SQS:

Amazon SQS is a 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.
Amazon SQS is a distributed queue system that enables web service applications to quickly and reliably queue messages that one component in the application generates to be consumed by another component. A queue is a temporary repository for messages that are awaiting processing.

Using Amazon SQS, you can decouple the components of an application so they run independently, easing message management between components. Any component of a distributed application can store messages in a fail-safe queue. Messages can contain up to 256 KB of text in any format. Any component can later retrieve the messages programmatically using the Amazon SQS API.

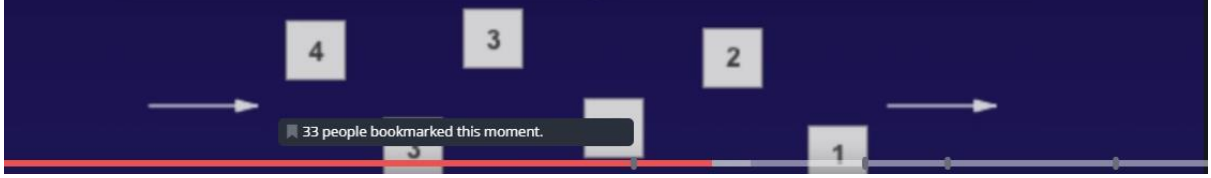
The queue acts as a buffer between the component producing and saving data, and the component receiving the data for processing. This means the queue resolves issues that arise if the producer is producing work faster than the consumer can process it, or if the producer or consumer are only intermittently connected to the network.

Autoscaling group can monitor the queue depth and start few more instances.

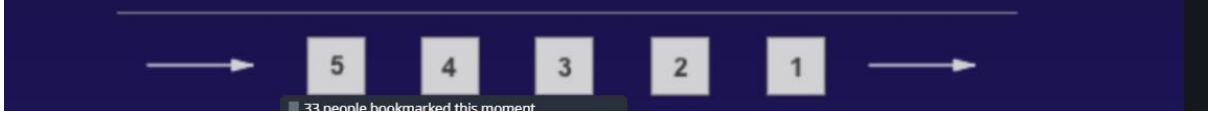
There are two types of Queue:

- Standard Queues (default)
- Fifo Queues

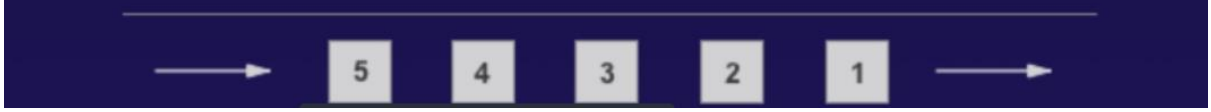
Amazon SQS offers standard as the default queue type. A standard queue lets you have a nearly-unlimited number of transactions per second. Standard queues guarantee that a message is delivered at least once. However, occasionally (because of the highly-distributed architecture that allows high throughput), more than one copy of a message might be delivered out of order. Standard queues provide best-effort ordering which ensures that messages are generally delivered in the same order as they are sent.



The FIFO queue complements the standard queue. The most important features of this queue type are FIFO (first-in-first-out) delivery and exactly-once processing: The order in which messages are sent and received is strictly preserved and a message is delivered once and remains available until a consumer processes and deletes it; duplicates are not introduced into the queue.



FIFO queues also support message groups that allow multiple ordered message groups within a single queue. FIFO queues are limited to 300 transactions per second (TPS), but have all the capabilities of standard queues.



It is tested in the exam(exactlyonce etc

SQS Exam Tips

- SQS is pull based, not pushed based.
- Messages are 256 KB in size.
- Messages can be kept in the queue from 1 minute to 14 days; the default retention period is 4 days.

SQS Exam Tips

- Visibility Time Out is the amount of time that the message is invisible in the SQS queue after a reader picks up that message. Provided the job is processed before the visibility time out expires, the message will then be deleted from the queue. If the job is not processed within that time, the message will become visible again and another reader will process it. This could result in the same message being delivered twice.
- Visibility timeout maximum is 12 hours.

SQS Exam Tips

- SQS guarantees that your messages will be processed at least once.
- Amazon SQS long polling is a way to retrieve messages from your Amazon SQS queues. While the regular short polling returns immediately (even if the message queue being polled is empty), long polling doesn't return a response until a message arrives in the message queue, or the long poll times out.

Long poll saves money as EC2 does not poll often.

SWF

Amazon Simple Workflow Service (Amazon SWF) is a web service that makes it easy to coordinate work across distributed application components. SWF enables applications for a range of use cases, including media processing, web application back-ends, business process workflows, and analytics pipelines, to be designed as a coordination of tasks.

Tasks represent invocations of various processing steps in an application which can be performed by executable code, web service calls, human actions, and scripts.

SWF vs SQS

- SQS has a retention period of up to 14 days; with SWF, workflow executions can last up to 1 year.
- Amazon SWF presents a task-oriented API, whereas Amazon SQS offers a message-oriented API.
- Amazon SWF ensures that a task is assigned only once and is never duplicated. With Amazon SQS, you need to handle duplicated messages and may also need to ensure that a message is processed only once.
- Amazon SWF keeps track of all the tasks and events in an application. With Amazon SQS, you need to implement your own application-level tracking, especially if your application uses multiple queues.

SWF Actors

- Workflow Starters — An application that can initiate (start) a workflow. Could be your e-commerce website following the placement of an order, or a mobile app searching for bus times.
- Deciders — Control the flow of activity tasks in a workflow execution. If something has finished (or failed) in a workflow, a Decider decides what to do next.
- Activity Workers — Carry out the activity tasks.

SNS:

Amazon Simple Notification Service (Amazon SNS) is a web service that makes it easy to set up, operate, and send notifications 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.

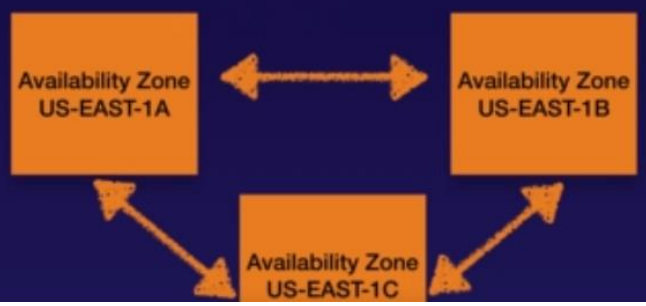
Push notifications to Apple, Google, Fire OS, and Windows devices, as well as Android devices in China with Baidu Cloud Push.

Besides pushing cloud notifications directly to mobile devices, Amazon SNS can also deliver notifications by SMS text message or email to Amazon Simple Queue Service (SQS) queues, or to any HTTP endpoint.

SNS allows you to group multiple recipients using topics. A topic is an "access point" for allowing recipients to dynamically subscribe for identical copies of the same notification.

One topic can support deliveries to multiple endpoint types — for example, you can group together iOS, Android and SMS recipients. When you publish once to a topic, SNS delivers appropriately formatted copies of your message to each subscriber.

To prevent messages from being lost, all messages published to Amazon SNS are stored redundantly across multiple availability zones.



SNS Benefits

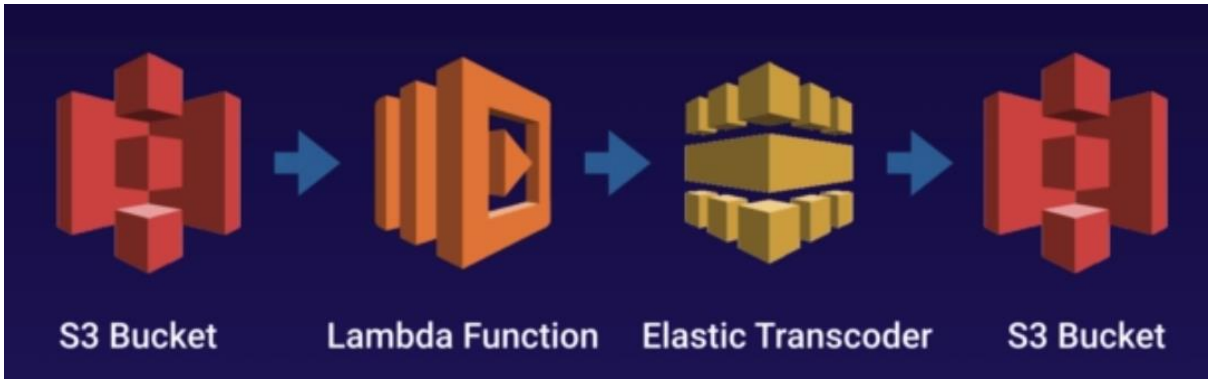
- Instantaneous, push-based delivery (no polling)
- Simple APIs and easy integration with applications
- Flexible message delivery over multiple transport protocols
- Inexpensive, pay-as-you-go model with no up-front costs
- Web-based AWS Management Console offers the simplicity of a point-and-click interface

SNS vs SQS?

- Both Messaging Services in AWS
- SNS - Push
- SQS - Polls (Pulls)

What Is Elastic Transcoder?

- Media Transcoder in the cloud.
- Convert media files from their original source format in to different formats that will play on smartphones, tablets, PCs, etc.
- Provides transcoding presets for popular output formats, which means that you don't need to guess about which settings work best on particular devices.
- Pay based on the minutes that you transcode and the resolution at which you transcode.

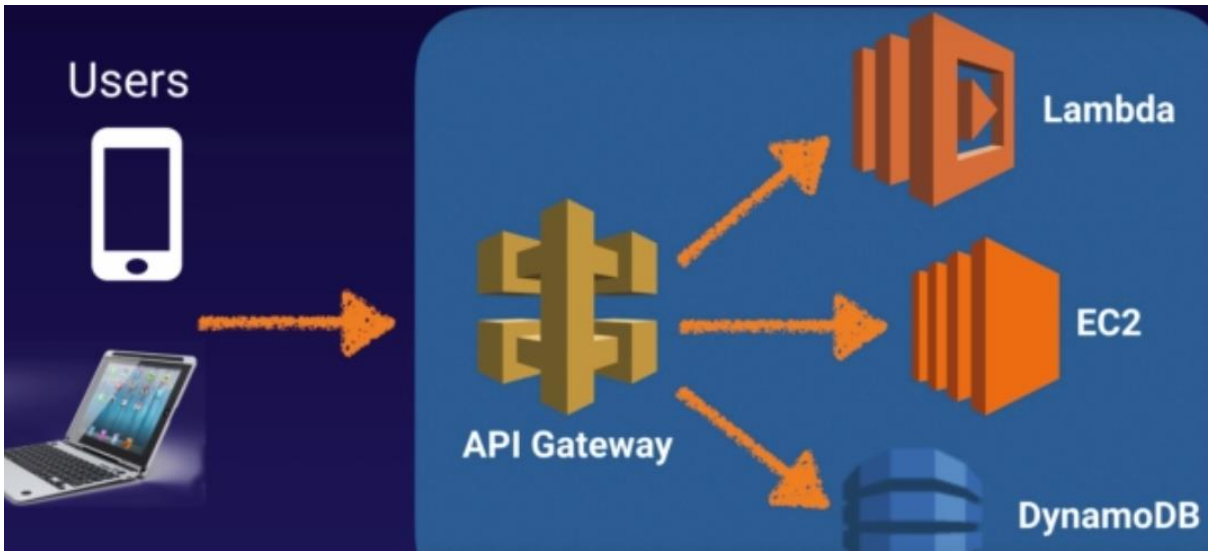


Just remember that Elastic Transcoder is a media transcoder in the cloud. It converts media files from their original source format in to different formats that will play on smartphones, tablets, PCs, etc.

API Gateway:

Amazon API Gateway is a fully managed service that makes it easy for developers to publish, maintain, monitor, and secure APIs at any scale.

With a few clicks in the AWS Management Console, you can create an API that acts as a "front door" for applications to access data, business logic, or functionality from your back-end services, such as applications running on Amazon Elastic Compute Cloud (Amazon EC2), code running on AWS Lambda, or any web application.



What Can API Gateway Do?

- Expose HTTPS endpoints to define a RESTful API
- Serverless-ly connect to services like Lambda & DynamoDB
- Send each API endpoint to a different target
- Run efficiently with low cost
- Scale effortlessly
- Track and control usage by API key
- Throttle requests to prevent attacks
- Connect to CloudWatch to log all requests for monitoring
- Maintain multiple versions of your API

How Do I Configure API Gateway?

- Define an API (container)
- Define Resources and nested Resources (URL paths)
- For each Resource:
  - Select supported HTTP methods (verbs)
  - Set security
  - Choose target (such as EC2, Lambda, DynamoDB, etc.)
  - Set request and response transformations

How Do I Deploy API Gateway?

- Deploy API to a stage:
  - Uses API Gateway domain, by default
  - Can use custom domain
  - Now supports AWS Certificate Manager: free SSL/TLS certs

You can enable API caching in Amazon API Gateway to cache your endpoint's response. With caching, you can reduce the number of calls made to your endpoint and also improve the latency of the requests to your API. When you enable caching for a stage, API Gateway caches responses from your endpoint for a specified time-to-live (TTL) period, in seconds. API Gateway then responds to the request by looking up the endpoint response from the cache instead of making a request to your endpoint.

In computing, the same-origin policy is an important concept in the web application security model. Under the policy, a web browser permits scripts contained in a first web page to access data in a second web page, but only if both web pages have the same origin.

This is done to prevent Cross-Site Scripting (XSS) attacks.

This is done to prevent Cross-Site Scripting (XSS) attacks.

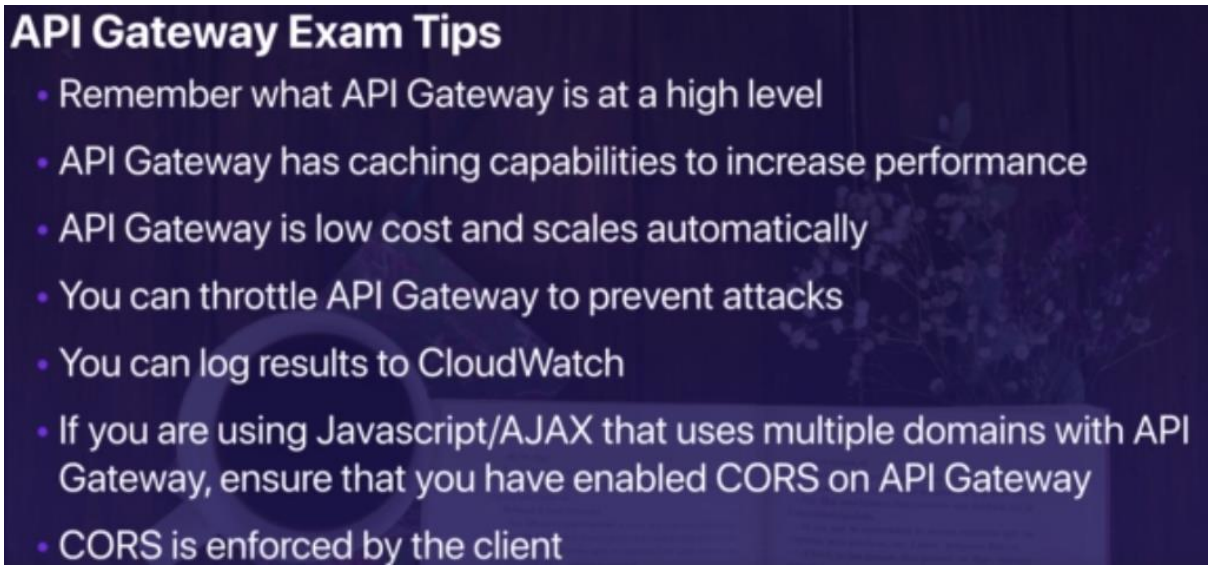
- Enforced by web browsers.
- Ignored by tools like PostMan and curl.

CORS is one way the server at the other end (not the client code in the browser) can relax the same-origin policy.

Cross-origin resource sharing (CORS) is a mechanism that allows restricted resources (e.g. fonts) on a web page to be requested from another domain outside the domain from which the first resource was served.

- Browser makes an HTTP OPTIONS call for a URL (OPTIONS is an HTTP method like GET, PUT, and POST)
- Server returns a response that says:
  "These other domains are approved to GET this URL."
- Error - "Origin policy cannot be read at the remote resource?" You need to enable CORS on API Gateway.

EXAM question above "Error" stuff



CORS:

The server is responsible for reporting the allowed origins. The web browser is responsible for enforcing that requests are only sent from allowed domains.

When CORS is not enabled a browser will rely on the same origin policy. The browser will only allow requests to the same host as the origin. The same origin policy is assumed when not origins are explicitly allowed.
An HTTP client other than a browser won't use either the same origin policy or CORS. Requests made from these other HTTP clients don't have an origin. Unless the Postman desktop app emulates a browser it will be able to make requests to any URL.

CORS and the same origin policy are needed because a browser does not implicitly trust the websites it visits. They don't protect the origin site, they protect the site receiving the cross origin request.

Without these policies a simple script that repeatedly loads a website could be distributed by ad networks or script injection and then any browser loading the script would contribute to a denial of service attack on the website. With CORS and the same origin policy a browser will limit the impact of this script.

CORS by example:

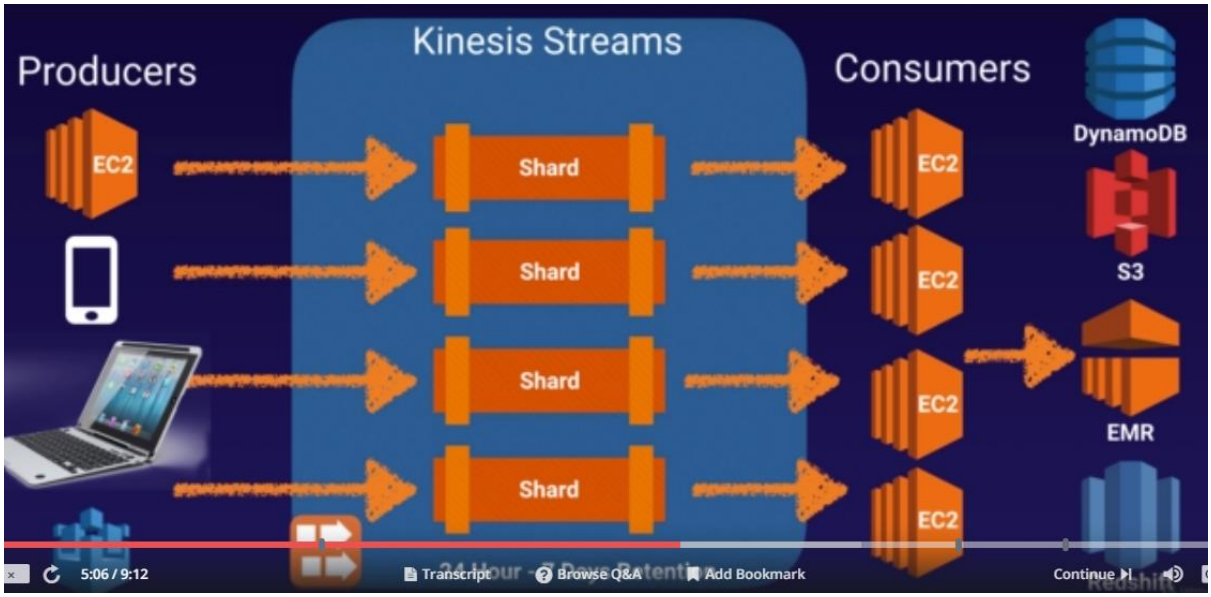
A web browser loads a page from `www.example.com`. The page includes a script that makes a request to `www.example.org`. The origin of the request is `www.example.com`. The browser either makes the request or sends an `OPTIONS` request first (the preflight request). When the server at `www.example.org` receives a request from an origin other than `www.example.org` it responds with a response header `Access-Control-Allow-Origin` which tells the browser the origins allowed to make requests. It may also respond with other headers like `Access-Control-Allow-Methods` and `Access-Control-Allow-Headers` that can restrict the types of allowed requests. When the browser is told what origins are allowed it will block future requests from disallowed origins.

KINESIS:

Streaming Data is data that is generated continuously by thousands of data sources, which typically send in the data records simultaneously, and in small sizes (order of Kilobytes.)

- Purchases from online stores (think amazon.com)
- Stock Prices
- Game data (as the gamer plays)
- Social network data
- Geospatial data (think uber.com)
- iOT sensor data

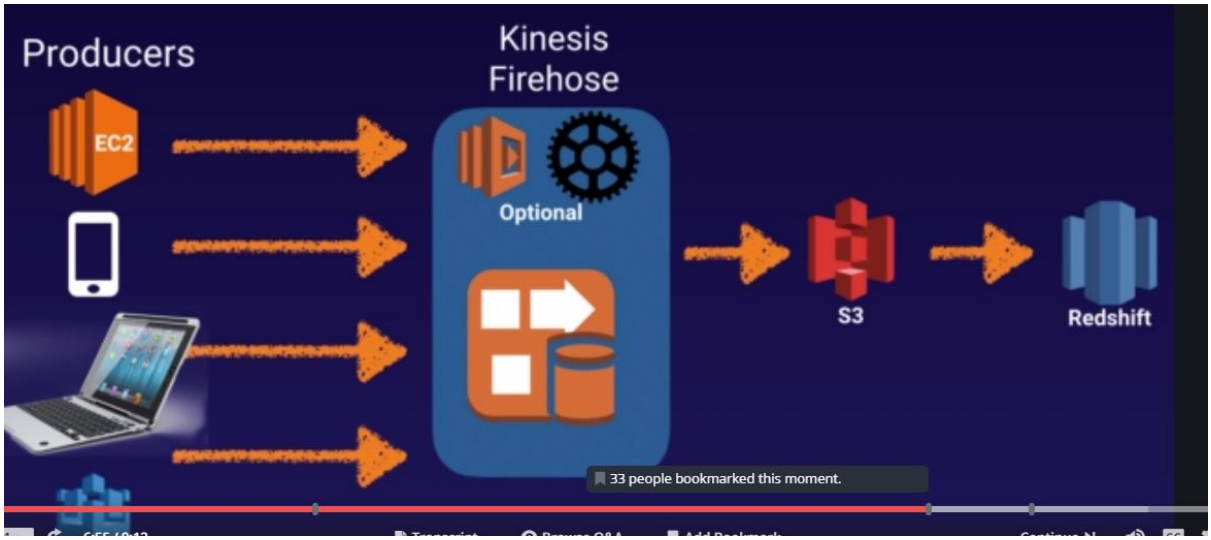
Amazon Kinesis is a platform on AWS to send your streaming data to. Kinesis makes it easy to load and analyze streaming data, and also providing the ability for you to build your own custom applications for you business needs.



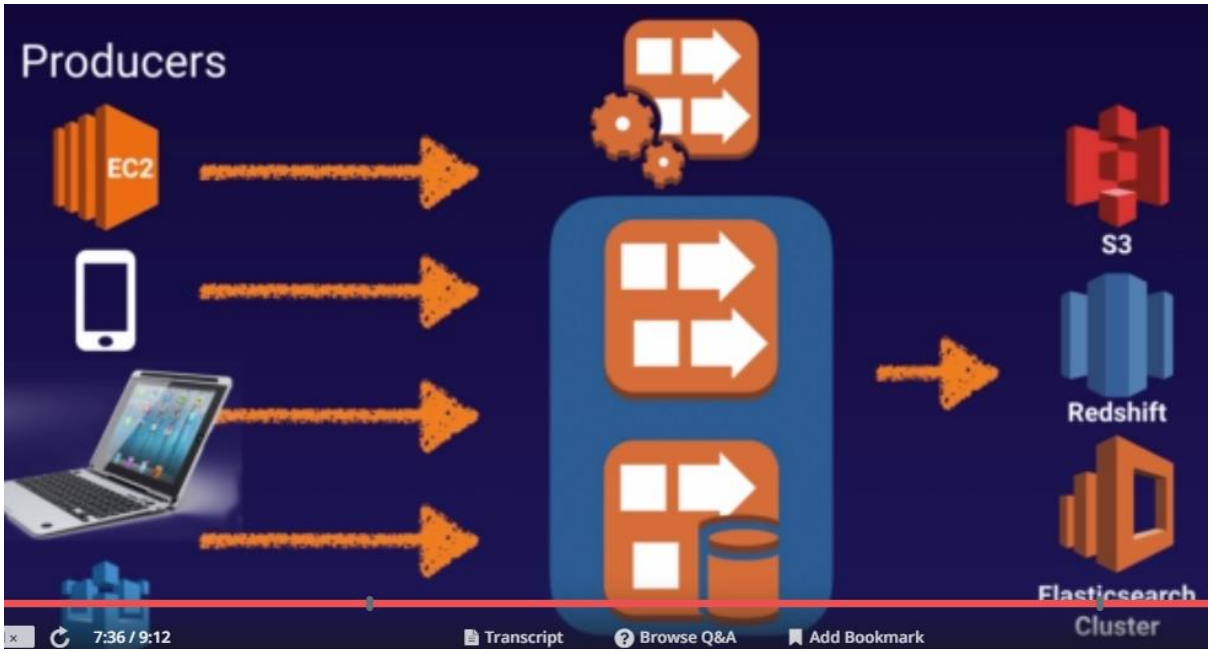
Store data for 24 hour to days

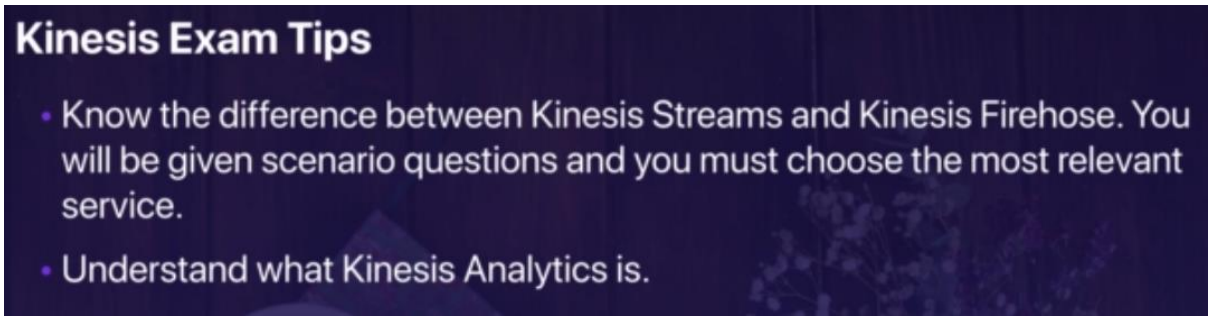
Kinesis Streams Consist Of Shards;

- 5 transactions per second for reads, up to a maximum total data read rate of 2 MB per second and up to 1,000 records per second for writes, up to a maximum total data write rate of 1 MB per second (including partition keys.)
- The data capacity of your stream is a function of the number of shards that you specify for the stream. The total capacity of the stream is the sum of the capacities of its shards.



Here, there is no persistence. The data should be processed as it comes. It cannot directly store the data into Redshift (due to lambda)



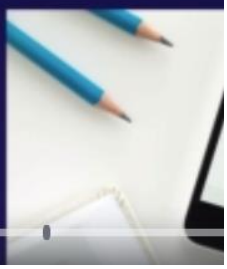


API Gateway comes a lot in the exam for 5 to 10 questions.

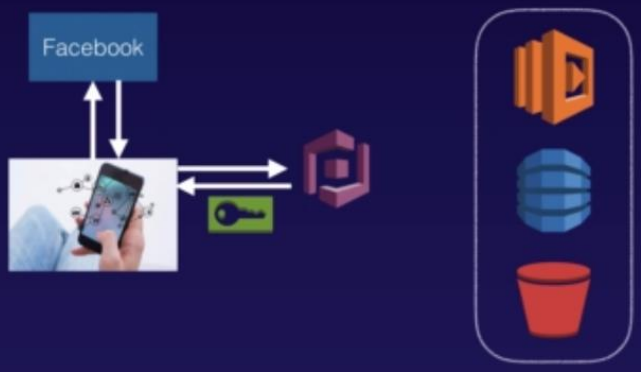
Web Identity Federation: Cognito

Amazon Cognito provides Web Identity Federation with the following features:

- Sign-up and sign-in to your apps
- Access for guest users
- Acts as an Identity Broker between your application and Web ID providers, so you don't need to write any additional code.
- Synchronizes user data for multiple devices
- Recommended for all mobile applications AWS services.



The recommended approach for Web Identity Federation using social media accounts like Facebook.

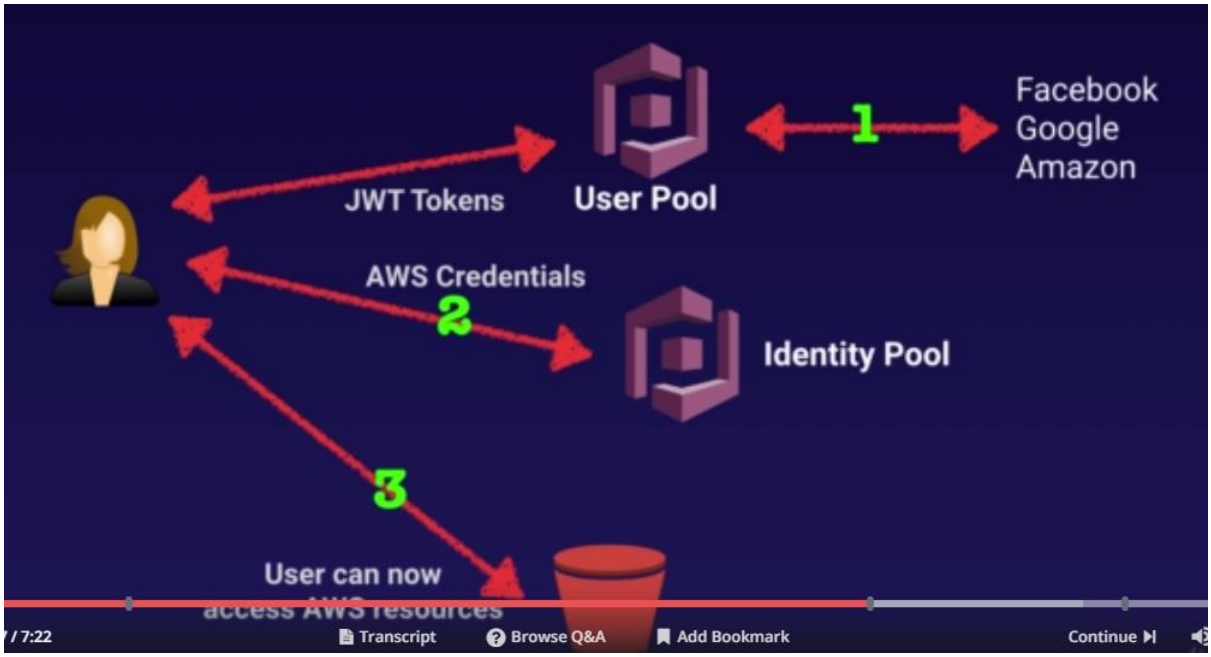


Cognito brokers between the app and Facebook or Google to provide temporary credentials which map to an IAM role allowing access to the required resources.

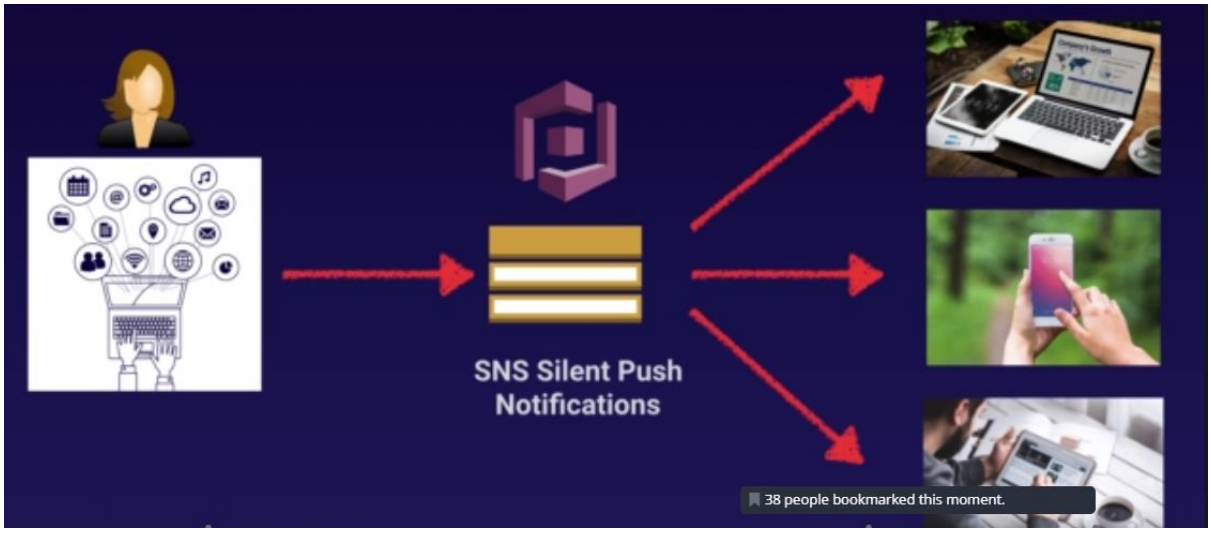
No need for the application to embed or store AWS credentials locally on the device and it gives users a seamless experience across all mobile devices.

User Pools are user directories used to manage sign-up and sign-in functionality for mobile and web applications.
Users can sign-in directly to the User Pool, or using Facebook, Amazon, or Google. Cognito acts as an Identity Broker between the identity provider and AWS. Successful authentication generates a JSON Web token (JWTs).

Identity Pools enable provide temporary AWS credentials to access AWS services like S3 or DynamoDB.



Cognito tracks the association between user identity and the various different devices they sign-in from. In order to provide a seamless user experience for your application, Cognito uses Push Synchronization to push updates and synchronize user data across multiple devices. Cognito uses SNS is to send a notification to all the devices associated with a given user identity whenever data stored in the cloud changes.



Cognito Exam Tips

- Federation allows users to authenticate with a Web Identity Provider (Google, Facebook, Amazon)
- The user authenticates first with the Web ID Provider and receives an authentication token, which is exchanged for temporary AWS credentials allowing them to assume an IAM role.
- Cognito is an Identity Broker which handles interaction between your applications and the Web ID provider (You don't need to write your own code to do this.)

Cognito Exam Tips

- User pool is user based. It handles things like user registration, authentication, and account recovery.
- Identity pools authorise access to your AWS resources.

SQS Exam Tips

- SQS is a way to de-couple your infrastructure
- SQS is pull based, not pushed based.
- Messages are 256 KB in size.
- Messages can be kept in the queue from 1 minute to 14 days; the default retention period is 4 days.
- Standard SQS and FIFO SQS
- Standard order is not guaranteed and messages can be delivered more than once.
- FIFO order is strictly maintained and messages are delivered only once.

1× 1:38 / 8:37 Transcript Browse Q&A Add Bookmark Continue

Difference between fifo and standard is critical for exam

SQS Exam Tips

- Visibility Time Out is the amount of time that the message is invisible in the SQS queue after a reader picks up that message. Provided the job is processed before the visibility time out expires, the message will then be deleted from the queue. If the job is not processed within that time, the message will become visible again and another reader will process it. This could result in the same message being delivered twice.
- Visibility timeout maximum is 12 hours.

SQS Exam Tips

- SQS guarantees that your messages will be processed at least once.
- Amazon SQS long polling is a way to retrieve messages from your Amazon SQS queues. While the regular short polling returns immediately (even if the message queue being polled is empty), long polling doesn't return a response until a message arrives in the message queue, or the long poll times out.

SWF vs SQS

- SQS has a retention period of up to 14 days; with SWF, workflow executions can last up to 1 year.
- Amazon SWF presents a task-oriented API, whereas Amazon SQS offers a message-oriented API.
- Amazon SWF ensures that a task is assigned only once and is never duplicated. With Amazon SQS, you need to handle duplicated messages and may also need to ensure that a message is processed only once.
- Amazon SWF keeps track of all the tasks and events in an application. With Amazon SQS, you need to implement your own application-level tracking, especially if your application uses multiple queues.

24 people bookmarked this moment.

SWF Actors - Exam Tips

- Workflow Starters — An application that can initiate (start) a workflow. Could be your e-commerce website following the placement of an order, or a mobile app searching for bus times.
- Deciders — Control the flow of activity tasks in a workflow execution. If something has finished (or failed) in a workflow, a Decider decides what to do next.
- Activity Workers — Carry out the activity tasks.

SNS Benefits

- Instantaneous, push-based delivery (no polling)
- Simple APIs and easy integration with applications
- Flexible message delivery over multiple transport protocols
- Inexpensive, pay-as-you-go model with no up-front costs
- Web-based AWS Management Console offers the simplicity of a point-and-click interface

SNS vs SQS?

- Both Messaging Services in AWS
- SNS - Push
- SQS - Polls (Pulls)

Just remember that Elastic Transcoder is a media transcoder in the cloud. It converts media files from their original source format in to different formats that will play on smartphones, tablets, PCs, etc.

API Gateway Exam Tips

- Remember what API Gateway is at a high level
- API Gateway has caching capabilities to increase performance
- API Gateway is low cost and scales automatically
- You can throttle API Gateway to prevent attacks
- You can log results to CloudWatch
- If you are using Javascript/AJAX that uses multiple domains with API Gateway, ensure that you have enabled CORS on API Gateway
- CORS is enforced by the client

34 people bookmarked this moment.

Kinesis Exam Tips

- Know the difference between Kinesis Streams and Kinesis Firehose. You will be given scenario questions and you must choose the most relevant service.
- Understand what Kinesis Analytics is.

Cognito Exam Tips

- Federation allows users to authenticate with a Web Identity Provider (Google, Facebook, Amazon)
- The user authenticates first with the Web ID Provider and receives an authentication token, which is exchanged for temporary AWS credentials allowing them to assume an IAM role.
- Cognito is an Identity Broker which handles interaction between your applications and the Web ID provider (You don't need to write your own code to do this.)

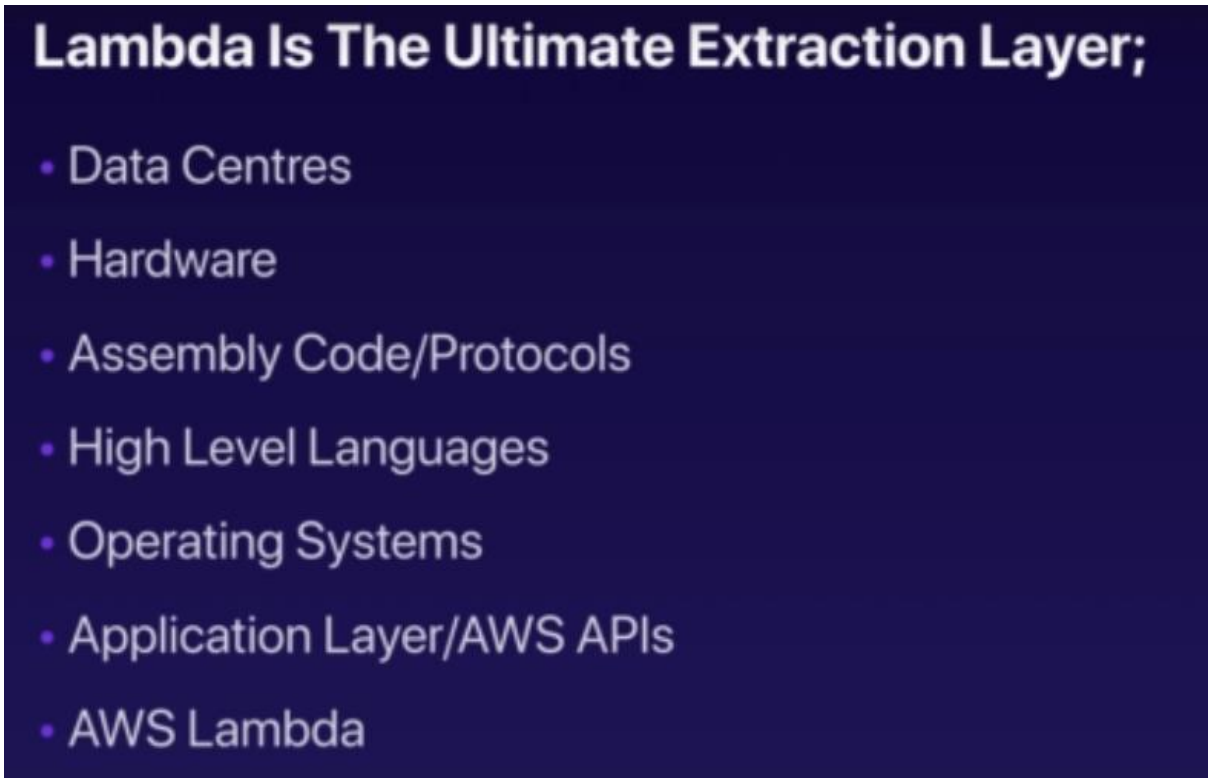
Cognito Exam Tips

- User pool is user based. It handles things like user registration, authentication, and account recovery.
- Identity pools authorise access to your AWS resources.

https://www.computernetworkingnotes.com/ccna-study-guide/basic-subnetting-in-computer-networks-explained.html

https://landing.google.com/sre/sre-book/toc/index.html

Serverless Architecture:

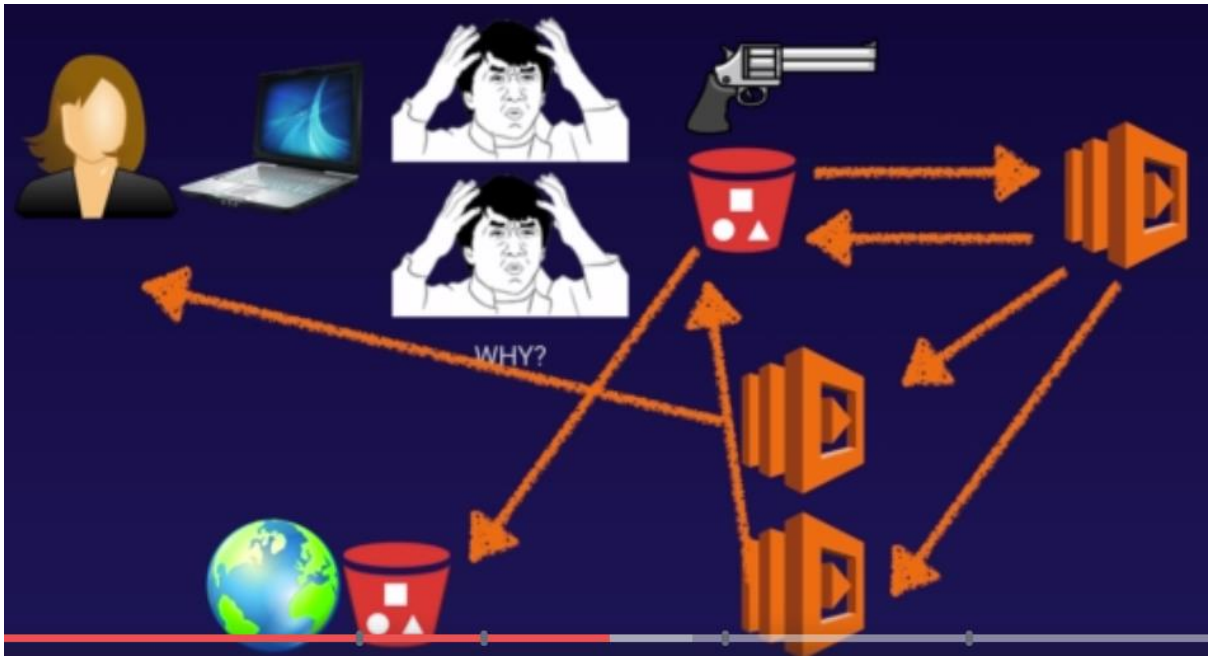


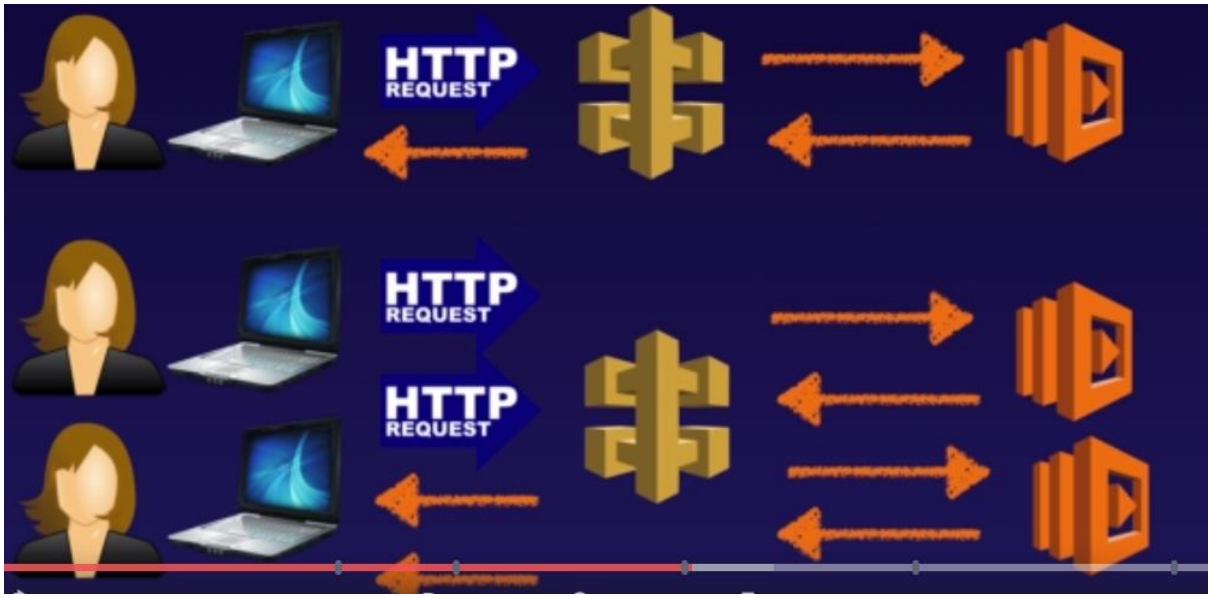
AWS Lambda is a compute service where you can upload your code and create a Lambda function. AWS Lambda takes care of provisioning and managing the servers that you use to run the code. You don't have to worry about operating systems, patching, scaling, etc.

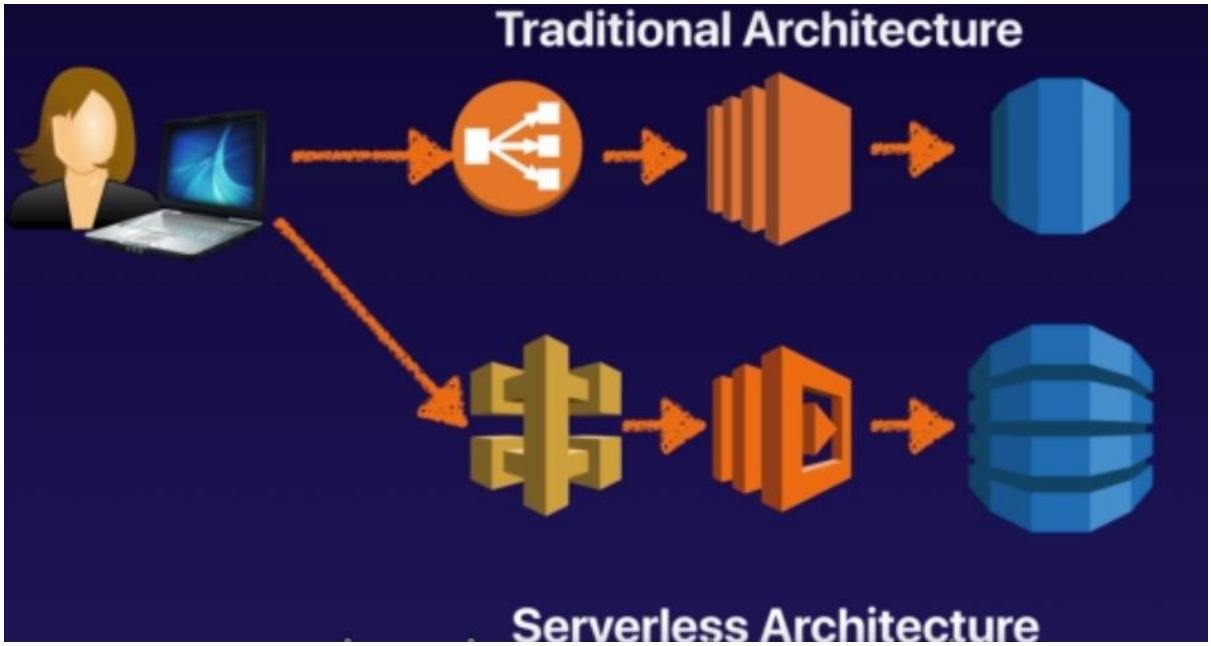
You can use Lambda in the following ways;

- As an event-driven compute service where AWS Lambda runs your code in response to events. These events could be changes to data in an Amazon S3 bucket or an Amazon DynamoDB table.
- As a compute service to run your code in response to HTTP requests using Amazon API Gateway or API calls made using AWS SDKs. This is what we use at A Cloud Guru.

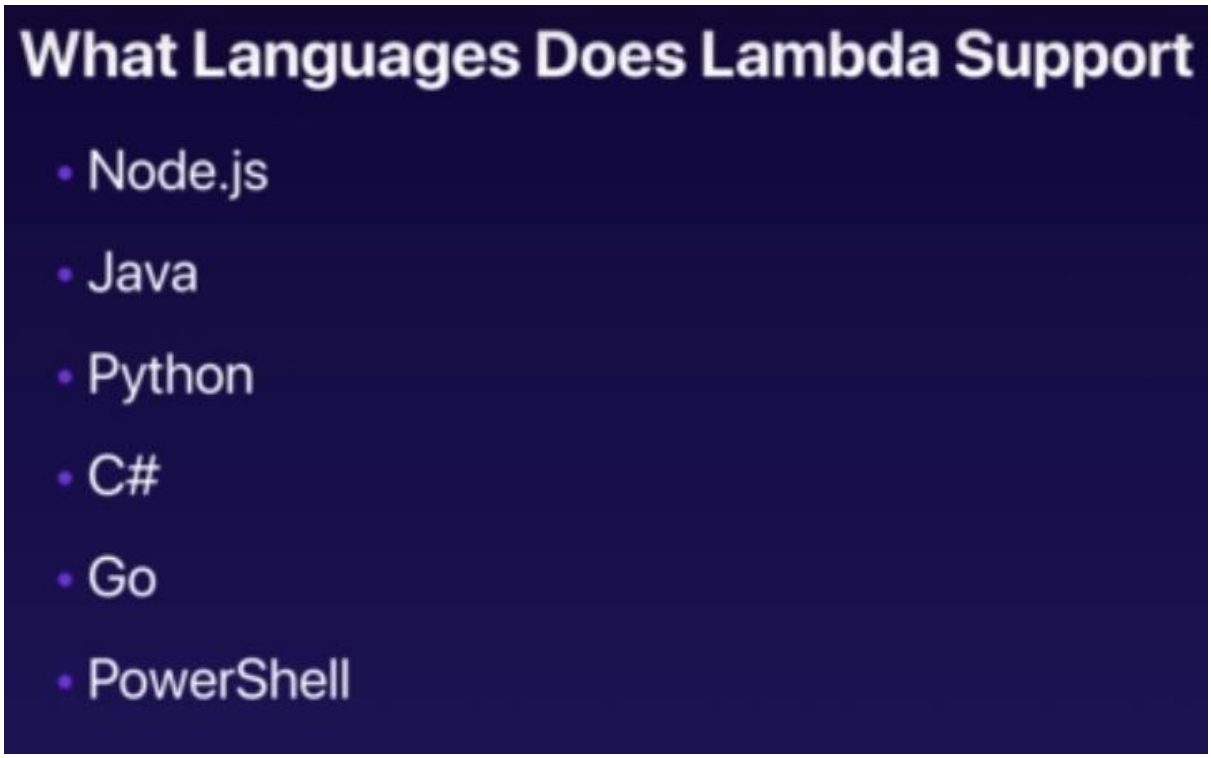
Creating a meme


WHY?


HTTP
REQUEST
HTTP
REQUEST
HTTP
REQUEST



Lambda automatically scales and we have to ensure the db is also serverless. Otherwise, it becomes bottleneck.



How Is Lambda Priced

1. **Number Of Requests**
   First 1 million requests are free. $0.20 per 1 million requests thereafter.

2. **Duration**
   Duration is calculated from the time your code begins executing until it returns or otherwise terminates, rounded up to the nearest 100ms. The price depends on the amount of memory you allocate to your function. You are charged $0.00001667 for every GB-second used.

Why Is Lambda Cool?

- NO SERVERS!
- Continuous Scaling
- Super super super cheap!

Lambda Exam Tips

- Lambda scales out (not up) automatically
- Lambda functions are independent, 1 event = 1 function
- Lambda is serverless
- Know what services are serverless!
- Lambda functions can trigger other lambda functions, 1 event can = x functions if functions trigger other functions

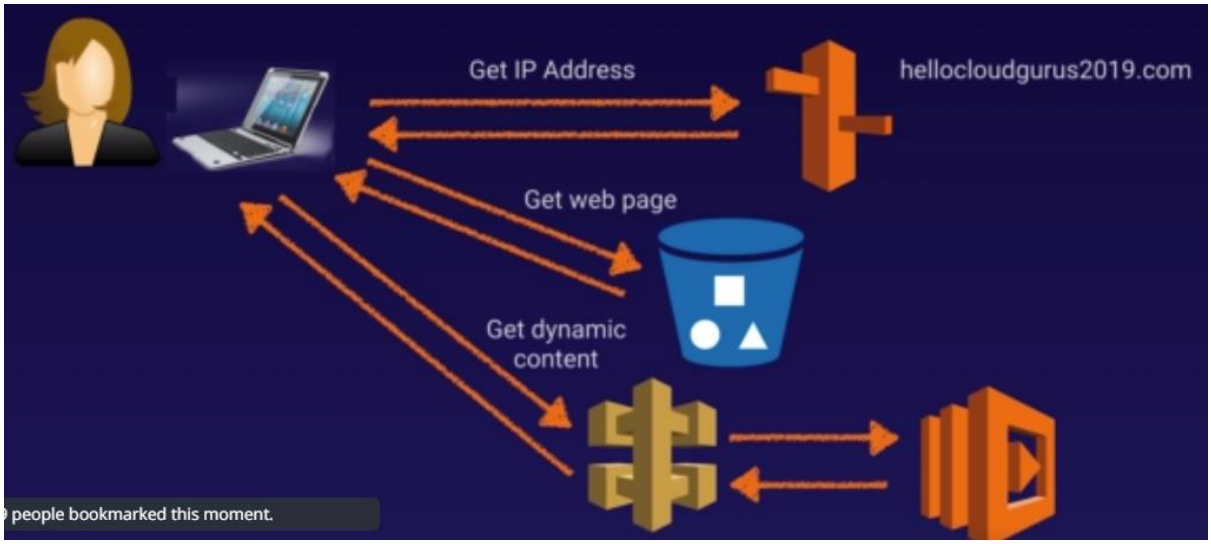
RDS is not serverless.

Arora is serverless, dynamodb, s3, api gateway, lambda are also serverless

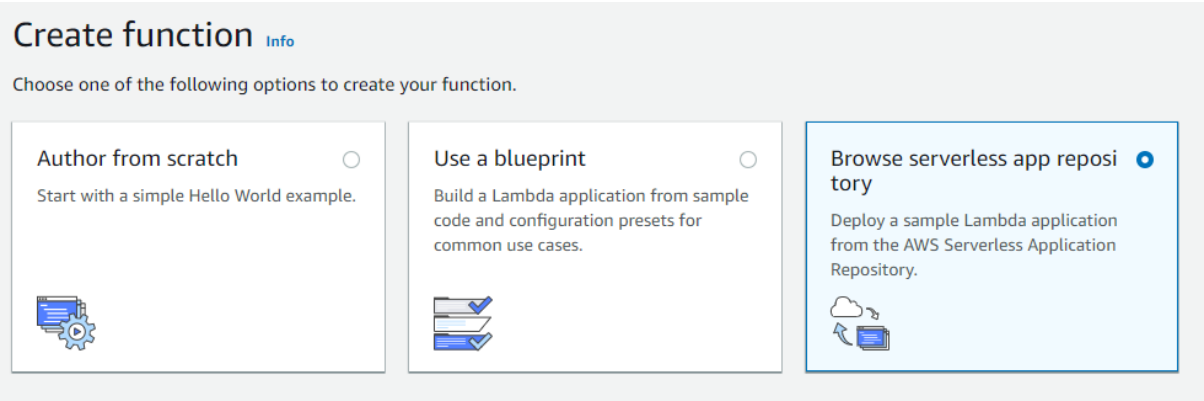
Lambda Exam Tips

- Architectures can get extremely complicated, AWS X-ray allows you to debug what is happening
- Lambda can do things globally, you can use it to back up S3 buckets to other S3 buckets etc
- Know your triggers

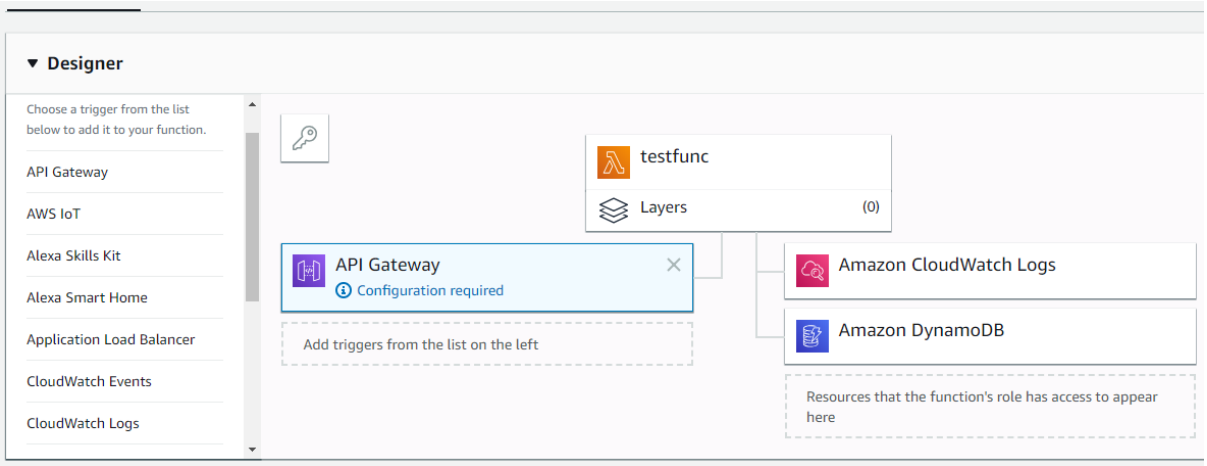
RDS cannot trigger lambda



Lambda is in compute section

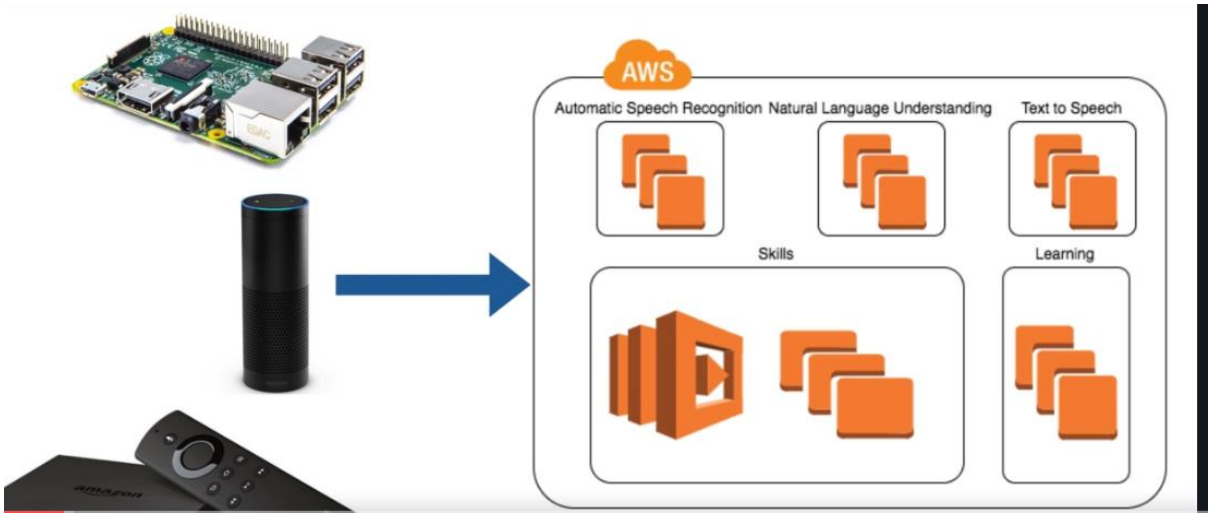


From scratch:

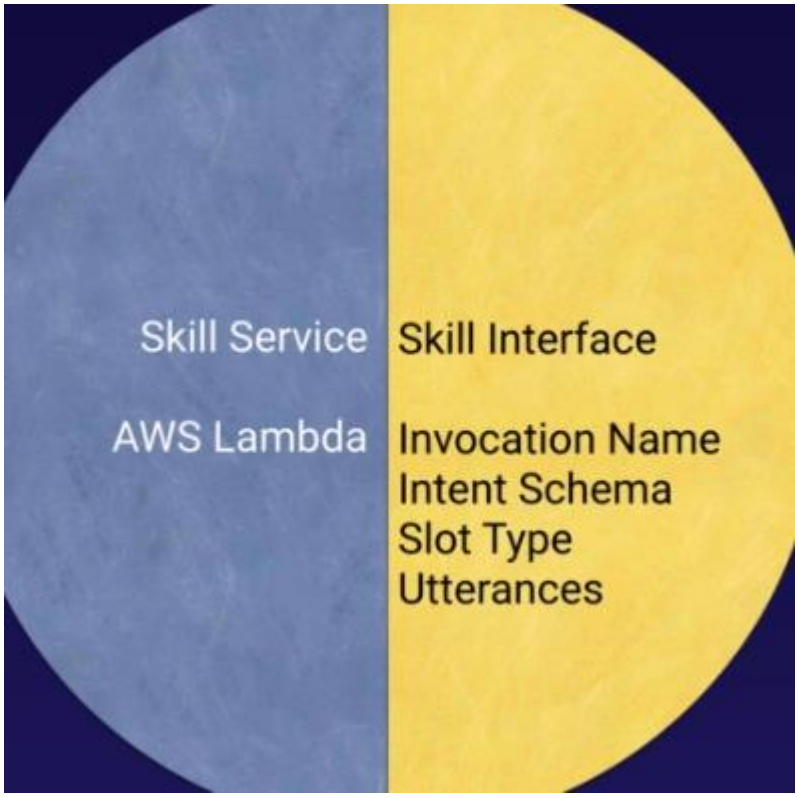


Knowing the triggers is an important aspect for exam.

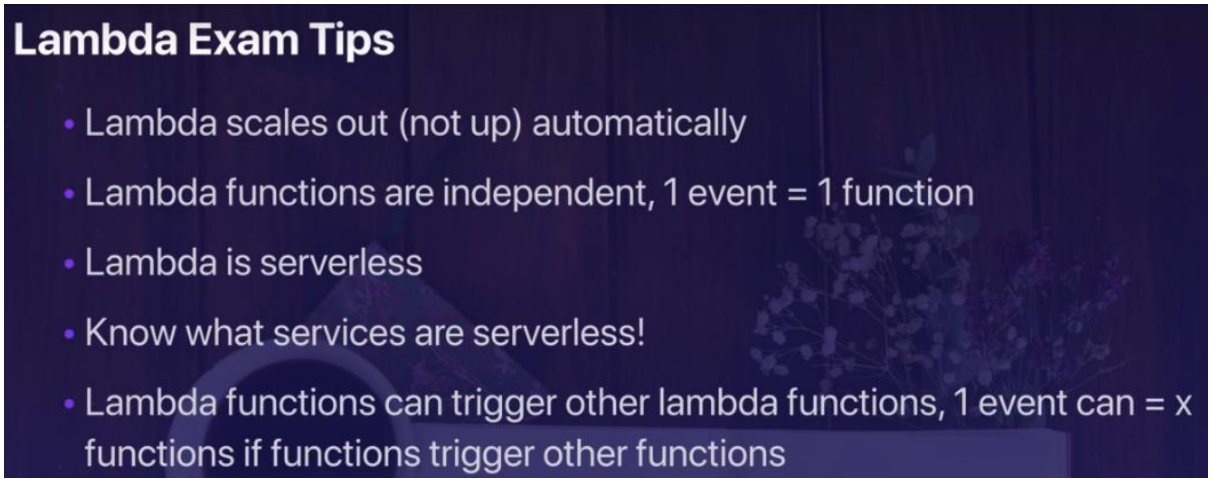
Alexa is similar to Siri

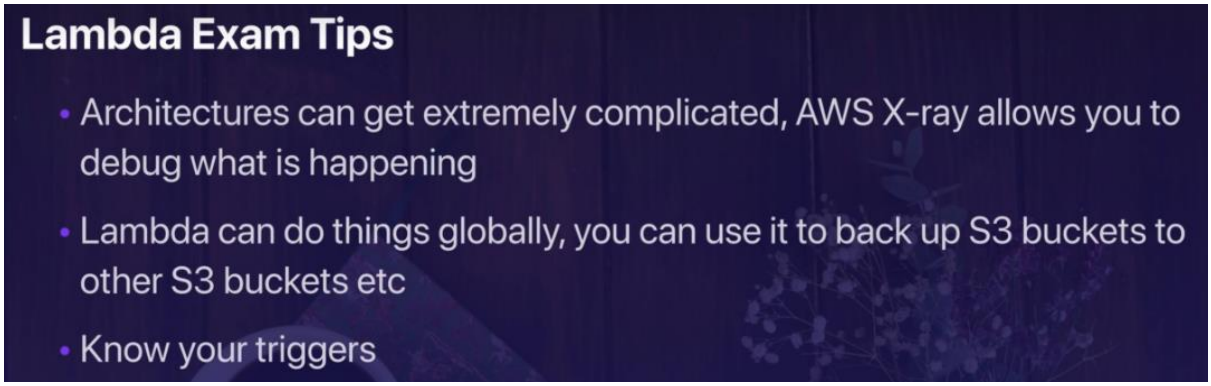


Setup the account at Developer.amazon.com



Amazon polly can be found under machine learning. It helps in text to speech etc





ARN: Amazon Resource Name

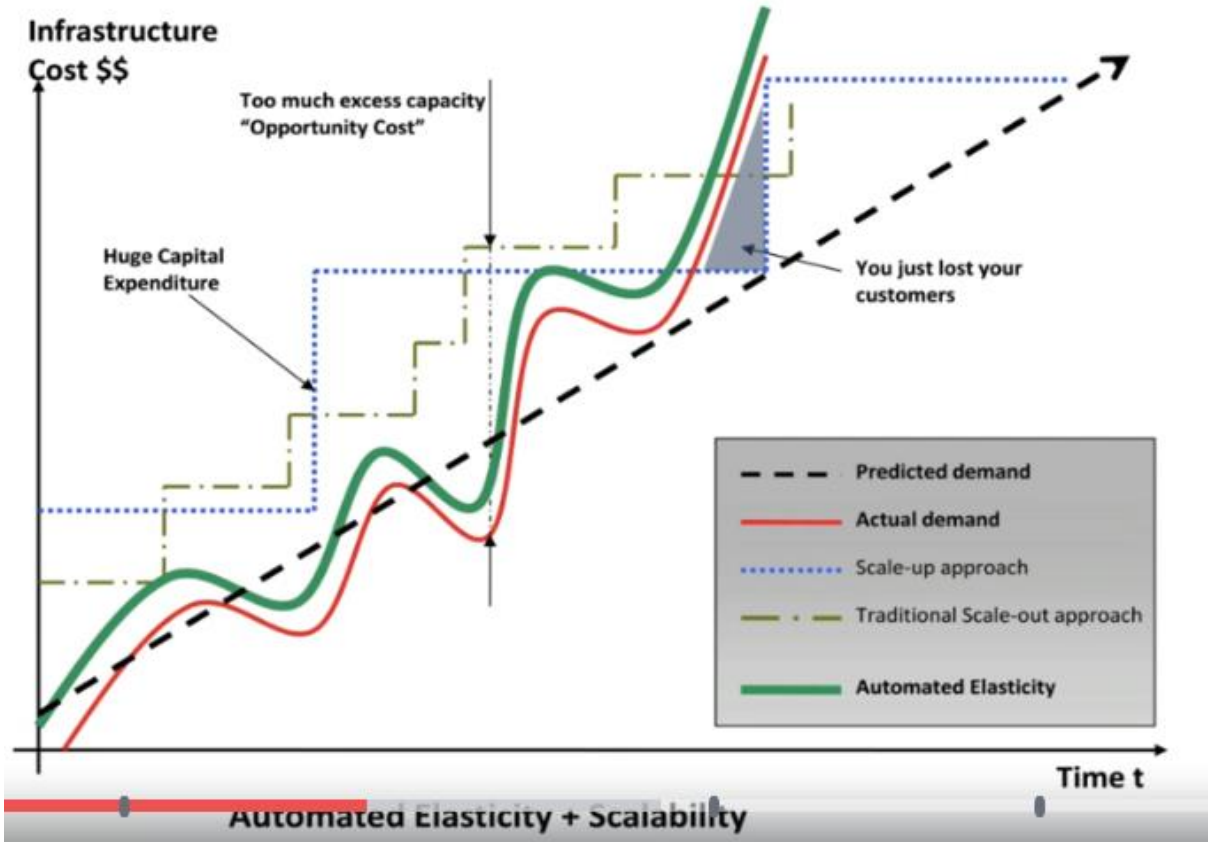
https://aws.amazon.com/architecture/well-architected/

Read AWS_Cloud_Best_Practices.pdf

Business benefits of cloud:

- Almost zero upfront infrastructure investment
- Just-in-time Infrastructure
- More efficient resource utilization
- Usage-based costing
- Reduced time to market

- Automation – “Scriptable infrastructure”
- Auto-scaling
- Proactive Scaling
- More Efficient Development lifecycle
- Improved Testability
- Disaster Recovery and Business Continuity
- “Overflow” the traffic to the cloud



Rule of thumb: Be a pessimist when designing architectures in the cloud; assume things will fail. In other words, always design, implement and deploy for automated recovery from failure.

In particular, assume that your hardware will fail. Assume that outages will occur. Assume that some disaster will strike your application. Assume that you will be slammed with more than the expected number of requests per second some day. Assume that with time your application software will fail too. By being a pessimist, you end up thinking about recovery strategies during design time, which helps in designing an overall system better.

The key is to build components that do not have tight dependencies on each other, so that if one component were to die (fail), sleep (not respond) or remain busy (slow to respond) for some reason, the other components in the system are built so as to continue to work as if no failure is happening.

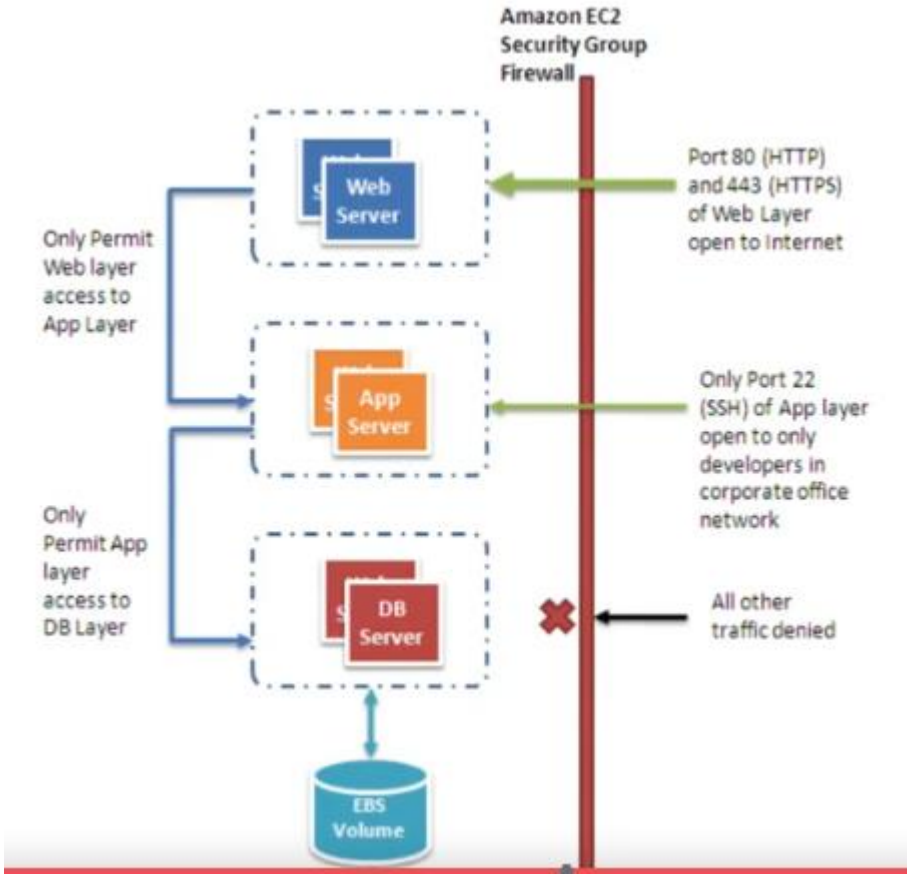
In essence, loose coupling isolates the various layers and components of your application so that each component interacts asynchronously with the others and treats them as a "black box".

For example, in the case of web application architecture, you can isolate the app server from the web server and from the database. The app server does not know about your web server and vice versa, this gives decoupling between these layers and there are no dependencies code-wise or functional perspectives.

In the case of batch- processing architecture, you can create asynchronous components that are independent of each other.

The cloud brings a new concept of elasticity in your applications. Elasticity can be implemented in three ways:

1. Proactive Cyclic Scaling: Periodic scaling that occurs at fixed interval (daily, weekly, monthly, quarterly)
2. Proactive Event-based Scaling: Scaling just when you are expecting a big surge of traffic requests due to a scheduled business event (new product launch, marketing campaigns)
3. Auto-scaling based on demand. By using a monitoring service, your system can send triggers to take appropriate actions so that it scales up or down based on metrics (utilization of the servers or network i/o, for instance)



Well architected framework:

There are 5 pillars

This has been developed by the Solutions Architecture team based on their experience with helping AWS customers. The well architected framework is a set of questions that you can use to evaluate how well your architecture is aligned to AWS best practices.

- Security
- Reliability
- Performance Efficiency
- Cost Optimisation
- Operational Excellence

- Design Principles
- Definition
- Best Practices
- Key AWS Services
- Resources

- Stop guessing your capacity needs
- Test systems at production scale
- Automate to make architectural experimentation easier
- Allow for evolutionary architectures
- Data-Driven architectures
- Improve through game days

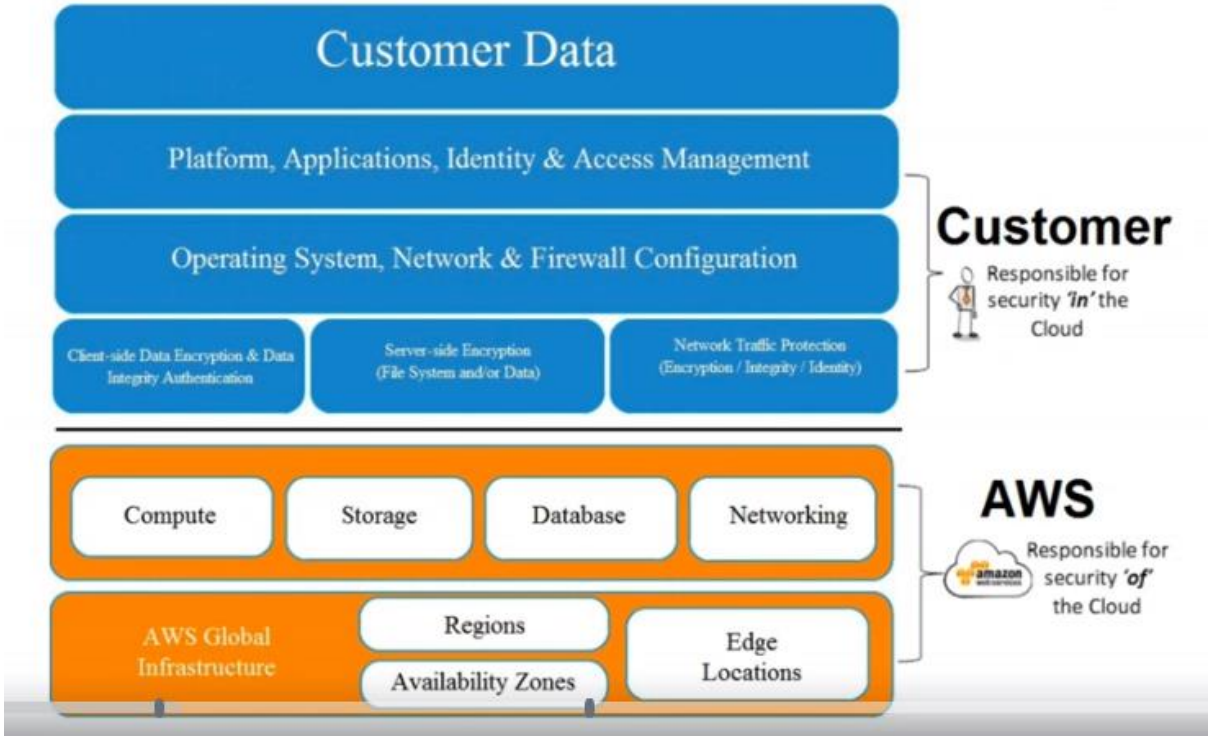
game day: event day

First pillar:

Security

- Apply security at all layers
- Enable traceability
- Automate responses to security events
- Focus on securing your system
- Automate security best practices

Responsibility Model:



Security in the cloud consists of 4 areas;

- Data protection
- Privilege management
- Infrastructure protection
- Detective controls

Before you begin to architect security practices across your environment, basic data classification should be in place. You should organise and classify your data in to segments such as publicly available, available to only members of your organisation, available to only certain members of your oganisation, available only to the board etc. You should also implement a least privilege access system so that people are only able to access what they need. However most importantly, you should encrypt everything where possible, whether it be at rest or in transit.

In AWS the following practices help to protect your data;

- AWS customers maintain full control over their data.
- AWS makes it easier for you to encrypt your data and manage keys, including regular key rotation, which can be easily automated natively by AWS or maintained by a customer.
- Detailed logging is available that contains important content, such as file access and changes.
- AWS has designed storage systems for exceptional resiliency. As an example, Amazon Simple Storage Service (S3) is designed for 11 nines of durability. (For example, if you store 10,000 objects with Amazon S3, you can on average expect to incur a loss of a single object once every 10,000,000 years.)

07:53 / 18:14 Transcript Browse Q&A Add Bookmark Continue

- Versioning, which can be part of a larger data lifecycle-management process, can protect against accidental overwrites, deletes, and similar harm.
- AWS never initiates the movement of data between regions. Content placed in a region will remain in that region unless the customer explicitly enable a feature or leverages a service that provides that functionality.

Best Practices - Data Protection Questions

A CLOUD GURU

- How are you encrypting and protecting your data at rest?
- How are you encrypting and protecting your data in transit? (SSL)

Best Practices - Privilege Management

A CLOUD GU

Privilege Management ensures that only authorized and authenticated users are able to access your resources, and only in a manner that is intended. It can include

- Access Control Lists (ACLs)
- Role Based Access Controls
- Password Management (such as password rotation policies)

- How are you protecting access to and use of the AWS root account credentials?
- How are you defining roles and responsibilities of system users to control human access to the AWS Management Console and APIs?
- How are you limiting automated access (such as from applications, scripts, or third-party tools or services) to AWS resources?
- How are you managing keys and credentials?

Enable multi factor authentication for the first one

Outside of Cloud, this is how you protect your data centre. RFID controls, security, lockable cabinets, CCTV etc. Within AWS they handle this, so really Infrastructure protection exists at a VPC level.

Best Practices - Infrastructure Protection Questions

A CLOUD GURU

- How are you enforcing network and host-level boundary protection?
- How are you enforcing AWS service level protection?
- How are you protecting the integrity of the operating systems on your Amazon EC2 instances?

Antivirus for 3d one

Best Practices - Detective Controls

A CLOUD GU

You can use detective controls to detect or identify a security breach.

AWS Services to achieve this include;

- AWS CloudTrail
- Amazon CloudWatch
- AWS Config
- Amazon Simple Storage Service (S3)
- Amazon Glacier

Best Practices - Detective Control Questions

- How are you capturing and analyzing AWS logs?

Cloudtrail is per region. It should be enabled separately for each region.

Key AWS Services

A CLOUD GURU

- Data protection
  - You can encrypt your data both in transit and at rest using; ELB, EBS, S3 & RDS
- Privilege management
  - IAM, MFA
- Infrastructure protection
  - VPC
- Detective controls
  - AWS Cloud Trail, AWS Config, Amazon Cloud Watch.

Security in the cloud consists of 4 areas;

- Data protection
- Privilege management
- Infrastructure protection
- Detective controls

Exam Tips - Security Pillar - Questions

A CLOUD GURU

- Data protection
  - How are you encrypting and protecting your data at rest?
  - How are you encrypting and protecting your data in transit? (SSL)
- Privilege management
  - How are you protecting access to and use of the AWS root account credentials?
  - How are you defining roles and responsibilities of system users to control human access to the AWS Management Console and APIs?
  - How are you limiting automated access (such as from applications, scripts, or third-party tools or services) to AWS resources?

How are you managing keys and credentials

Exam Tips - Security Pillar - Questions

A CLOUD GURU

- Infrastructure protection
  - How are you enforcing network and host-level boundary protection?
  - How are you enforcing AWS service level protection?
  - How are you protecting the integrity of the operating systems on your Amazon EC2 instances?
- Detective controls
- How are you capturing and analyzing AWS logs?

Read white paper

Reliability

The reliability pillar covers the ability of a system to recover from service or infrastructure outages/disruptions as well as the ability to dynamically acquire computing resources to meet demand.

Design Principles

A CLOUD

- Test recovery procedures
- Automatically recover from failure
- Scale horizontally to increase aggregate system availability
- Stop guessing capacity

Reliability in the cloud consists of 3 areas;

- Foundations
- Change management
- Failure management

vo - Reliability
Lecture 194

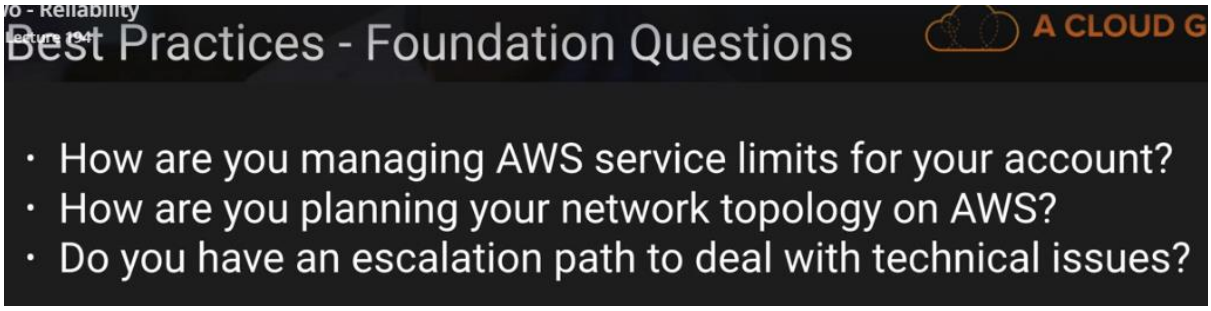
Best Practices - Foundations

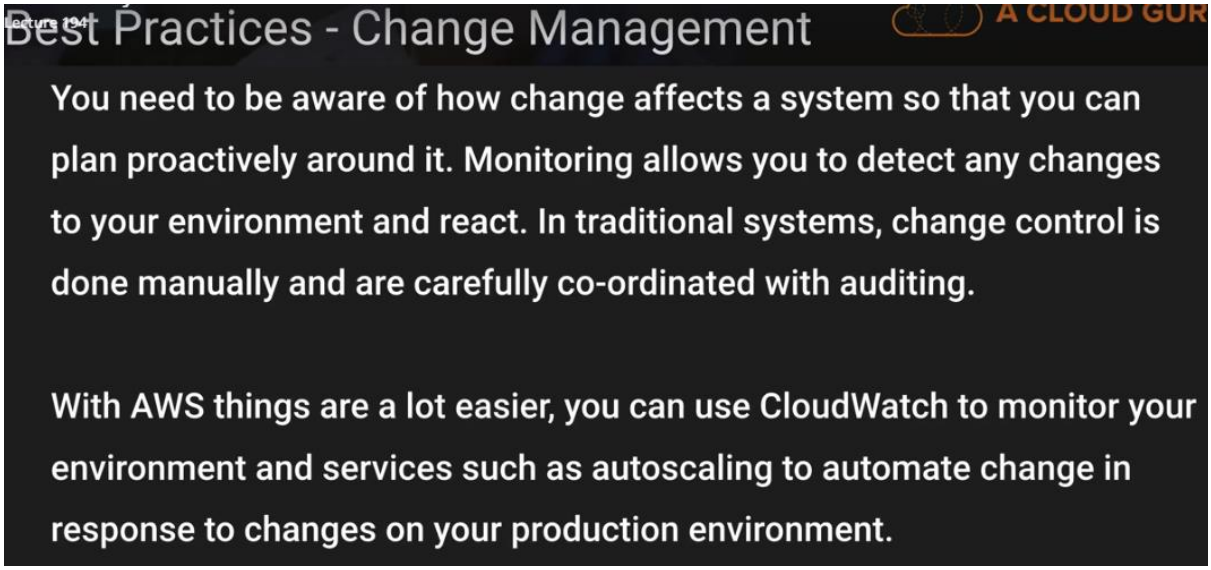
A CLOUD GURU

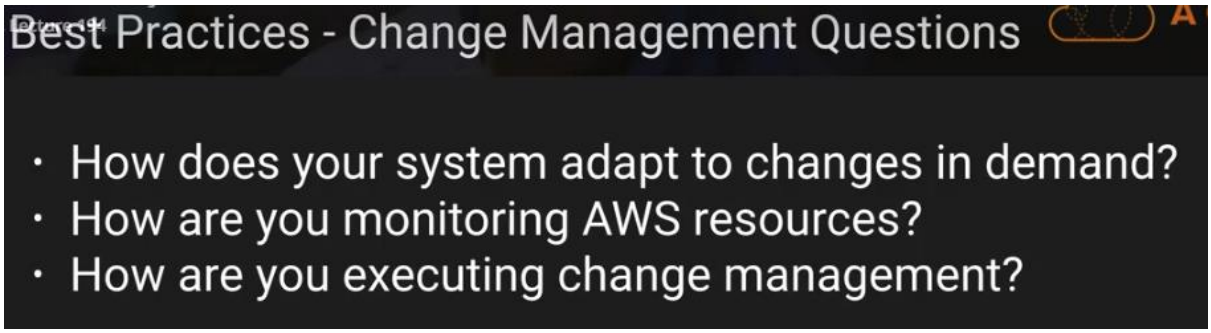
Before building a house, you always make sure that the foundations are in place before you lay the first brick. Similarly before architecting any system, you need to make sure you have the prerequisite foundations. In traditional IT one of the first things you should consider is the size of the comms link between your HQ and your datacenter. If you misprovision this link, it can take 3 - 6 months to upgrade which can cause a huge disruption to your traditional IT estate.

With AWS, they handle most of the foundations for you. The cloud is designed to be essentially limitless meaning that AWS handle the networking and compute requirements themselves. However they do set service limits to stop customers from accidentally over-provisioning resources.

AWS Service limits: To ensure accidentally we don't over provision resources.







Best Practices - Failure Management

A CLOUD GU

With cloud, you should always architect your systems with the assumptions that failure will occur. You should become aware of these failures, how they occurred, how to respond to them and then plan on how to prevent these from happening again.

- How are you backing up your data?
- How does your system withstand component failures?
- How are you planning for recovery?

Key AWS Services

- Foundations
  - IAM, VPC
- Change Management
  - AWS CloudTrail
- Failure Management
  - AWS CloudFormation

Exam Tips - Reliability Pillar - Questions

- Foundations
  - How are you managing AWS service limits for your account?
  - How are you planning your network topology on AWS?
  - Do you have an escalation path to deal with technical issues?
- Change management
  - How does your system adapt to changes in demand?
  - How are you monitoring AWS resources?
  - How are you executing change management?
- Failure Management
  - How are you backing up your data?
  - How does your system withstand component failures?
  - How are you planning for recovery?

The Performance Efficiency Pillar

The Performance Efficiency pillar focuses on how to use computing resources efficiently to meet your requirements and how to maintain that efficiency as demand changes and technology evolves.

Design Principles

- Democratize advanced technologies
- Go global in minutes
- Use server-less architectures

Definition

Performance Efficiency in the cloud consists of 4 areas;

- Compute
- Storage
- Database
- Space-time trade-off

Best Practices - Compute

When architecting your system it is important to choose the right kind of server. Some applications require heavy CPU utilization, some require heavy memory utilization etc.

With AWS servers are virtualized and at the click of a button (or API call) you can change the type of server in which your environment is running on. You can even switch to running with no servers at all and use AWS Lambda.

Best Practices - Compute Questions

- How do you select the appropriate instance type for your system?
- How do you ensure that you continue to have the most appropriate instance type as new instance types and features are introduced?
- How do you monitor your instances post launch to ensure they are performing as expected?
- How do you ensure that the quantity of your instances matches demand?

Best Practices - Storage

The optimal storage solutions for your environment depends on a number of factors. For example;

- Access Method - Block, File or Object
- Patterns of Access - Random or Sequential
- Throughput Required
- Frequency of Access - Online, Offline or Archival
- Frequency of Update - Worm, Dynamic
- Availability Constraints
- Durability Constraints

Best Practices - Storage

At AWS the storage is virtualized. With S3 you can have 11 x 9's durability, Cross Region Replication etc. With EBS you can choose between different storage mediums (such as SSD, Magnetic, PIOPS etc). You can also easily move volumes between the different types of storage mediums.

Best Practices - Storage Questions

- How do you select the appropriate storage solution for your system?
- How do you ensure that you continue to have the most appropriate storage solution as new storage solutions and features are launched?
- How do you monitor your storage solution to ensure it is performing as expected?
- How do you ensure that the capacity and throughput of your storage solutions matches demand?

Best Practices - Database

The optimal database solution depends on a number of factors. Do you need database consistency, do you need high availability, do you need No-SQL, do you need DR etc?

With AWS you get a LOT of options. RDS, DynamoDB, Redshift etc.

Best Practices - Database Questions

- How do you select the appropriate database solution for your system?
- How do you ensure that you continue to have the most appropriate database solution and features as new database solution and features are launched?
- How do you monitor your databases to ensure performance is as expected?
- How do you ensure the capacity and throughput of your databases matches demand?

Best Practices - Space-Time trade-off

Using AWS you can use services such as RDS to add read replicas, reducing the load on your database and creating multiple copies of the database. This helps to lower latency.

You can use Direct Connect to provide predictable latency between your HQ and AWS.

You can use the global infrastructure to have multiple copies of your environment, in regions that is closest to your customer base.

You can also use caching services such as ElastiCache or CloudFront to reduce latency

Best Practices - Space-Time trade-off

- How do you select the appropriate proximity and caching solutions for your system?
- How do you ensure that you continue to have the most appropriate proximity and caching solutions as new solutions are launched?
- How do you monitor your proximity and caching solutions to ensure performance is as expected?
- How do you ensure that the proximity and caching solutions you have matches demand?

Key AWS Services

- Compute
  - Autoscaling
- Storage
  - EBS, S3, Glacier
- Database
  - RDS, DynamoDB, Redshift
- Space-Time Trade-Off
  - CloudFront, ElastiCache, Direct Connect, RDS Read Replicas etc

The Cost Optimization Pillar

Use the Cost Optimization pillar to reduce your costs to a minimum and use those savings for other parts of your business. A cost-optimized system allows you to pay the lowest price possible while still achieving your business objectives.

At A Cloud Guru, we built our platform on Serverless Technology, allowing us to have an extremely low cost platform (and we can then pass those savings on to you).

Brng down test environments when not in use.. keep them up and running only for 40 hours instead of 160

Design Principles

- Transparently attribute expenditure
- Use managed services to reduce cost of ownership
- Trade capital expense for operating expense
- Benefit from economies of scale
- Stop spending money on data center operations

Cost Optimization in the cloud consists of 4 areas;

- Matched supply and demand
- Cost-effective resources
- Expenditure awareness
- Optimizing over time

Best Practices - Matched supply and demand

Try to optimally align supply with demand. Don't over provision or under provision, instead as demand grows, so should your supply of compute resources. Think of things like Autoscaling which scale with demand. Similarly in a server-less context, use services such as Lambda that only execute (or respond) when a request (demand) comes in.

Services such as CloudWatch can also help you keep track as to what your demand is.

Best Practices - Cost-Effective Resources

Using the correct instance type can be key to cost savings. For example you might have a reporting process that is running on a t2-Micro and it takes 7 hours to complete. That same process could be run on an m4.2xlarge in a manner of minutes. The result remains the same but the t2.micro is more expensive because it ran for longer.

A well architected system will use the most cost efficient resources to reach the end business goal.

Above is important

Best Practices - Cost-Effective Resources Questions

- Have you selected the appropriate resource types to meet your cost targets?
- Have you selected the appropriate pricing model to meet your cost targets?
- Are there managed services (higher-level services than Amazon EC2, Amazon EBS, and Amazon S3) that you can use to improve your ROI?

Best Practices - Expenditure Awareness

With cloud you no longer have to go out and get quotes on physical servers, choose a supplier, have those resources delivered, installed, made available etc. You can provision things within seconds, however this comes with its own issues. Many organisations have different teams, each with their own AWS accounts. Being aware of what each team is spending and where is crucial to any well architected system. You can use cost allocation tags to track this, billing alerts as well as consolidated billing.

Best Practices - Expenditure Awareness Questions

A CLOUD GURU

- What access controls and procedures do you have in place to govern AWS costs?
- How are you monitoring usage and spending?
- How do you decommission resources that you no longer need, or stop resources that are temporarily not needed?
- How do you consider data-transfer charges when designing your architecture?

Best Practices - Optimizing Over Time

A CLOUD GURU

AWS moves FAST. There are hundreds of new services (and potentially 1000 new services this year). A service that you chose yesterday may not be the best service to be using today. For example consider MySQL RDS, Aurora was launched at re:invent 2014 and is now out of preview. Aurora may be a better option now for your business because of its performance and redundancy. You should keep track of the changes made to AWS and constantly re-evaulate your existing architecture. You can do this by subscribing to the AWS blog and by using services such as Trusted Advisor.

Key AWS Services

- Matched supply and demand
  - Autoscaling
- Cost-effective resources
  - EC2 (reserved instances), AWS Trusted Advisor
- Expenditure awareness
  - CloudWatch Alarms, SNS
- Optimizing over time
  - AWS Blog, AWS Trusted Advisor

Exam Tips - Cost Optimization - Questions

- Matched Supply & Demand
  - How do you make sure your capacity matches but does not substantially exceed what you need?
  - How are you optimizing your usage of AWS services?
- Cost-effective resources
  - Have you selected the appropriate resource types to meet your cost targets?
  - Have you selected the appropriate pricing model to meet your cost targets?
  - Are there managed services (higher-level services than Amazon EC2, Amazon EBS, and Amazon S3) that you can use to improve your ROI?

Exam Tips - Cost Optimization - Questions

- Expenditure awareness
  - What access controls and procedures do you have in place to govern AWS costs?
  - How are you monitoring usage and spending?
  - How do you decommission resources that you no longer need, or stop resources that are temporarily not needed?
  - How do you consider data-transfer charges when designing your architecture?
- Optimizing over time
  - How do you manage and/or consider the adoption of new services?

Operational Excellence Pillar

The Operational Excellence pillar includes operational practices and procedures used to manage production workloads.

This includes how planned changes are executed, as well as responses to unexpected operational events.

Change execution and responses should be automated. All processes and procedures of operational excellence should be documented, tested, and regularly reviewed.

Design Principles

- Perform operations with code
- Align operations processes to business objectives
- Make regular, small, incremental changes
- Test for responses to unexpected events
- Learn from operational events and failures
- Keep operations procedures current

There are three best practice areas for Operational Excellence in the cloud:

- Preparation
- Operation
- Response

Best Practices - Preparation

A CLOUD GURU

Effective preparation is required to drive operational excellence.

Operations checklists will ensure that workloads are ready for production operation, and prevent unintentional production promotion without effective preparation.

Workloads should have;

Runbooks - operations guidance that operations teams can refer to so they can perform normal daily tasks.

Playbooks - Guidance for responding to unexpected operational events. Playbooks should include response plans, as well as escalation paths and stakeholder notifications.

Best Practices - Preparation

A CLOUD GURU

In AWS there are several methods, services, and features that can be used to support operational readiness, and the ability to prepare for normal day-to-day operations as well as unexpected operational events.

CloudFormation can be used to ensure that environments contain all required resources when deployed in production, and that the configuration of the environment is based on tested best practices, which reduces the opportunity for human error.

Lecture 197

Best Practices - Preparation

A CLOUD GUR

Implementing **Auto Scaling**, or other automated scaling mechanisms, will allow workloads to automatically respond when business-related events affect operational needs.

Services like **AWS Config** with the AWS Config rules feature create mechanisms to automatically track and respond to changes in your AWS workloads and environments.

It is also important to use features like **tagging** to make sure all resources in a workload can be easily identified when needed during operations and responses.

Lecture 197

Best Practices - Preparation

A CLOUD GU

The following questions focus on preparation considerations for operational excellence:

- What best practices for cloud operations are you using?
- How are you doing configuration management for your workload?

Lecture 197

Best Practices - Preparation

A CLOUD GU

Be sure that documentation does not become stale or out of date as procedures change. Also make sure that it is thorough. Without application designs, environment configurations, resource configurations, response plans, and mitigation plans, documentation is not complete. If documentation is not updated and tested regularly, it will not be useful when unexpected operational events occur. If workloads are not reviewed before production, operations will be affected when undetected issues occur. If resources are not documented, when operational events occur, determining how to respond will be more difficult while the correct resources are identified.

06:31 / 14:01 Transcript Browse Q&A Add Bookmark Continue

Best Practices - Operations

A CLOUD GU

Operations should be standardized and manageable on a routine basis. The focus should be on automation, small frequent changes, regular quality assurance testing, and defined mechanisms to track, audit, roll back, and review changes. Changes should not be large and infrequent, they should not require scheduled downtime, and they should not require manual execution. A wide range of logs and metrics that are based on key operational indicators for a workload should be collected and reviewed to ensure continuous operations.

Best Practices - Operations

A CLOUD G

In AWS you can set up a continuous integration / continuous deployment (CI/CD) pipeline (e.g., source code repository, build systems, deployment and testing automation). Release management processes, whether manual or automated, should be tested and be based on small incremental changes, and tracked versions. You should be able to revert changes that introduce operational issues without causing operational impact.

Lecture 197

Best Practices - Operations

A CLOUD GU

Routine operations, as well as responses to unplanned events, should be automated. Manual processes for deployments, release management, changes, and rollbacks should be avoided. Releases should not be large batches that are done infrequently. Rollbacks are more difficult in large changes, and failing to have a rollback plan, or the ability to mitigate failure impacts, will prevent continuity of operations. Align monitoring to business needs, so that the responses are effective at maintaining business continuity. Monitoring that is ad hoc and not centralized, with responses that are manual, will cause more impact to operations during unexpected events.

09:05 / 14:01 Transcript Browse Q&A Add Bookmark Continue

Lecture 197

Best Practices - Responses

A CLOUD GU

Responses to unexpected operational events should be automated. This is not just for alerting, but also for mitigation, remediation, rollback, and recovery. Alerts should be timely, and should invoke escalations when responses are not adequate to mitigate the impact of operational events. Quality assurance mechanisms should be in place to automatically roll back failed deployments. Responses should follow a pre-defined playbook that includes stakeholders, the escalation process, and procedures. Escalation paths should be defined and include both functional and hierarchical escalation capabilities. Hierarchical escalation should be automated, and escalated priority should result in stakeholder notifications.

In AWS there are several mechanisms to ensure both appropriate alerting and notification in response to unplanned operational events, as well as automated responses.

10:06 / 14:01 Transcript Browse Q&A Add Bookmark Continue

Preparation: AWS Config provides a detailed inventory of your AWS resources and configuration, and continuously records configuration changes. AWS Service Catalog helps to create a standardized set of service offerings that are aligned to best practices. Designing workloads that use automation with services like Auto Scaling, and Amazon SQS, are good methods to ensure continuous operations in the event of unexpected operational events.

Operations: AWS CodeCommit, AWS CodeDeploy, and AWS CodePipeline can be used to manage and automate code changes to AWS workloads. Use AWS SDKs or third-party libraries to automate operational changes. Use AWS CloudTrail to audit and track changes made to AWS environments.

Exam Tips - Operational Excellence - Questions

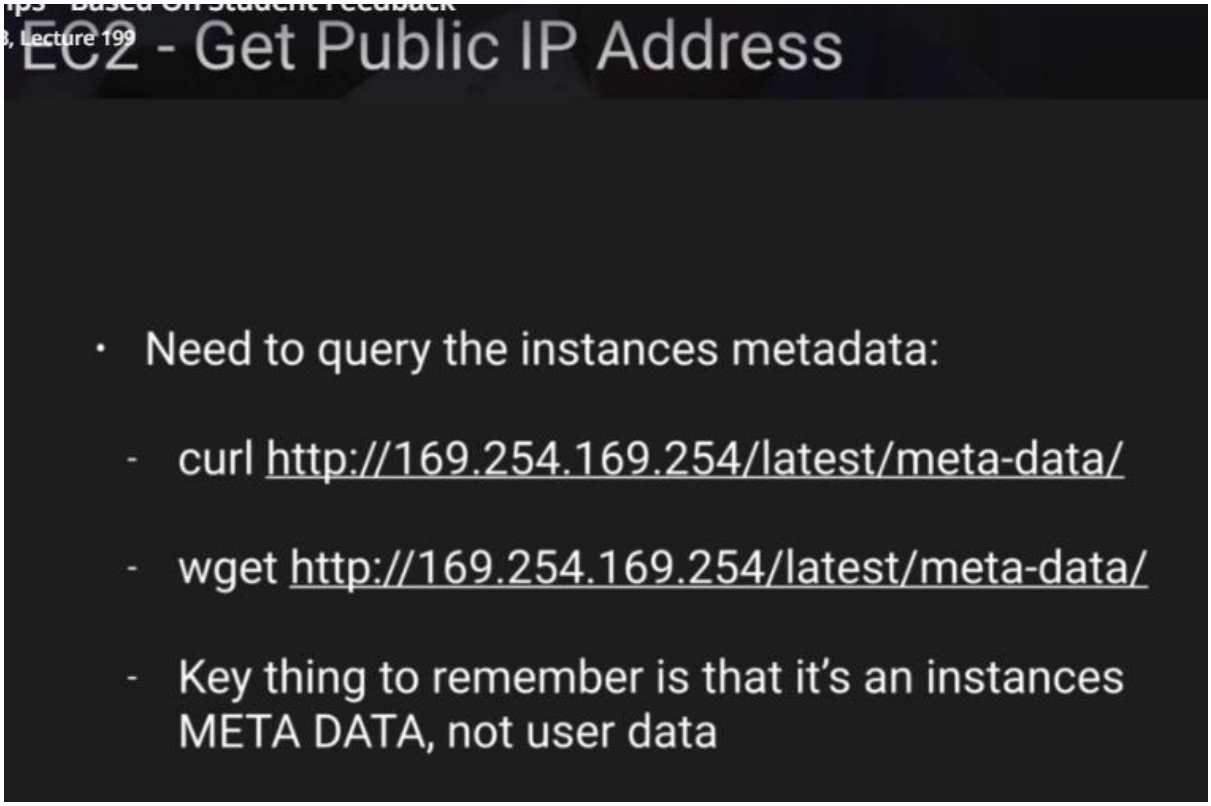
A CLOUD GURU

- Preparation
  - What best practices for cloud operations are you using?
  - How are you doing configuration management for your workload?
- Operations
  - How are you evolving your workload while minimizing the impact of change?
  - How do you monitor your workload to ensure it is operating as expected?
- Responses
  - How do you respond to unplanned operational events?
  - How is escalation managed when responding to unplanned operational events?

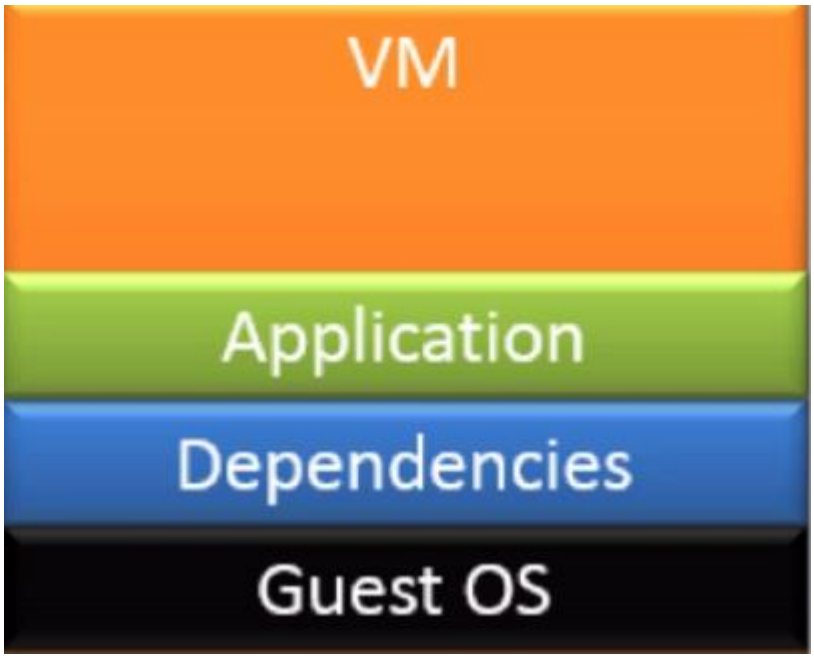
167 people bookmarked this moment
13:45 / 14:01 Transcript Browse Q&A Add Bookmark Continue

OpsWorks

- Orchestration Service that uses Chef
- Chef consists of recipes to maintain a consistent state
- Look for the term “chef” or “recipes” or “cook books” and think OpsWorks



Docker:



Dependencies could be tomcat php etc

What is Docker?

A CLOUD GU

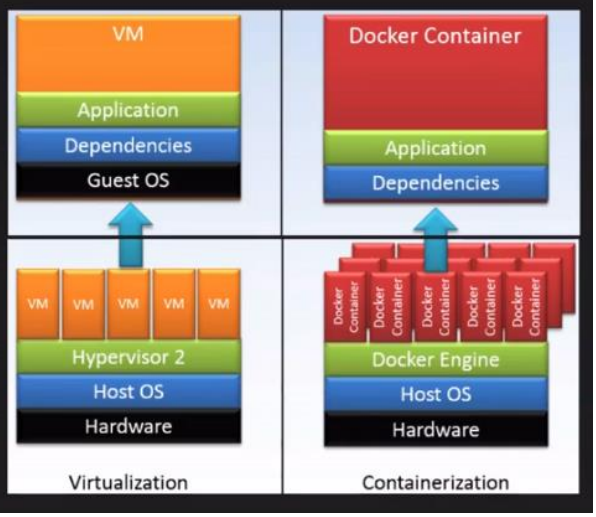
- Docker is a software platform that allows you to build, test, and deploy applications quickly.
- Docker is highly reliable: you can quickly deploy and scale applications into any environment and know your code will run.
- Docker is infinitely scalable: Running Docker on AWS is a great way to run distributed applications at any scale.
- Docker packages software into standardized units called Containers:
  - Containers allow you to easily package an application's code, configurations, and dependencies into easy to use building blocks that deliver environmental consistency, operational efficiency, developer productivity, and version control.

Virtualisation vs Containerisation

A CLOUD GURU

Virtualisation vs containerisation

- Traditional Virtual Machine
- Container.
- And thats where the technologies diverge.
- Traditional virtualisation has density compromises.
- Docker achieves higher density, and improved portability by removing the per container Guest OS



Containerisation Benefits

A CLOUD GURU

- Escape from dependancy hell....
- Consistent progression from DEV -> TEST -> QA -> PROD
- Isolation - performance or stability issues with App A in container A, wont impact App B in Container B.
- Much better resource management
- Extreme code portability
- Micro-Services

Docker Components

- Docker Image
- Docker Container
- Layers / Union File System
- DockerFile
- Docker Daemon/ Engine
- Docker Client
- Docker Registries / Docker Hub

Amazon EC2 Container Service (Amazon ECS) is a highly scalable, fast, container management service that makes it easy to run, stop, and manage Docker containers on a cluster of EC2 instances. Amazon ECS lets you launch and stop container-based applications with simple API calls, allows you to get the state of your cluster from a centralized service, and gives you access to many familiar Amazon EC2 features.

ECS is a managed version of docker by amazon.

- ECS is a Regional service that you can use in one or more AZs across a new, or existing, VPC to schedule the placement of containers across your cluster based on your resource needs, isolation policies, and availability requirements.
- Amazon ECS eliminates the need for you to operate your own cluster management and configuration management systems, or to worry about scaling your management infrastructure.
- ECS can also be used to create a consistent deployment and build experience, manage and scale batch and ETL workloads, and build sophisticated application architectures on a microservices model.

- Containers are a method of operating system virtualization that allow you to run an application and its dependencies in resource-isolated processes.
- Containers have everything the software needs to run — including libraries, system tools, code, and runtime.
- Containers are created from a read-only template called an Image.

What's a Docker Image?

A CLOUD GUR

- An Image is a read-only template with instructions for creating a Docker container. It contains:
  - an ordered collection of root filesystem changes and the corresponding execution parameters for use within a container runtime.
- An Image is created from a Dockerfile, a plain text file that specifies the components that are to be included in the container.
- Images are stored in a Registry, such as DockerHub or AWS ECR.

Container Registries

A CLOUD GUR

- Amazon EC2 Container Registry (Amazon ECR) is a managed AWS Docker registry service that is secure, scalable, and reliable. Amazon ECR supports private Docker repositories with resource-based permissions using AWS IAM so that specific users or Amazon EC2 instances can access repositories and images. Developers can use the Docker CLI to push, pull, and manage images.

ECS Task Definitions

A CLOUD GUR

- A Task Definition is required to run Docker containers in Amazon ECS.
- Task Definitions are text files in JSON format that describe one or more containers that form your application.
- Some of the parameters you can specify in a task definition include:
  - Which Docker images to use with the containers in your task
  - How much CPU and memory to use with each container
  - Whether containers are linked together in a task

ECS Task Definitions (cont.)

A CLOUD GURU

- The Docker networking mode to use for the containers in your task
- What (if any) ports from the container are mapped to the host container instance
- Whether the task should continue to run if the container finishes or fails
- The command the container should run when it is started
- What (if any) environment variables should be passed to the container when it starts
- Any data volumes that should be used with the containers in the task
- What (if any) IAM role your tasks should use for permissions

ECS Services

A CLOUD GUR

- An Amazon ECS service allows you to run and maintain a specified number (or, the "desired count") of instances of a task definition simultaneously in an ECS cluster.
- Think of Services like Auto-Scaling groups for ECS.
- If a task should fail or stop, the Amazon ECS service scheduler launches another instance of your task definition to replace it and maintain the desired count of tasks in the service.

ECS Clusters

A CLOUD GU

- An Amazon ECS cluster is a logical grouping of container instances that you can place tasks on. When you first use the Amazon ECS service, a default cluster is created for you, but you can create multiple clusters in an account to keep your resources separate.
- Concepts:
  - Clusters can contain multiple different container instance types.
  - Clusters are region-specific.
  - Container instances can only be part of one cluster at a time.
  - You can create IAM policies for your clusters to allow or restrict users' access to specific clusters.

ECS Scheduling

A CLOUD GU

- Service Scheduler:
  - Ensures that ensures that the specified number of tasks are constantly running and reschedules tasks when a task fails (for example, if the underlying container instance fails for some reason).
  - Can ensure tasks are registered against an ELB.
- Custom Scheduler:
  - You can create your own schedulers that meet your business needs.
  - Leverage third-party schedulers, such as Blox.
- The Amazon ECS schedulers leverage the same cluster state information provided by the Amazon ECS API to make appropriate placement decisions.

ECS Container Agent

A CLOUD GURU

The Amazon ECS container agent allows container instances to connect to your cluster. The Amazon ECS container agent is included in the Amazon ECS-optimized AMI, but you can also install it on any EC2 instance that supports the Amazon ECS specification. The Amazon ECS container agent is only supported on EC2 instances.

- Pre-installed on special ECS AMIs.
- Linux-based:
  - Works with Amazon Linux, Ubuntu, Red Hat, CentOS, etc.
  - Will **not** work with Windows

ECS Security

A CLOUD G

- IAM Roles:
  - EC2 instances use an IAM role to access ECS.
  - ECS tasks use an IAM role to access services and resources.
- Security Groups attach at the instance-level (i.e. the host … not the task or container.)
- You can access and configure the OS of the EC2 instances in your ECS cluster.

ECS Limits

- Soft Limits:
  - Clusters per Region (default = 1000)
  - Instances per Cluster (default = 1000)
  - Services per Cluster (default = 500)
- Hard Limits
  - One Load Balancer per Service
  - 1000 Tasks per Service (the "desired count")
  - Max. 10 Containers per Task Definition
  - Max. 10 Tasks per instance (host)

ECS Exam Tips

A CLOUD GUR

- ECS - Amazon's managed EC2 container service. Allows you to manage Docker containers on a cluster of EC2 instances
- Containers are a method of operating system virtualization that allow you to run an application and its dependencies in resource-isolated processes.
- Containers are created from a read-only template called an Image.
- An Image is a read-only template with instructions for creating a Docker container.
- Images are stored in a Registry, such as DockerHub or AWS ECR.
- Amazon EC2 Container Registry (Amazon ECR) is a managed AWS Docker registry service

ECS Exam Tips

A CLOUD GUR

- A Task Definition is required to run Docker containers in Amazon ECS.
- Task Definitions are text files in JSON format that describe one or more containers that form your application.
- Think of a task definition as a cloud formation template but for docker. Configure things such as the amount of CPU, RAM etc
- An Amazon ECS service allows you to run and maintain a specified number (or, the "desired count") of instances of a task definition simultaneously in an ECS cluster.
- Think of Services like Auto-Scaling groups for ECS.
- An Amazon ECS cluster is a logical grouping of container instances that you can place tasks on.

ECS Exam Tips

A CLOUD GUR

- Clusters can contain multiple different container instance types.
- Clusters are region-specific.
- Container instances can only be part of one cluster at a time.
- You can create IAM policies for your clusters to allow or restrict users' access to specific clusters.
- You can schedule ECS in two ways
  - Service Scheduler
  - Customer scheduler
- ECS agent to connect EC2 instances to your ECS cluster. LINUX ONLY
- IAM with ECS to restrict access
- Security groups operate at the instance level, not at the task or container level

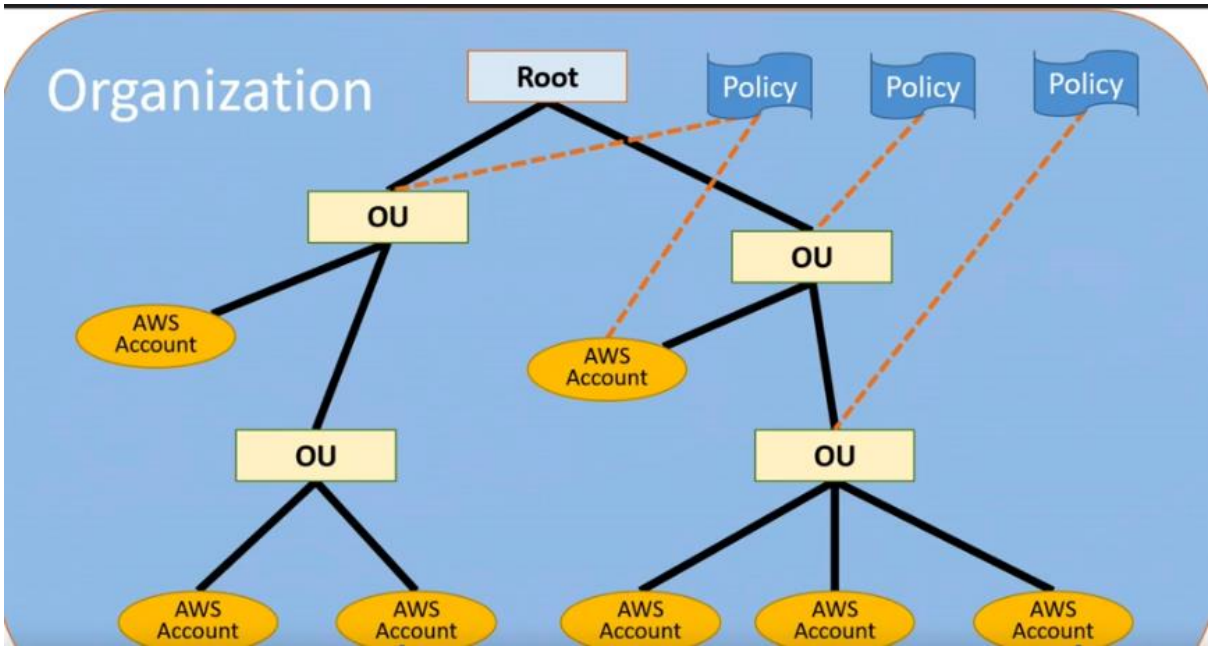
201 people bookmarked this moment.

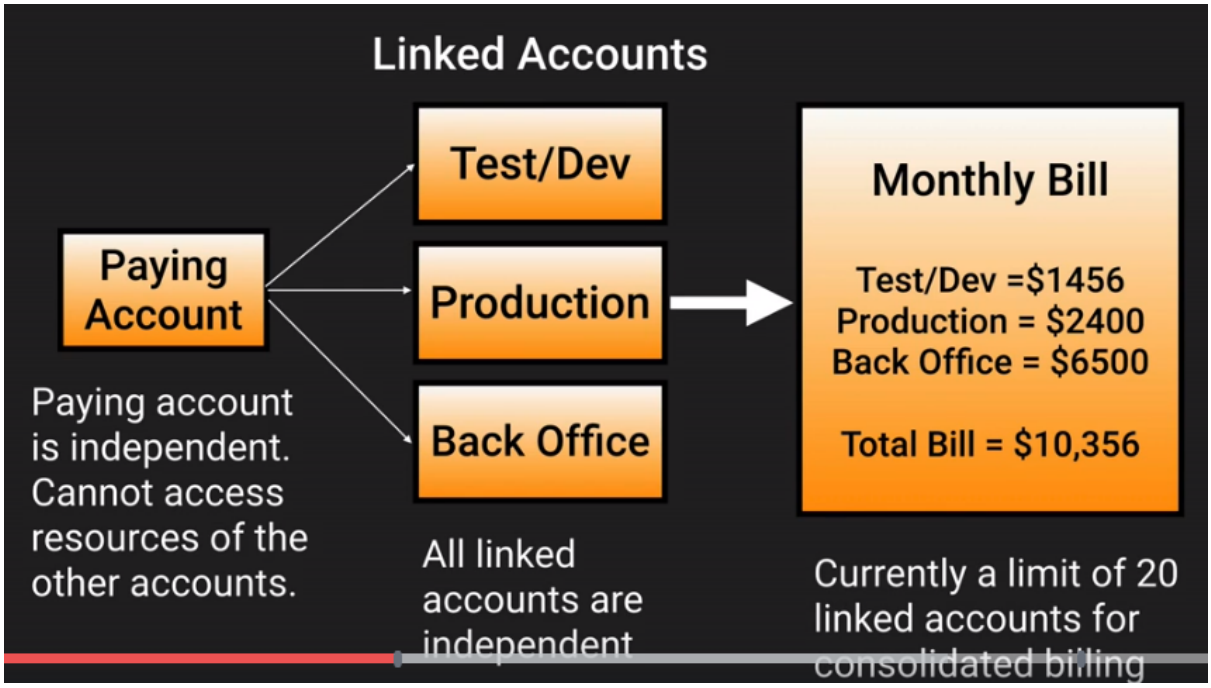
AWS Organizations & Consolidated Billing

AWS Organizations is an account management service that enables you to consolidate multiple AWS accounts into an organization that you create and centrally manage.

Available in two feature sets;

- Consolidated Billing
- All Features



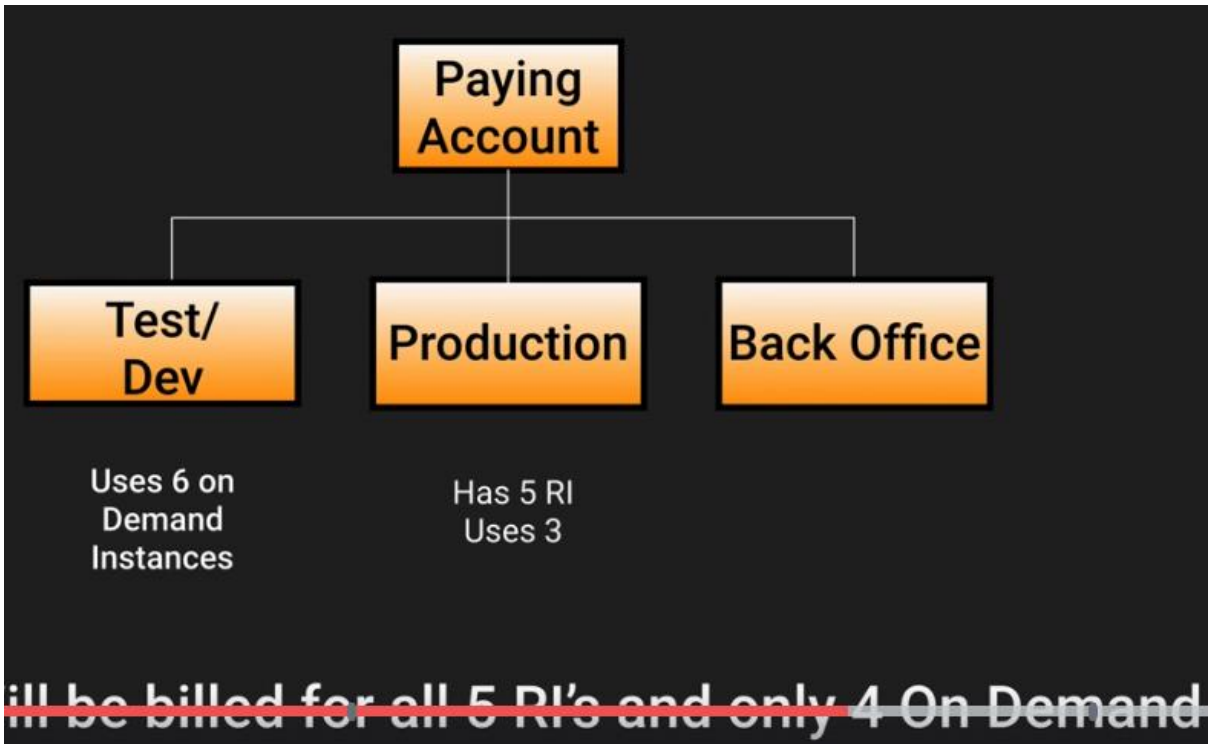


- One bill per AWS account
- Very easy to track charges and allocate costs
- Volume pricing discount

Lecture 200

S3 pricing

	Standard Storage
First 1 TB / month	$0.0300 per GB
Next 49 TB / month	$0.0295 per GB
Next 450 TB / month	$0.0290 per GB
Next 500 TB / month	$0.0285 per GB
Next 4000 TB / month	$0.0280 per GB
Over 5000 TB / month	$0.0275 per GB



Best practices:

- Always enable multi-factor authentication on root account.
- Always use a strong and complex password on root account
- Paying account should be used for billing purposes only. Do not deploy resources in to paying account.

Other Things To Note

- Linked Accounts
  - 20 linked accounts only
  - To add more visit https://aws-portal.amazon.com/gp/aws/html-forms-controller/contactus/aws-account-and-billing

- Billing Alerts
  - When monitoring is enabled on the paying account the billing data for all linked accounts is included
  - You can still create billing alerts per individual account

- CloudTrail
  - Per AWS Account and is enabled per region.
  - Can consolidate logs using an S3 bucket
    1. Turn on CloudTrail in the paying account
    2. Create a bucket policy that allows cross account access
    3. Turn on CloudTrail in the other accounts and use the bucket in the paying account

- Consolidated billing allows you to get volume discounts on all your accounts.

- Unused reserved instances for EC2 are applied across the group.

- CloudTrail is on a per account and per region basis but can be aggregated in to a single bucket in the paying account.

AWS Organizations allows you to manage multiple AWS accounts at once. With Organizations, you can create groups of accounts and then apply policies to those groups.

AWS Organizations

- Centrally Manage Policies Across Multiple AWS Accounts
- Control Access To AWS Services
- Automate AWS Account Creation And Management
- Consolidate Billing Across Multiple AWS Accounts

Central Management

AWS Organizations allows you to manage multiple AWS accounts at once. You can create groups of accounts, and then attach policies to a group to ensure the correct policies are applied across the accounts. Organizations enables you to centrally manage policies across multiple accounts, without requiring custom scripts and manual processes.

Control Access

With AWS Organizations, you can create Service Control Policies (SCPs) that centrally control AWS service use across multiple AWS accounts. You can specifically Allow or Deny individual AWS Services. For example you could deny the use of Kinesis or DynamoDB to you HR group within your AWS Organization. Even if IAM in that account allows it, SCP will override it.

Automate AWS Account Creation

You can use the AWS Organizations APIs to automate the creation and management of new AWS accounts. The Organizations APIs enable you to create new accounts programmatically, and to add the new accounts to a group. The policies attached to the group are automatically applied to the new account.

Consolidated Billing

AWS Organizations enables you to set up a single payment method for all the AWS accounts in your organization through consolidated billing. With consolidated billing, you can see a combined view of charges incurred by all your accounts, as well as take advantage of pricing benefits from aggregated usage, such as volume discounts for Amazon EC2 and Amazon S3.

Cross Account Access:

What is Cross Account Access?

A CLOUD GURU

Many AWS customers use separate AWS accounts for their development and production resources. This separation allows them to cleanly separate different types of resources and can also provide some security benefits.

Cross account access makes it easier for you to work productively within a multi-account (or multi-role) AWS environment by making it easy for you to switch roles within the AWS Management Console. You can now sign in to the console using your IAM user name then switch the console to manage another account without having to enter (or remember) another user name and password.

Steps

- Identify our account numbers
- Create a group in IAM - Dev
- Create a user in IAM - Dev
- Log in to Production
- Create the “read-write-app-bucket” policy
- Create the “UpdateApp” Cross Account Role
- Apply the newly created policy to the role
- Log in to the Developer Account
- Create a new inline policy
- Apply it to the Developer group
- Login as John
- Switch Accounts

153 people bookmarked this moment.

What Are Tags?

- Key Value Pairs attached to AWS resources
- Metadata (data about data)
- Tags can sometimes be inherited
  - Autoscaling, CloudFormation, and Elastic Beanstalk can create other resources

TAGGING & RESOURCE GROUPS

ource Groups & Tagging

on 23, Lecture 208

What Are Resource Groups?

A CLOUD GU

Resource groups make it easy to group your resources using the tags that are assigned to them. You can group resources that share one or more tags.

Resource groups contain information such as;

- Region
- Name
- Health Checks

Specific information

- For EC2 - Public & Private IP Addresses
- For ELB - Port Configurations
- For RDS - Database Engine etc

people bookmarked this moment.

AWS Resouce Groups

- Classic Resource Groups
- AWS Systems Manager

lecture 206

Security Token Service (STS)

A CLOUD GU

Grants users limited and temporary access to AWS resources. Users can come from three sources;

Federation (typically Active Directory)

- Uses Security Assertion Markup Language (SAML)
- Grants temporary access based off the users Active Directory credentials. Does not need to be a user in IAM
- Single sign on allows users to log in to AWS console without assigning IAM credentials

- Federation with Mobile Apps
  - Use Facebook/Amazon/Google or other OpenID providers to log in
- Cross Account Access
  - Let's users from one AWS account access resources in another

Lecture 206

Understanding The Key Terms

A CLOUD GU

- Federation
  - Combining or joining a list of users in one domain (such as IAM) with a list of users in another domain (such as Active Directory, Facebook etc)
- Identity Broker
  - A service that allows you to take an identity from point A and join it (federate it) to point B
- Identity Store
  - Services like Active Directory, Facebook, Google etc.
- Identities
  - A user of a service like Facebook etc.

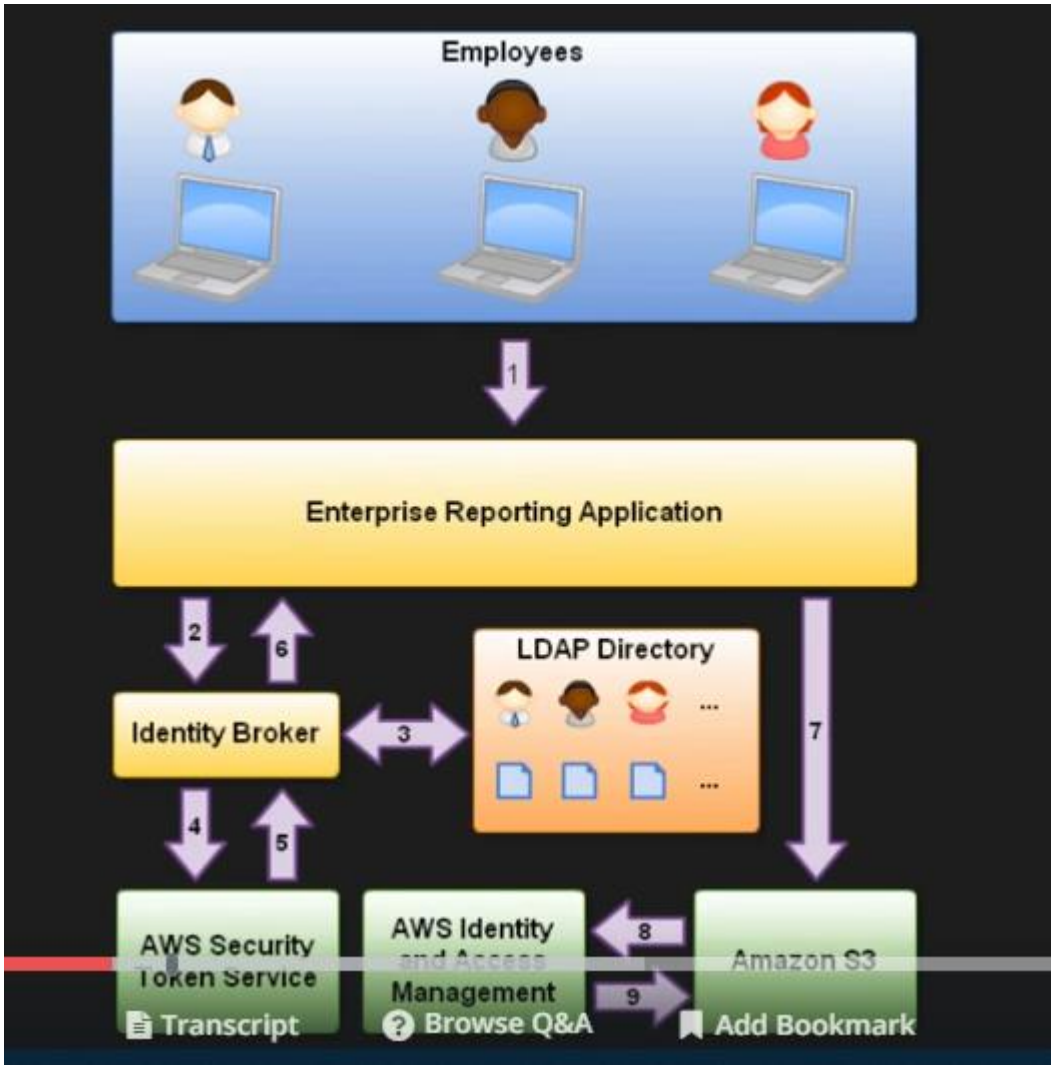
Lecture 206

Scenario

A CLOUD GUR

You are hosting a company website on some EC2 web servers in your VPC. Users of the website must log in to the site which then authenticates against the companies active directory servers which are based on site at the companies head quarters.

Your VPC is connected to your company HQ via a secure IPSEC VPN. Once logged in the user can only have access to their own S3 bucket. How do you set this up?



Lecture 206

Scenario

1. Employee enters their username and password
2. The application calls an Identity Broker. The broker captures the username and password.
3. The Identity Broker uses the organization's LDAP directory to validate the employee's identity.
4. The Identity Broker calls the new GetFederationToken function using IAM credentials. The call must include an IAM policy and a duration (1 to 36 hours), along with a policy that specifies the permissions to be granted to the temporary security credentials.
5. The Security Token Service confirms that the policy of the IAM user making the call to GetFederationToken gives permission to create new tokens and then returns four values to the application: An access key, a secret access key, a token, and a duration (the token's lifetime).

6. The Identity Broker returns the temporary security credentials to the reporting application

7. The data storage application uses the temporary security credentials (including the token) to make requests to Amazon S3

8. Amazon S3 uses IAM to verify that the credentials allow the requested operation on the given S3 bucket and key

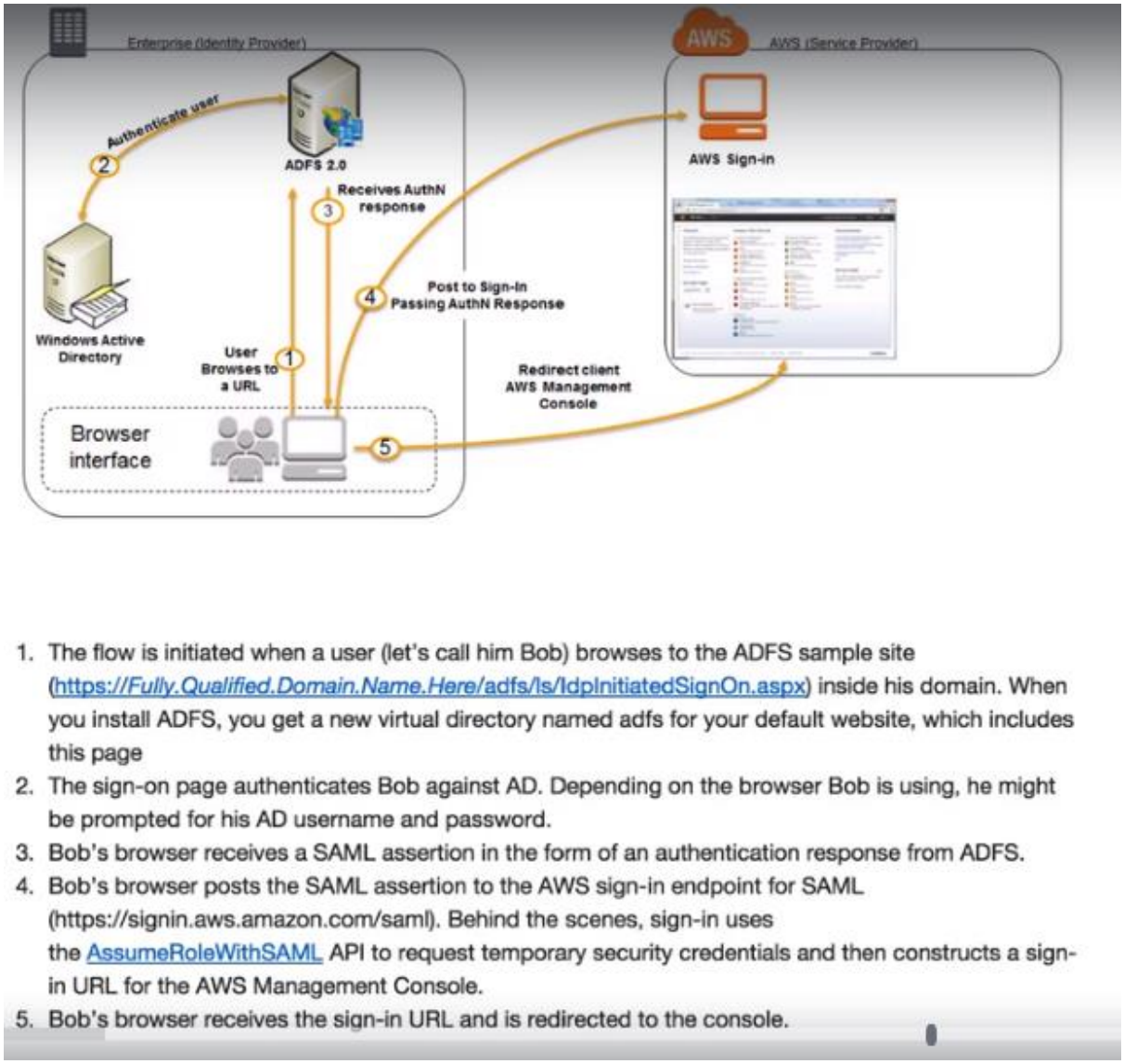
9. IAM provides S3 with the go-ahead to perform the requested operation

1. Develop an Identity Broker to communicate with LDAP and AWS STS.

2. Identity Broker always authenticates with LDAP first, THEN with AWS STS

3. Application then gets temporary access to AWS resources

LDAP Intergration



1. The flow is initiated when a user (let's call him Bob) browses to the ADFS sample site (https://*Fully.Qualified.Domain.Name.Here*/adfs/ls/IdpInitiatedSignOn.aspx) inside his domain. When you install ADFS, you get a new virtual directory named adfs for your default website, which includes this page
2. The sign-on page authenticates Bob against AD. Depending on the browser Bob is using, he might be prompted for his AD username and password.
3. Bob's browser receives a SAML assertion in the form of an authentication response from ADFS.
4. Bob's browser posts the SAML assertion to the AWS sign-in endpoint for SAML (https://signin.aws.amazon.com/saml). Behind the scenes, sign-in uses the AssumeRoleWithSAML API to request temporary security credentials and then constructs a sign-in URL for the AWS Management Console.
5. Bob's browser receives the sign-in URL and is redirected to the console.

SAML is used for LDAP

Authenticate first with LDAP then temporary security token will be given.

Workspaces:

What is Workspaces?

It's basically VDI. A WorkSpace is a cloud-based replacement for a traditional desktop. A WorkSpace is available as a bundle of compute resources, storage space, and software application access that allow a user to perform day-to-day tasks just like using a traditional desktop. A user can connect to a WorkSpace from any supported device (PC, Mac, Chromebook, iPad, Kindle Fire, or Android tablets) using a free Amazon WorkSpaces client application and credentials set up by an administrator, or their existing Active Directory credentials if Amazon WorkSpaces is integrated with an existing Active Directory domain.

0:32 / 3:07 Transcript Browse Q&A Add Bookmark Continue

Workspaces - Quick Facts

- Windows 7 Experience, provided by Windows Server 2008 R2.
- By default, users can personalize their WorkSpaces with their favorite settings for items such as wallpaper, icons, shortcuts, etc. This can be locked down by an administrator however.
- By default you will be given local administrator access, so you can install your own appplications.
- Workspaces are persistent.
- All data on the D:\ is backed up every 12 hours.
- You do not need an AWS account to login to workspaces