Understanding AWS Core Services

Guided Notes

I am excited that you are on the journey to get your AWS Certified Cloud Practitioner certification. This guided outline is meant to complement the video course. Here are a few tips to help you get the most out of these resources:

1. Print this out before you start the video course.
2. Follow along with the course and fill out areas in this document as you watch the course. You'll notice that the module names in the course are the bold headings here in these notes. In addition, clips in the module have their titles in this document too. Not all clips have notes.
3. Review your notes against the completed notes that can be found in the exercise files.
4. Keep this document after you finish the course as a part of the materials you will use to study for the exam.

Remember, this course is just the first step in your journey to achieve this certification. Follow along with the remainder of courses in this path, and then register for the exam.

Don't forget to reach out on Twitter and LinkedIn to let me know how you are doing along the way.

Interacting with AWS

Learning Outcomes

- Interaction Methods
  - AWS Console
    - You should know what use cases would be best to be done within the AWS console
    - Know how to login to the console
  - AWS Command Line Interface (CLI)
    - You should know when it would make sense to leverage the CLI
    - Know where to find the installation instructions for your platform
  - AWS Software Development Kit (SDK)
    - Know when the use of the SDK makes sense

Links You'll Need

- AWS Console
- AWS CLI Installation Instructions
- AWS SDK's

Methods of Interacting with AWS

Three methods of interacting with AWS services:

1. ***AWS Console***
2. ***AWS CLI***
3. ***AWS SDK***

The AWS Management Console is a ***web*** and ***mobile app*** based interface for interacting with most all of the 150+ AWS services. All major browsers and mobile operating systems are supported.

The AWS SDK is supported in the following languages:

Java	*.NET*	*Node.js*
JavaScript (Browser)	*PHP*	*Python*
Ruby	*Go*	*C++*
Kotlin	Rust	Swift

Using the AWS CLI

Generating an access key:

Generating an access key is not required to use the AWS CloudShell, but it is required if you want to use the CLI on your machine outside of the browser.

1. Log into the AWS Console.
2. Select your username in the top bar and select My Security Credentials in the dropdown menu.
3. Next, select the Access Keys option.
4. Select the option to Create New Access Key (if this is a root account, you should delete these when you are done with them)
5. Download your key file

You can launch the AWS CloudShell using the icon in the top bar:

Scenarios

The following scenarios are presented in the course as a way to explore your understanding of the module. Include your answer here in this outline, as well as your notes on the solution to each scenario.

SCENARIO 1

- Roger's company runs several production workloads in AWS
- They have a new web application that manages digital assets for marketing
- They need to automatically create a user account in Amazon Cognito on sign-up
- They want this step seamlessly integrated into the application
- Which interaction method would Roger's company use for this?

What's Your Answer: ***Software Development Kit (SDK)***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 2

- Eliza's company is considering transitioning to AWS
- They want to leverage Amazon Relational Database Service
- Eliza wants to test out a single database on the service
- What interaction method would Eliza use for this use case?

What's Your Answer: ***AWS Console***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 3

- Jennifer's company is a startup
- They created a social network for entrepreneurs with a web and mobile app
- Jennifer has a set of tasks she needs to run on AWS each day to generate reports
- What interaction method would Jennifer use for this use case?

What's Your Answer: ***AWS Command Line Interface (CLI)***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

Module Wrap Up

Take a minute to write down any areas from this module that you don't fully understand or where you still have questions:

Compute Services

Learning Outcomes

- Understand the three different compute services that we introduced in this module:
  - Amazon EC2
    - Be able to define EC2 and what it does
    - Know what instance types are for EC2 and what capabilities they cover
    - Know when to use the different purchase types for EC2
    - Understand what an AMI is and what it provides to an EC2 instance
  - AWS Elastic Beanstalk
    - Be able to explain what Elastic Beanstalk is and how it differs from EC2
    - Know the different capabilities that are included with the service
  - AWS Lambda
    - Be able to define Lambda and explain how it differs from both EC2 and Elastic Beanstalk
    - Understand how you are charged for Lambda usage
    - Note that Lambda is the core of a serverless approach

Links You'll Need

- Amazon EC2
- AWS Elastic Beanstalk
- AWS Lambda
- AWS Elastic Beanstalk - Sample Applications

Amazon EC2 Overview

"Amazon EC2 is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers." - Amazon Web Services

The four concepts that we need to know to launch an EC2 instance are:

1. *Instance Types*
2. *Root Device Type*
3. *Amazon Machine Image (AMI)*
4. *Purchase Option*

The instance type defines the *processor*, memory, and *storage type*.

The two root device types for an EC2 instance are:

Instance store - Ephemeral storage that is physically attached to the host the virtual server is running on

Elastic Block Store (EBS) - Persistent storage that exists separately from the host the virtual server is running on

Amazon EC2 Purchase Types

Amazon EC2 Purchase Options

1. *On-demand* - You pay by the second for the instances that are launched

2. ***Reserved*** - You purchase discount EC2 instance types in advance for 1 or 3 years. This also provides a capacity reservation for the selected instance type.
3. ***Savings Plans*** - You commit to a level of usage for EC2, Fargate, or Lambda for a 1 or 3 year period at a discount. This does not reserve capacity.
4. ***Spot*** - You can leverage unused EC2 capacity in a region for a large discount.
5. ***Dedicated*** - You purchase the capacity for a dedicated physical server.

Reserved Instance Cost Models:

All Upfront - Entire cost for the 1 or 3 year period is paid upfront

Partial Upfront - Part of 1 or 3 year cost is paid upfront along with a reduced monthly cost

No Upfront - No upfront payment is made, but there will be a reduced monthly cost

Launching EC2 Instances

1. Log into the AWS Console.
2. Open the EC2 service dashboard (search for EC2 in the 'Search' input).
3. Select the **Launch Instance** option.
4. Enter the name of the instance you will be launching.
5. Use the **Amazon Linux 2 AMI**, which will likely be pre-selected for you.

6. Be sure that the **t2.micro** instance type is selected (it should be selected by default).
7. Next, you will need to select a Key Pair. If you don't have one, you can create one by pressing the **Create a New Key Pair** link.
8. Next, you will need to adjust the **Network settings** for the instance. For the **Allow SSH traffic from**, select the **My IP** option. Also, be sure the **Allow HTTP traffic from the Internet** option is selected.
9. Leave the storage settings with their default values.
10. Scroll down to **Advanced Details** and open these settings. In the **User data** field, enter the text included below these instructions.
11. Press the **Launch instance** button.
12. Once the instance launch has been successfully initiated, click on the ID link for the instance (it will start with "i-" followed by a collection of numbers and letters).
13. Once the instance has transitioned from pending to running, copy the public DNS into your browser. You should see the test page in your browser. Be sure to use http and not https to open the hostname.
14. Finally, back in the AWS console select the instance and then navigate to **Instance State**. Select **Terminate**. Confirm your decision.

User Data:

```
#!/bin/bash
yum install httpd -y
systemctl enable httpd
systemctl start httpd
```

AWS Elastic Beanstalk Overview

Elastic Beanstalk is a ***platform*** as a service solution on AWS.

Note the Supported Application Platforms for Elastic Beanstalk:

Java, .NET, PHP, Node.js, Python, Ruby, Go, Docker

Launching an App on Elastic Beanstalk

1. Navigate to the Elastic Beanstalk Tutorials and Samples page. Select a sample application to download to your local machine.
2. Log into the AWS console and navigate to the Elastic Beanstalk service page.
3. If you see the "Welcome to AWS Elastic Beanstalk" screen, select **Create Application**.

4. In the screen that follows, give your application a name and select the platform (it will need to be the same platform as the sample application you downloaded).
5. Select the option to upload your code, and then upload the zip file you downloaded that contains your sample application.
6. Select the option to **Create application**.
7. Wait for the application and then navigate to the URL near the top of the page.
8. After viewing the application, navigate back to the console and select **Actions - Terminate Environment.**

AWS Lambda Overview

"*AWS Lambda* lets you run code without *provisioning* or *managing* servers. You pay only for the compute time you consume. You can run code for virtually any type of application or backend service - all with zero administration." - Amazon Web Services

AWS Lambda is the primary service for *serverless* architectures.

Container Services

"*Containers* are packages of software that contain all of the necessary elements to run in any environment... *containers* virtualize the operating system and run anywhere, from a private data center to the public cloud or even on a developer's personal laptop." - Google

AWS App Runner - requires no prerequisite knowledge of containers or infrastructure.

Amazon EKS (Elastic Kubernetes Service) - ideal for organizations looking to run Kubernetes on AWS.

Amazon ECS (Elastic Container Service) - full-featured container orchestration solution that natively integrates with many AWS services.

AWS Fargate - serverless compute engine that reduces what you have to maintain. It works with both *ECS* and *EKS*.

Scenarios

The following scenarios are presented in the course as a way to explore your understanding of the module. Include your answer here in this outline, as well as your notes on the solution to each scenario.

SCENARIO 1

- Sylvia's company is in the process of moving multiple workloads into AWS
- One workload is an application that will be leveraged for at least 5 more years
- The organization is looking to be as cost efficient as possible for its EC2 usage
- What EC2 purchase option should be chosen for this application?

What's Your Answer: ***All Upfront Reserved – 3 years***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 2

- Edward is looking to deploy his PHP web application to a virtual server
- He doesn't have experience managing EC2 instances on AWS
- He needs the ability to scale this application to meet user demand
- What is the best compute option for Edward based on these criteria?

What's Your Answer: ***AWS Elastic Beanstalk***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 3

- Cindy's company is transitioning to the cloud for its data processing workloads
- These workloads happen daily and can start or stop without a problem
- This workload will be leveraged for at least one year
- What EC2 purchase option would be the most cost-efficient choice?

What's Your Answer: *Spot Instances*

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

Module Wrap Up

Take a minute to write down any areas from this module that you don't fully understand or where you still have questions:

Content and Network Delivery Services

Learning Outcomes

- Be able to explain the purpose of each of the following services:
  - Amazon Route 53
  - Amazon Virtual Private Cloud (VPC)
  - AWS Direct Connect
  - Amazon API Gateway
  - Amazon CloudFront
  - Elastic Load Balancing
- Be able to explain the differences between two cloud scaling approaches:
  - Vertical Scaling (scale up)
  - Horizontal Scaling (scale out)

Helpful Links

- Amazon Route 53
- Amazon VPC
- AWS Direct Connect
- Amazon API Gateway
- Amazon CloudFront
- Elastic Load Balancing

Amazon VPC and Direct Connect

Write the definition for Amazon Virtual Private Cloud (VPC):

A logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define.

AWS Direct Connect - A cloud service solution that makes it easy to establish a dedicated network connection from your data center to AWS.

Amazon Route 53

Amazon Route 53 is a ___global___ service (meaning it does not require region selection).

Elastic Load Balancing

Distributes traffic across multiple ___targets___

Integrates with ___EC2___, ___ECS___, and ___Lambda___.

Types of load balancers:

1. ***Application Load Balancer (ALB)***
2. ***Network Load Balancer (NLB)***
3. ***Classic Load Balancer***

Types of Scaling:

___Vertical___ Scaling - You "scale up" your instance type to a larger instance type with additional resources

___Horizontal___ Scaling - You "scale out" and add additional instances to handle the demand of your application

Amazon CloudFront and API Gateway

CloudFront utilizes AWS ___edge___ ___locations___.

Supports both ___static___ and ___dynamic___ content.

Amazon API Gateway is a fully managed API management service.

AWS Global Accelerator

The AWS Global Accelerator is a networking service that can route your traffic through the AWS *global* *network* *infrastructure* to improve performance.

Scenarios

The following scenarios are presented in the course as a way to explore your understanding of the module. Include your answer here in this outline, as well as your notes on the solution to each scenario.

SCENARIO 1

- Jane's company maintains two corporate data centers
- They want their data centers to work alongside AWS for specific workloads
- She is wondering if there is a way to have a persistent connection to AWS
- What service from AWS would you recommend her company implement?

What's Your Answer: ***AWS Direct Connect***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 2

- Tim's company serves content through their site to users around the globe
- They are looking to optimize performance to users around the world
- They want to leverage a Content Delivery Network (CDN)
- Which service would enable optimized performance globally for their content?

What's Your Answer: ***Amazon CloudFront***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 3

- Ellen's company has an internal application that runs on an EC2 server
- Currently there is downtime as demand is greater than capacity for the server
- Ellen is trying to decide if she should use bigger servers or more servers
- Which scaling approach would you recommend and what services should they use?

What's Your Answer: ***Horizontal Scaling using Elastic Load Balancing***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

Module Wrap Up

Take a minute to write down any areas from this module that you don't fully understand or where you still have questions:

File Storage Services

Learning Outcomes

- Be able to explain the core features of Amazon S3
  - Different storage classes
  - Multiple availability zone (durability)
  - URL file access
  - Lifecycle policies
  - S3 Transfer Acceleration
- Be able to identify when S3 Glacier or S3 Glacier Deep Archive would be a good choice
- Know the differences between the two EC2 storage options:
  - Elastic Block Store (EBS)
  - Elastic File Store (EFS)
- Understand when the data transfer services should be leveraged
  - AWS Snowball
  - AWS Snowmobile

Helpful Links

- Amazon S3
- Amazon S3 Glacier
- Amazon Elastic Block Store (EBS)
- Amazon Elastic File System (EFS)
- AWS Snowball
- AWS Snowmobile

Amazon S3 Overview

S3 Non-Archival Storage Classes

Storage Class	Description
`S3 Standard`	the default storage class and is for frequently accessed data
`S3 Intelligent-Tiering`	will move your data to the correct storage class based on usage
`S3 Standard - IA`	for infrequently accessed data with the

	standard resilience
S3 One Zone Infrequent Access	is for infrequently access data that is only stored in one AZ

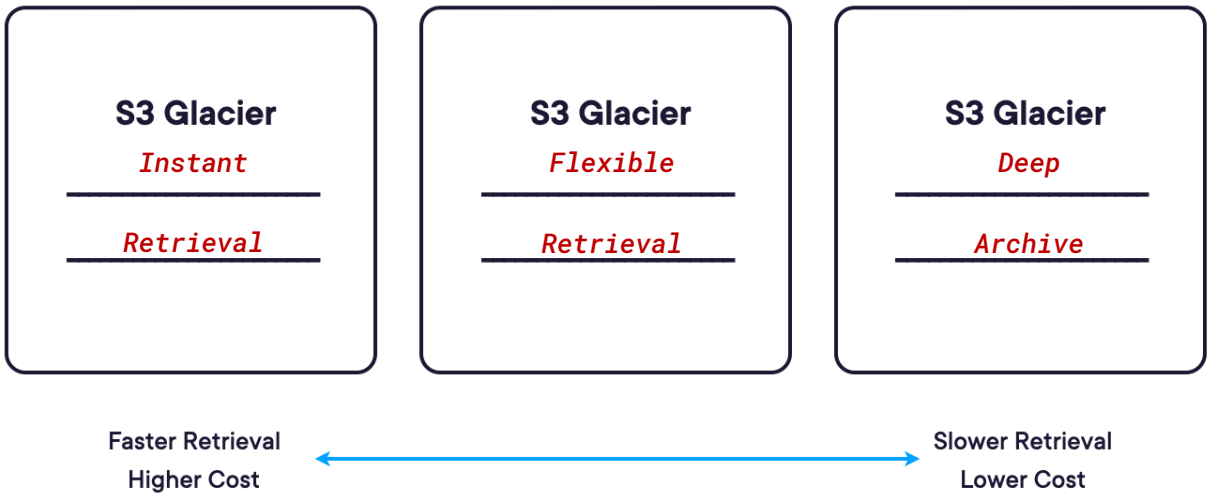
S3 Transfer Acceleration is a feature that can be enabled per bucket that allows for optimized uploading of data using the AWS Edge Locations as a part of Amazon CloudFront.

Hosting a Website on Amazon S3

1. Log into the AWS Console, and select the S3 service.
2. Click the **Create Bucket** button.
3. In the dialog, give the bucket a unique name.
4. For Object Ownership, select **ACLs enabled**.
5. Deselect the option to **Block all Public Access** under the Block Public Access settings for this bucket category. Once the warning appears you will need to click the checkbox in the acknowledgement. Click **Next**.
6. Under Default Encryption, select **Enable**.
7. Press the **Create Bucket** button.
8. Next, click on the newly created bucket in the list.
9. Next, click the **Upload** button. From the dialog, click the **Add Files** button.
10. Select the files from the exercise files. Click **Next**.
11. Press the **Upload** button.
12. Select the ps-logo.jpg file from the list. Attempt to navigate to the Object URL for this image.
13. Navigate back to the console and click on the image in the list. Click the permissions option to edit the permissions.
14. Press the **Edit** button and then click the option to enable **Read** access for **Everyone**.
15. Confirm the warning by selecting the checkbox.
16. Press the **Save changes** button.
17. Reload the image URL, and it should load without issue.
18. Back in the console, navigate to the bucket and then select the Properties tab.
19. From the properties tab, select scroll to Static Website Hosting and select the **Edit** button.
20. Next, select the option to **Enable** this feature**.** Enter index.html for the index document, Click **Save.**
21. Navigate to the URL for the static website hosting option. You will see that it is forbidden.
22. Next, navigate back to the console and select the index.html file. Update the permissions just as you did for the image.
23. Next, navigate back to the static website hosting URL. The site should now work.

Amazon S3 Glacier

Amazon S3 supports ___archive___ storage with S3 Glacier, which provides three different storage classes differentiated by cost and retrieval time.



AWS recommends ___Instant Retrieval___ for medical images, news media assets, or user-generated content archives.

AWS recommends ___Flexible Retrieval___ for backup, disaster recovery, and offsite data storage needs.

AWS recommends ___Deep Archive___ for long-term data storage to meet compliance requirements or cases where the lowest cost is critical.

Elastic Block Store

Amazon Elastic Block Store (EBS) is _____block_____ storage designed to be connected to a single _____EC2_____ instance that can scale to support _____petabytes_____ of data and supports multiple volume types based on need.

Please fill in the following table related to EBS volume types:

Volume Type Name	Description
General Purpose SSD	***Cost effective type designed for general workloads***
Provisioned IOPS SSD	high performance volume for low latency applications
Throughput Optimized HDD	is designed for frequently accessed data
Cold HDD	***Designed for less frequently accessed workloads***

Elastic File System

Amazon Elastic File System (EFS) is a fully managed ***NFS*** file system designed to support ***Linux*** workloads.

Amazon FSx for ***Windows File Server*** is a fully managed native Windows file system.

Data Transfer with AWS Snowball

Please fill in the following table related to data transfer services on AWS:

AWS Snowball	AWS Snowmobile
Designed for large-scale data transfer	*Designed for large-scale data transfer*
Supports petabyte scale transfer	Supports exabyte scale transfer
Physical device is delivered by AWS	*Ruggedized shipping container is delivered to your location*
You connect the Snowball to your network and upload your data	*AWS sets up a connection to your network*
Device is returned by local carrier	*You load your data on the Snowmobile*
AWS receives data and loads your data into S3	AWS will load data into S3 when the container is received at an AWS location

Scenarios

The following scenarios are presented in the course as a way to explore your understanding of the module. Include your answer here in this outline, as well as your notes on the solution to each scenario.

SCENARIO 1

- Elaine launched a site that offers daily tutorials for developers
- She uses S3 to store the assets needed per tutorial
- These assets are very popular within the week the tutorial is launched
- After this initial week, these assets are rarely accessed
- How could Elaine reduce her S3 costs while maintaining durability?

What's Your Answer: *S3 Lifecycle rules with S3-Standard IA storage class*

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 2

- Esteban works for a social networking company and they are moving to AWS
- They have 2 PB of user-generated content that they need to migrate
- Esteban is trying to determine if there is a faster than uploading over the internet
- Would there be another approach you would recommend for Esteban's company?

What's Your Answer: *AWS Snowball*

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 3

- Emily works for a company that produces a messaging app
- She is looking for a shared file system between 8 different Linux EC2 instances
- The file system would need to support roughly 1 PB of data
- What approach would you recommend for Emily?

What's Your Answer: ***Amazon Elastic File System***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

Module Wrap Up

Take a minute to write down any areas from this module that you don't fully understand or where you still have questions:

Database Services and Utilities

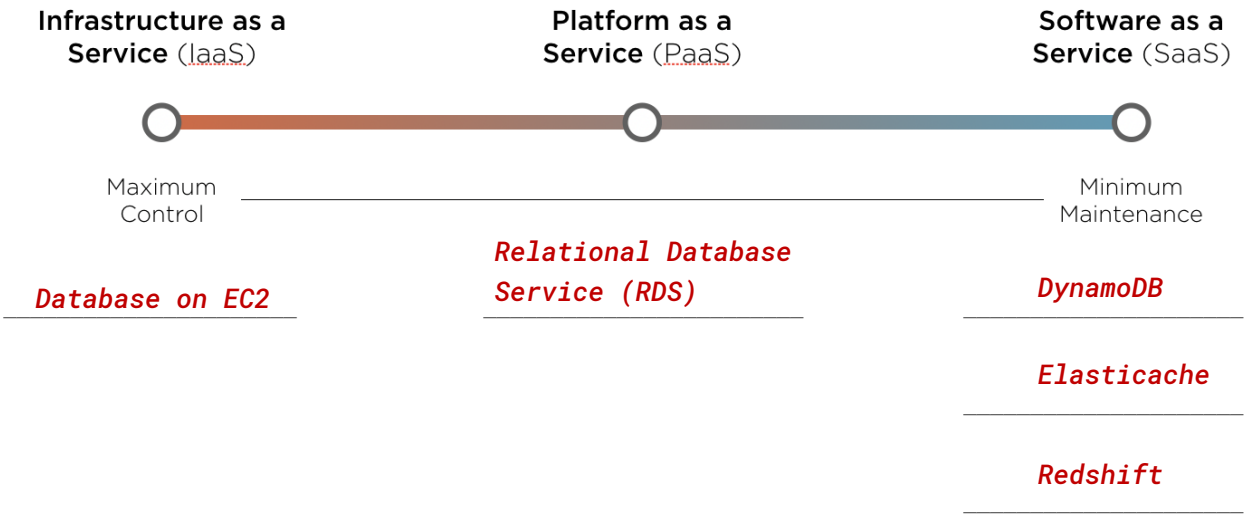
Learning Outcomes

- Be able to define the following database services:
  - Amazon Relational Database Service (RDS)
    - Understand what the Amazon Aurora database engine is within RDS
  - Amazon DynamoDB
  - Amazon Elasticache
- Be able to define the following data warehousing services and know when they would be used
  - Amazon Redshift and Redshift Spectrum
- Know when someone would leverage the AWS Database Migration Service

Helpful Links

- Amazon RDS
- Amazon Aurora
- Amazon DynamoDB
- Amazon Redshift and Redshift Spectrum
- Amazon Elasticache
- AWS Database Migration Service

Overview



Amazon Relational Database Service

Amazon RDS is a fully managed service for _____*relational*_____ databases.

Supported Amazon RDS Platforms:

1. *MySQL*
2. *PostgreSQL*
3. *MariaDB*
4. *Oracle*
5. *SQL Server*
6. *Amazon Aurora*

"_____*Amazon Aurora*_____ is a MySQL and PostgreSQL-compatible relational database built for the cloud, that combines the performance and availability of traditional enterprise databases with the simplicity and cost-effectiveness of open-source databases." - Amazon Web Services

Amazon DynamoDB Overview

Amazon DynamoDB is a fully managed _____*NoSQL*_____ database service.

"DynamoDB can handle more than _____*10 Trillion*_____ requests per day and can support peaks of more than _____*20 Million*_____ requests per second." - Amazon Web Services

Amazon Elasticache & Redshift

Amazon Elasticache is an in-memory data store that supports the ***Memcached*** and ***Redis*** engines.

Enter the service name based on the description:

Service	Description
Amazon Redshift	Data warehousing solution that supports petabytes of data
Amazon Redshift Spectrum	Service that enables querying exabytes of data stored in S3

Additional Database Services

Amazon DocumentDB

MongoDB compatible database

Amazon Neptune

Managed graph database service

Amazon MemoryDB for Redis

Redis compatible in-memory database

Amazon Timestream

Serverless time series database service

Scenarios

The following scenarios are presented in the course as a way to explore your understanding of the module. Include your answer here in this outline, as well as your notes on the solution to each scenario.

SCENARIO 1

- Jennifer is an IT executive in a financial services company
- They are transitioning their data warehouse to AWS for analysis
- The data warehouse would need to support up to 2 PB of data
- Which approach would you recommend for Jennifer?

What's Your Answer: ***Amazon Redshift***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 2

- Sam is a DevOps engineer at a tech company
- Sam needs to launch a MySQL database for a new web application
- They need to have direct access to the virtual server that MySQL is running on
- What approach would you recommend for Sam's company?

What's Your Answer: ***Database on EC2***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 3

- Frank is the CTO at a gaming company
- They are trying to determine how to store real-time user analytics
- They need low latency and the ability to scale to handle up to 1 million players
- Frank wants to minimize the amount of time it takes to maintain the database
- Which AWS approach would you recommend for Frank?

What's Your Answer: *DynamoDB*

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

Module Wrap Up

Take a minute to write down any areas from this module that you don't fully understand or where you still have questions:

App Integration Services

Learning Outcomes

- Be able to define the AWS messaging services but also know the differences in how they work
    - Amazon Simple Queue Service (SQS)
        - Know the two types of queues and how they are different
        - Be able to explain how SQS can enable fault tolerance
    - Amazon Simple Notification Service (SNS)
- Understand the purpose of AWS Step Functions and how they are defined

Helpful Links

- Amazon Simple Queue Service (SQS)
- Amazon Simple Notification Service (SNS)
- AWS Step Functions

AWS Messaging Services

Fill in the service in the table based on the description:

Service	Description
Amazon SNS	Fully managed pub/sub messaging service
Amazon SQS	Fully managed message queue service

Within Amazon SNS, messages are organized according to *topics* .

Within Amazon SQS, messages are organized into *queues*. There are two types of these. They are *standard* and *fifo*.

AWS Step Functions

AWS Step Functions enables orchestration of workflows through a fully managed service.

With AWS Step Functions, you are charged per ***state*** ***transition***.

Within AWS Step Functions, workflows are defined using ***Amazon States Language***.

Scenarios

The following scenarios are presented in the course as a way to explore your understanding of the module. Include your answer here in this outline, as well as your notes on the solution to each scenario.

SCENARIO 1

- Ruth started a non-profit that assigns volunteers to opportunities
- Recently their database server went down, and users were unable to signup
- While the situation is better, there is still some downtime expected in the future
- She wants to explore an AWS service that could prevent lost user signups
- What service would you recommend to Ruth?

What's Your Answer: *Amazon Simple Queue Service (SQS)*

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 2

- Jessi created a list of onboarding steps for new customers for their new app
- These steps detail integrations with their CRM, emails to the user, and analytics
- Jessi is worried about the time it will take to build all of this from scratch
- Is there an AWS service that can help with this approach?

What's Your Answer: *AWS Step Functions*

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 3

- Roger's company is an eCommerce company building a custom platform
- They are still adding new functionality
- He wants aspects of the platform to listen for events like orders and refunds
- They don't yet know all of the elements that would need to respond to events
- Is there a service that would allow current and future parts of the platform to listen for these events?

What's Your Answer: ***Amazon Simple Notification Service (SNS)***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

Module Wrap Up

Take a minute to write down any areas from this module that you don't fully understand or where you still have questions:

Management and Governance Services

Learning Outcomes

- Understand the benefit of AWS CloudTrail
  - Know where CloudTrail logs can be stored
- Know what services can help you monitor your AWS infrastructure
  - Amazon CloudWatch
  - AWS Config
- Be able to explain the purpose of AWS Systems Manager
- Be able to explain the value of launching infrastructure with AWS CloudFormation
- Be able to explain the purpose of AWS Control Tower

Helpful Links

- AWS CloudTrail
- Amazon CloudWatch
- AWS Config
- AWS Systems Manager
- AWS CloudFormation
- AWS Control Tower
- AWS OpsWorks

AWS CloudTrail

"AWS CloudTrail provides event history of your AWS account activity, including actions taken through the AWS Management Console, AWS SDKs, command line tools, and other AWS services." - Amazon Web Services

CloudTrail inserts an audit trail in an S3 Bucket or into CloudWatch Logs.

Amazon CloudWatch and AWS Config

Fill in the following table by entering the service name based on the description:

Service	Description
Amazon CloudWatch	Provides metrics, logs, and alarms for infrastructure
AWS Config	Continually evaluates infrastructure against a set of rules
AWS Systems Manager	Provides operational data and automation across infrastructure

Amazon CloudWatch allows for custom dashboards based on collected metrics.

"*AWS Config* continuously monitors and records your AWS resource configurations and allows you to automate the evaluation of recorded configurations against desired configurations." - Amazon Web Services

AWS Config provides specific *Conformance packs* with rules for specific compliance standards.

AWS Systems Manager

AWS Systems Manager provides multiple tools that make it easier to manage your AWS infrastructure.

AWS CloudFormation

_____*AWS CloudFormation*_____ is a managed service for provisioning infrastructure based on templates. The templates can be written in _____*YAML*_____ or _____*JSON*_____.

_____*Drift detection*_____ is a feature that enables you to find changes in your infrastructure after it was launched by CloudFormation.

AWS OpsWorks

"AWS OpsWorks is a _____*configuration*_____ _____*management*_____ service that provides managed instances of _____*Chef*_____ and _____*Puppet*_____."

AWS Organizations and Control Tower

AWS Control Tower - A service to create a _____*Multi-account*_____ environment on AWS that follows the recommended best practices in operational efficiency, security, and governance. It provides a way to create new AWS accounts based on _____*templates*_____.

Scenarios

The following scenarios are presented in the course as a way to explore your understanding of the module. Include your answer here in this outline, as well as your notes on the solution to each scenario.

SCENARIO 1

- Elliott is an operations engineer at a financial services company
- He recently discovered that someone had disabled a security setting on a server
- He is concerned that events like this might go unnoticed until a breach
- Which service would allow the organization to continually track configuration of infrastructure?

What's Your Answer: *AWS Config*

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 2

- James is the lead architect at a SaaS company
- They will be launching a new application that includes several components
- He is looking to minimize manual work required when creating infrastructure
- What service would enable James to automate much of this effort?

What's Your Answer: *AWS CloudFormation*

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

SCENARIO 3

- Candace is the CTO at a manufacturing company
- A cloud server needed to support their manufacturing process was deleted
- They want to make sure the follow up with the person who deleted this instance
- Which service could show the individual that deleted this specific server?

What's Your Answer: ***AWS CloudTrail***

Why did you pick this answer:

If you didn't get this one right, what insight did you gain from the explanation:

Module Wrap Up

Take a minute to write down any areas from this module that you don't fully understand or where you still have questions:

Next Steps

Complete all of the courses in this path to prepare for your AWS Certified Cloud Practitioner exam. In the last course of this path, we will include steps for registering, studying, and taking the exam.

Stay in Touch

If you have questions along the way, feel free to reach out to **David Tucker** on Twitter (@_davidtucker_) or through his website. Also, feel free to connect on LinkedIn.

For More Information

As a part of creating this course, the following resources from Amazon Web Services were referenced. If you want to learn more, feel free to go check out these resources directly:

- AWS Services
- Amazon EC2
- AWS Lambda
- Amazon Aurora
- Amazon DynamoDB
- AWS Config