput for a large table in a single operation
- Provisioned Throughput
    - 400 HTTP status code - ProvisionedThroughputExceededException error will indicate that you exceeded your max allowed provisioned throughput for a table or for one or more GSI's
    - Unit of read provisioned throughput:
        - All reads are rounded up to increments of 4 KB
        - Eventual consistent reads (default) consist of 2 reads per second

        - Divide by 2 if eventually consistent
        - Example:
            - Application requires to read 10 items of 1 KB per second using eventual consistency, whats the read throughput
            - Calculate the number of read units per item needed
            - 1 KB rounded to the nearest 4 KB increment = 4 (KB) or a single chunk
            - 4 KB / 4 KB = 1 read unit per item
            - 1 x 10 read items = 10
            - Using eventual consistency is 10 /2 = 5
            - 5 units of read throughput
        - Example 2:
            - Application requires to read 10 items of 6 KB per second using eventual

            - 8 KB rounded to the nearest 4 KB increment = 8 (KB) or 2 chunks of 4 KB
            - 8 KB / 4 KB = 2 read unit per item
            - 2 x 10 read items = 20

        - Using eventual consistency is 20 /2 = 10
        - 10 units of read throughput
    - Unit of write provisioned throughput:
        - All writes are 1 KB
        - All writes consist of 1 write per second
        - Example:
            - Application requires to write 5 items with each being 10KB in size per second
            - Each write unit consists of 1 KB of data, need to write 5 items per second with each item using 10 KB of data

            - Application requires to write 12 items with each being 100KB in size per second
            - Each write unit consists of 1 KB of data, need to write 12 items per second with each item using 100 KB of data
            - 12 items * 100 KB = 1200 write units
            - Write throughput is 1200 units
- Web Identity Providers:
    - Authenticate users using Web Identity Providers such as Facebook, Google, Amazon or any other ID Connect-compatible identity provider
    - Accomplished using AssumeRoleWithWebIdentity API
    - Need to create a role first
    - Process:

        - Token, App ID of provider, and ARN of IAM Role sent to AssumeRoleWithIdentity API endpoint
        - AWS issues temporary security credentials back to the user allowing the user to access resources (1 hour default)
        - Temporary security credentials response consist of 4 things:
            - AccessKeyID, SecretAccessKey, SessionToken
            - Expiration (time limit, 1 hour by default)
            - AssumeRoleID
            - SubjectFromWebIdentityToken

US East (N. Virginia) Region	Default Limit

Maximum capacity units per account:	80,000 read capacity units and 80,000 write capacity units

All Region Resource or Operation	Default Limit
Maximum capacity units per table or global secondary index:	10,000 read capacity units and 10,000 write capacity units
Maximum capacity units per account:	20,000 read capacity units and 20,000 write

For additional information about DynamoDB Limits, see Limits in Amazon DynamoDB

Security and Identity:

IAM (Identity and Access Management):

- This topic is covered in AWS Solutions Architect Study Guide
- Allows for centralized control and shared access to your AWS Account and/or AWS services
- By default when you create a user, they have NO permissions to do anything
- Root account has full admin access upon account creation
- Not region specific, can be shared between all regions
- Granular permission sets for AWS resources
- Includes Federation Integration which taps into Active Directory, Facebook, Linkedin, etc. for authentication
- Multi-factor authentication support
- Allows configuration of temporary access for users, devices and services
- Set up and manage password policy and password rotation policy for IAM users
- Integration with many different AWS services

    - Users - End users (people)
    - Groups - Collection of users under one set of permissions
    - Roles - Assigned to AWS resources, specifying what the resource (such as EC2) is allowed to access on another resource (S3)
    - Policies - Document that defines one or more permissions
- Policies can be applied to users, groups and roles
- You can assign up to 10 policies to a single group
- Policy documents must have a version, and a statement in the body; The statement must consist of Effects (Allow, Deny), Actions(Which action to allow/deny such a * for all actions), and Resources (affected resources such as * for all resources)
- All resources can share the same policy document
- There are 3 different types of roles:

with the current AWS account

    - Identity provider access roles
        - Roles for facebook or similar Identity providers
- In order for a new IAM user to be able to log into the console, the user must have a password set
- By default a new users access is only accomplished through the use of the access key/secret access key
- If the users password is a generated password, it also will only be shown at the time of creation.
- Customizable Console Sign-in link can be configured on the main IAM page (aws.yourdomain.com)
- Customizable Console Sign-in links must be globally unique. If a sign in link name is already taken, you must choose an alternative
- Root account is email address that you used to register your account
- Recommended that root account is not used for login, and should be secured with Multi-factor

- Access Key ID is equivalent to a user-name, Secret Access Key is equivalent to a password
- When creating a user's credentials, you can only see/download the credentials at the time of creation not after.
- Access Keys can be retired, and new ones can be created in the event that secret access keys are lost
- To create a user password, once the users have been created, choose the user you want to set the password for and from the User Actions drop list, click manage password. Here you can opt to create a generated or custom password. If generated, there is an option to force the user to set a custom password on next login. Once a generated password has been issued, you can see the password which is the same as the access keys. Its shown once only

- Click on Policies from the left side menu and choose the policies that you want to apply to your users. When you pick a policy that you want applied to a user, select the policy, and then from the

Resource or Operation	Default Limit
Groups per account:	100
Instance profiles:	100
Roles:	250
Server Certificates:	20
Users:	5000

For additional information about API Gateway Limits, see Limits in IAM entities and objects

Directory Service:

AWS Directory Service makes it easy to setup and run Microsoft Active Directory (AD) in the AWS cloud, or connect your AWS resources with an existing on-premises Microsoft Active Directory.

- Active Directory flow is initialted when a user browses to the ADFS integrated site:
  - The sign-on page will authenticate the user against Active Directory
  - Depending on the browser used, the user may be prompted to input their AD

form of an auth response from AD Federation Services (ADFS)
  - The users browser posts the SAML assertion to the AWS sign-in endpoint for SAML
  - The AWS console uses the AssumeRoleWithSAML API to request temporary security credentials and then constructs a sign-in URL for the AWS Console
  - The users browser receives the sign-in URL and is redirected to the conole
  - The process is transparent to the user; They start at an interanl web site and end up on the console without having to supply credentials
  - Remember the API call to request temporary security credentials from the AWS platform is **AssumeRoleWithSAML**
  - The sign-in endpoint for SAML is https://signin.aws.amazon.com/saml
- When using ADFS the user always authenticates with AD first before recieving security credentials

Web Identity Federation:

  - The AWS console has a link to a Web Identity Federation Playground, that allows you to test logins using services such as FB, LinkedIn, etc...
  - Once logged in using the playground, you get a response containing an accessToken that is good for 5016 seconds
  - This went to the service such as FB, authenticated with the service, recieved an accessToken, using the token, AWS will grant temporary security credentials by making an AssumeRoleWithWebIdentity request
  - When the AssumeRoleWithWebIdentity request is formed, a trust policy is created granting all access via the recieved accessToken
  - Authenticate with web service (facebook, etc..) first, then get temporary security credentials via AccessToken sent to AssumeRoleWithWebIdentity request, and finally with the temporary security credentials, user is able to access AWS resources

Management Tools:

Cloud Formation:

AWS CloudFormation gives developers and systems administrators an easy way to create and manage a collection of related AWS resources, provisioning and updating them in an orderly and predictable fashion.

- Can be written in either JSON or YAML
- Free service, however resources created by CloudFormation are subject to normal pricing
- Fn::GetAtt is a function that will allow you to get attributes of newly launched resources from CloudFormation such as DNS name, IP address, etc..
- If errors are encountered the stack launch will terminate and rollback all resources that were created

Resource or Operation	Default Limit
Stacks:	200

Application Services:

SQS (Simple Queue Service):

Web service that gives you access to a message queue that can be used to store messages while waiting for a computer to process them. SQS is a distributed queue system that enables applications to quickly and reliably queue messages that one component of the application generates to be consumed by another component. A queue is a temp repository for messages that are awaiting processing

- Used to allow customers the ability to decouple infrastructure components
- Very first service AWS released. Even older then EC2
- Messages can contain up to 256 KB of text in any format
- Acts as a buffer between the component producing and saving data, and the component receiving and processing the data
- Ensures deliver of each message at least once and supports multiple readers and writers interacting with the same queue
- A single queue can be used simultaneously by many distributed application components, with no need for those components to coordinate or communicate with each other
- Will always be available and deliver messages
- Does not guarantee FIFO delivery of messages
- Messages can be delivered multiple times and in any order

- SQS always asynchronously PULLs messages from the queue
- Retention period of 14 days
- 12 hour visibility timeout by default

- If you find that the default visibility timeout period (12 hours) is insufficient to fully process and delete the message, the visibility timeout can be extended using the ChangeMessageVisibility action
- If the ChangeMessageVisibility action is specified to set an extended timeout period, SQS restarts the timeout period using the new value
- Engineered to provide delivery of all messages at least one
- Default short polling will return messages immediately if messages exist in the queue
- Long polling is a way to retrieve messages from a queue as soon as they are available; long polling requests don't return a response until a message arrives in the queue

- First million messages free, then $.50 per additional million thereafter
- Single request can have from 1 to 10 messages, up to a max payload of 256KB
- Each 64KB chunk of payload is billed as 1 request. If you send a single API request with a 256KB payload, you will be billed for 4 requests (256/64 KB chunks)
- "Decouple" = SQS on exam
- Auto-scaling supported
- Message prioritization is not supported
- Process:
  - Component 1 sends a message to the queue
  - Component 2 retrieves the message from the queue and starts the visibility timeout period
  - Visibility timer only starts when the message is picked up from the queue

  - The process is only complete when the queue receives the command to delete the message from the queue

For additional information about SQS Limits, see Limits in Amazon SQS

SWF (Simple Workflow Service)

Simple Workflow Service is a web service that makes it easy to coordinate work across distributed application components. Enabled for a range of uses such as media processing, web back ends, business

- This topic is covered in AWS Solutions Architect Study Guide
- Build, run and scale background jobs or tasks that have sequential steps
- Way to process human oriented tasks using a framework
- SQS has a retention period of 14 days, vs SWF has up to a 1 year for work-flow executions
- Workflow retention is always shown in seconds (3.1536E+07 seconds)
- "Task could take a month" = SWF, as SQS only has a 14 day retention
- Presents a task-oriented API, whereas SQS offers a message-oriented API
- Ensures a teaks is assigned only once and is never duplicated; SQS duplicate messages are allowed, and must be handled
- Keeps track of all tasks and events in an application, SQS would need an implementation of a custom application-level tracking mechanism

- You can register a domain by using the AWS console or using the RegisterDomain action in the SWF API
- Domain parameters are specified in JSON format
- SWF Actors:
  - Workflow starters - An application that can initiate a Workflow
  - Decider's - Control the flow or coordination of activity tasks such as concurrency, or scheduling in a work-flow execution; If something has finished in a work-flow (or fails), a

decider decides what to do next
    - Activity Workers - Programs that interact with SWF to get tasks, process received tasks, and return the results
- Brokers the interactions between workers and the decider; Allows the decider to get consistent views into the progress of tasks and to initiate new tasks in an ongoing manner
Stores tasks, assigns them to workers when they are ready and monitors their progress

execution state, and can run independently, with the ability to scale quickly

For additional information about SWF Limits, see Limits in Amazon SWF

Mobile Services:

SNS (Simple Notification Service):

from the cloud. It provides developers with a highly scalable, flexible, and cost-effective capability to publish messages from an application and immediately deliver them to subscribers or other applications.

- Web service that allows customers to setup, operate, and send notifications from the cloud
- Can push to Apple, Google, FireOS, and Windows devices, as well as Android devices in China with Baidu cloud push
- Follows the publish-subscribe (pub-sub) messaging paradigm, with notifications being delivered to clients using a push mechanism that eliminates the need to poll for updates
- Can deliver notifications by SMS, email, SQS queues, or any HTTP endpoint
- SNS notifications can be used to trigger lambda functions
- When a message is published to an SNS topic that has a lambda function subscribed to it, the

- Allows you to group multiple recipients using topics
- Topics are access points for allowing recipients to dynamically subscribe for copies of the notification
- One topic can support deliveries to multiple endpoint types, for example, IOS, Android, and SMS recipients can be grouped together
- When message is published, SNS delivers appropriately formatted copies of your message to each subscriber
- Email notifications will be JSON formated not XML
- Subscriptions have to be confirmed
- Subscription expire after 3 days if they are not confirmed
- TTL is the number of seconds since the message was published
- If the message is not delivered within the TTL time, then the message will expire

- Simple API and easy integration with applications
- Flexible message deliver over multiple transport protocols
- Inexpensive, pay as you go model
- Web based AWS management console offers simplicity of point and click interface
- $.50 per million SNS requests
- $.06 per 100,000 notification deliveries over HTTP
- $0.75 per 100 notifications over SMS
- $2.00 per 100,000 notification deliveries over email
- Can be used in conjunction with SQS to fan a single message out to multiple SQS queues
- Remember:
    - SNS - PUSH

- SNS - PUSH
- SQS - PULL (poll)

- HTTP/S
- Email
- Email-JSON
- SQS
- Application
- Lambda
- Messages can be customized for each of the available protocols

Resource or Operation	Default Limit
Topics :	100,000

Delivery rate for promotional SMS messages:	20 Messages per second
Delivery rate for transactional SMS messages:	20 Messages per second

White Paper Review:

- Shared security model
  - AWS:
    - Responsible for securing the underlying infrastructure

    - Infrastructure comprised of hardware, software, networking, and facilities that run AWS services
    - Responsible for the security configuration of its products that are considered managed services, such as DynamoDB, RDS, Redshift, Elastic MapReduce, lambda, and Workspaces.
  - User:
    - Responsible for anything put on the cloud
    - EC2, VPC, S3 security configuration and management tasks
    - Account Management (MFA, SSL, TLS, CloudTrail API/User activity logging)