AWS Certified Solutions Architect - Associate

Week 3 – Content Review

22 JULY 2021

DHAVAL SONI

infostretch

Agenda

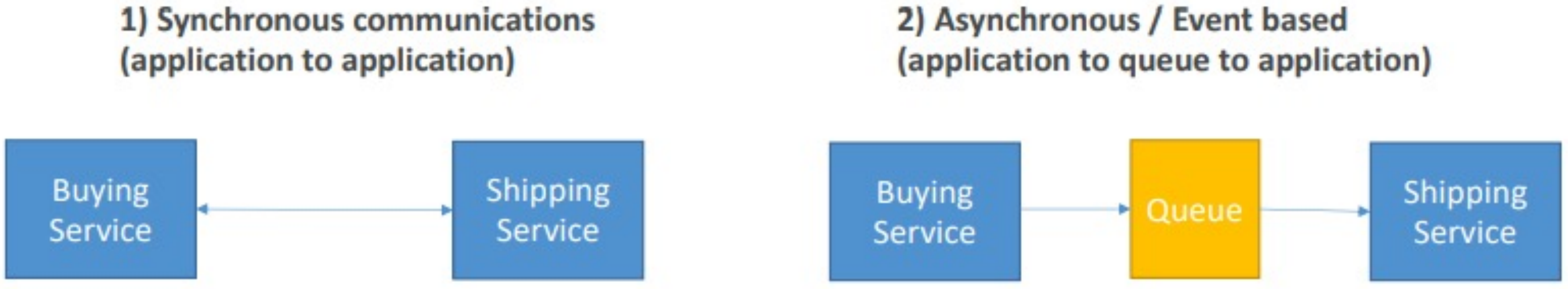
- Week 3 content review
  - Decoupling Applications: SNS, SQS, Active MQ, Kinesis
  - Containers on AWS: ECS, Fargate, EKS
  - AWS Monitoring & Audit: CloudWatch, CloudTrail
  - Databases in AWS
  - Lambda, DynamoDB, API Gateway
- Topics for Week 4
- Q & A

infostretch

CONTENT REVIEW

Decoupling Applications

- When we start deploying multiple applications, they will inevitably need to communicate with one another
- There are two patterns of application communication



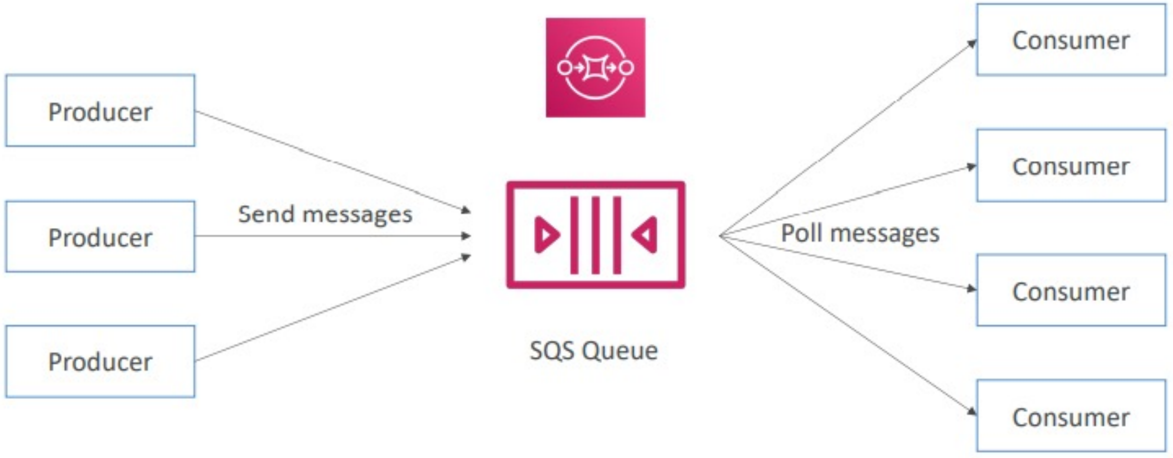
1) Synchronous communications (application to application)

2) Asynchronous / Event based (application to queue to application)

- Synchronous between applications can be problematic if there are sudden spikes of traffic
- What if you need to suddenly encode 1000 videos but usually it's 10?
- In that case, it's better to **decouple** your applications,
  - using SQS: queue model
  - using SNS: pub/sub model
  - using Kinesis: real-time streaming model

Amazon SQS

- Oldest offering (over 10 years old)
- Fully managed service, used to decouple applications
- **Attributes:**
  - Unlimited throughput, unlimited number of messages in queue
  - Default retention of messages: 4 days, maximum of 14 days
  - Low latency (< 10 ms on publish and receive)
  - Limitation of 256KB per message sent
- Can have duplicate messages (at least once delivery, occasionally)
- Can have out of order messages (best effort ordering)

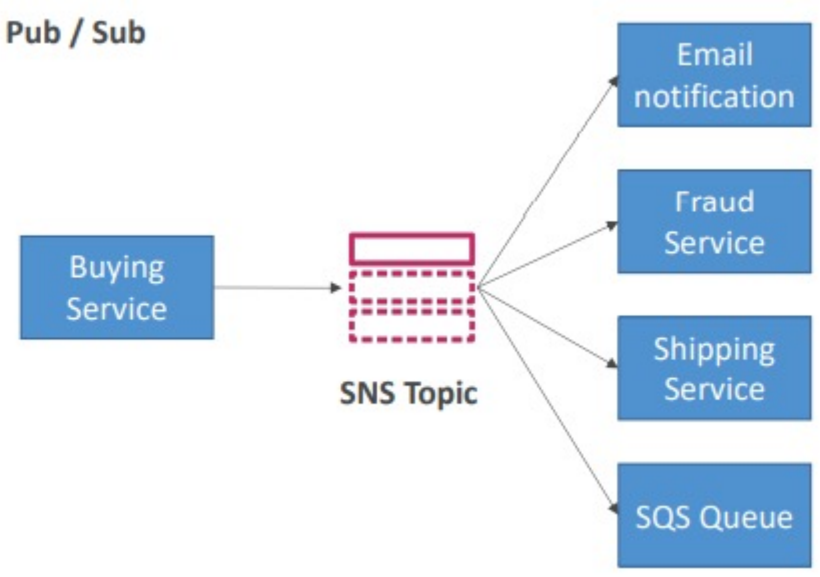


SQS - Security

- **Encryption:**
  - In-flight encryption using HTTPS API
  - At-rest encryption using KMS keys
  - Client-side encryption if the client wants to perform encryption/decryption itself
- **Access Controls**: IAM policies to regulate access to the SQS API
- **SQS Access Policies** (similar to S3 bucket policies)
  - Useful for cross-account access to SQS queues
  - Useful for allowing other services (SNS, S3…) to write to an SQS queue

SNS

- The “event producer” only sends message to one SNS topic
- As many “event receivers” (subscriptions) as we want to listen to the SNS topic notifications
- Each subscriber to the topic will get all the messages (note: new feature to filter messages)
- Up to 10,000,000 subscriptions per topic
- 100,000 topics limit
- Subscribers can be:
  - SQS
  - HTTP / HTTPS (with delivery retries – how many times)
  - Lambda
  - Emails
  - SMS messages
  - Mobile Notifications



Topic Publish (using the SDK)

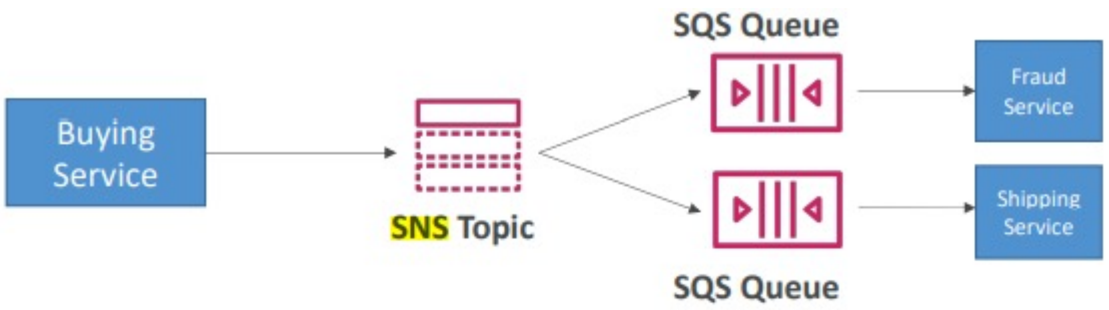
- Create a topic
- Create a subscription (or many)
- Publish to the topic

Direct Publish (for mobile apps SDK)

- Create a platform application
- Create a platform endpoint
- Publish to the platform endpoint
- Works with Google GCM, Apple APNS, Amazon ADM…

SNS – Security and Fan Out

- **Encryption:**
  - In-flight encryption using HTTPS API
  - At-rest encryption using KMS keys
  - Client-side encryption if the client wants to perform encryption/decryption itself
- **Access Controls:** IAM policies to regulate access to the SNS API
- **SNS Access Policies** (similar to S3 bucket policies)
  - Useful for cross-account access to SNS topics
  - Useful for allowing other services (S3…) to write to an SNS topic

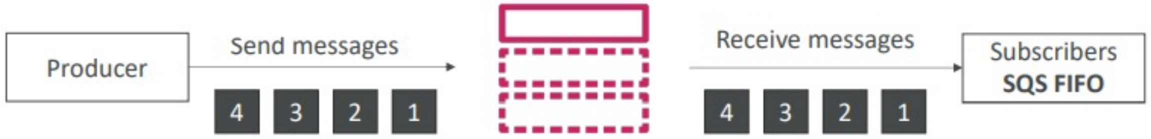


SNS + SQS: Fan Out

- Push once in SNS, receive in all SQS queues that are subscribers
- Fully decoupled, no data loss
- SQS allows for: data persistence, delayed processing and retries of work
- Ability to add more SQS subscribers over time
- Make sure your SQS queue **access policy** allows for SNS to write

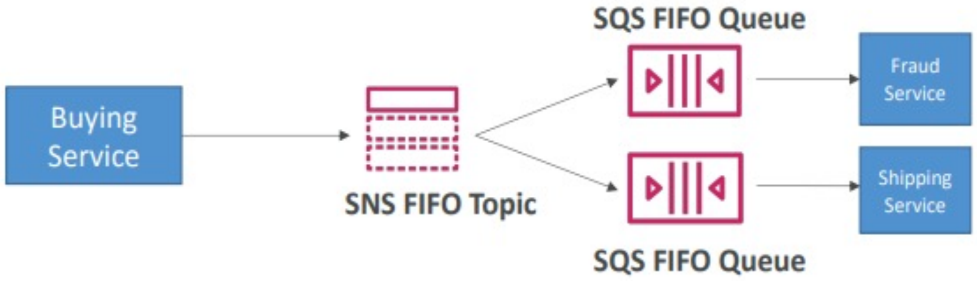
SNS – FIFO Topic and Fan Out

- FIFO = First In First Out (ordering of messages in the topic)
- Similar features as SQS FIFO:
  - Ordering by Message Group ID (all messages in the same group are ordered)
  - Deduplication using a Deduplication ID or Content Based Deduplication
- Can only have SQS FIFO queues as subscribers
- Limited throughput (same throughput as SQS FIFO)



SNS FIFO + SQS FIFO: Fan Out

- In case you need fan out + ordering + deduplication



SNS – Message Filtering

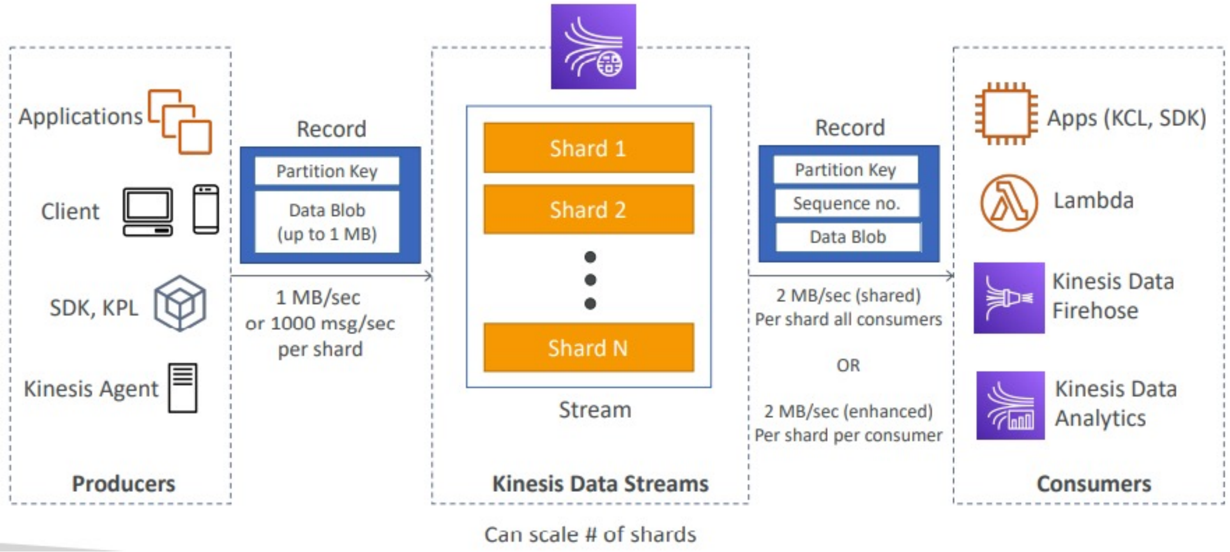
- JSON policy used to filter messages sent to SNS topic's subscriptions
- If a subscription doesn't have a filter policy, it receives every message

Kinesis

- Makes it easy to **collect, process,** and **analyze** streaming data in real-time
- Ingest real-time data such as: Application logs, Metrics, Website clickstreams, IoT telemetry data…
- **Kinesis Data Streams:** capture, process, and store data streams
- **Kinesis Data Firehose:** load data streams into AWS data stores
- **Kinesis Data Analytics:** analyze data streams with SQL or Apache Flink
- **Kinesis Video Streams:** capture, process, and store video streams

Kinesis Data Streams

- Billing is per shard provisioned, can have as many shards as you want
- Retention between 1 day (default) to 365 days
- Ability to reprocess (replay) data
- Once data is inserted in Kinesis, it can't be deleted (immutability)
- Data that shares the same partition goes to the same shard (ordering)
- Producers: AWS SDK, Kinesis Producer Library (KPL), Kinesis Agent
- Consumers:
  - Write your own: Kinesis Client Library (KCL), AWS SDK
  - Managed: AWS Lambda, Kinesis Data Firehose, Kinesis Data Analytics



Kinesis Data Firehose

- Fully Managed Service, no administration, automatic scaling, serverless
  - AWS: Redshift / Amazon S3 / ElasticSearch
  - 3rd party partner: Splunk / MongoDB / DataDog / NewRelic / …
  - Custom: send to any HTTP endpoint
- Pay for data going through Firehose
- **Near Real Time**
  - 60 seconds latency minimum for non full batches
  - Or minimum 32 MB of data at a time
- Supports many data formats, conversions, transformations, compression
- Supports custom data transformations using AWS Lambda
- Can send failed or all data to a backup S3 bucket

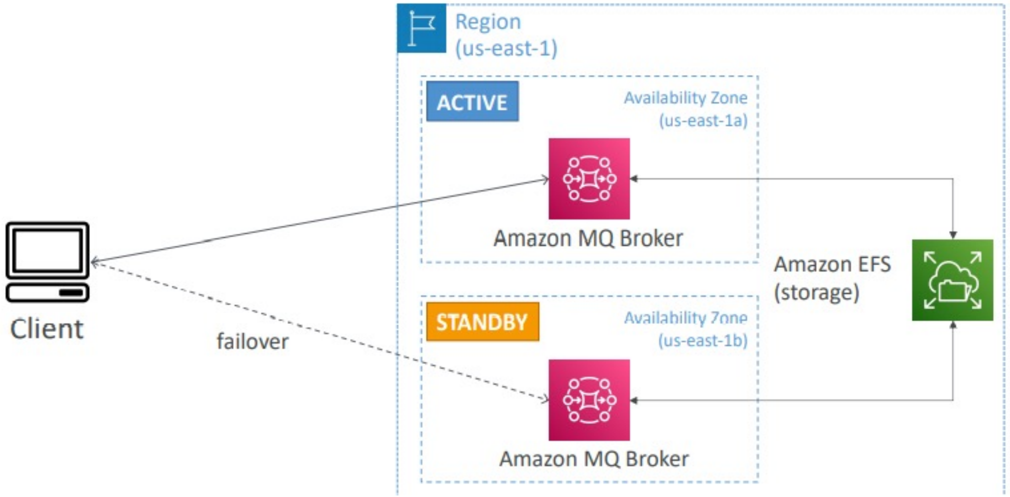
Kinesis Data Analytics (SQL application)

- Perform real-time analytics on Kinesis Streams using SQL
- Fully managed, no servers to provision
- Automatic scaling
- Real-time analytics
- Pay for actual consumption rate
- Can create streams out of the real-time queries
- Use cases:
  - Time-series analytics
  - Real-time dashboards
  - Real-time metrics

Amazon MQ

- SQS, SNS are “cloud-native” services, and they’re using proprietary protocols from AWS.
- Traditional applications running from on-premise may use open protocols such as: MQTT, AMQP, STOMP, Openwire, WSS
- **When migrating to the cloud**, instead of re-engineering the application to use SQS and SNS, we can use Amazon MQ
- **Amazon MQ = managed Apache ActiveMQ**
- Amazon MQ doesn’t “scale” as much as SQS / SNS
- Amazon MQ runs on a dedicated machine, can run in HA with failover
- Amazon MQ has both queue feature (~SQS) and topic features (~SNS)

MQ – High Availability



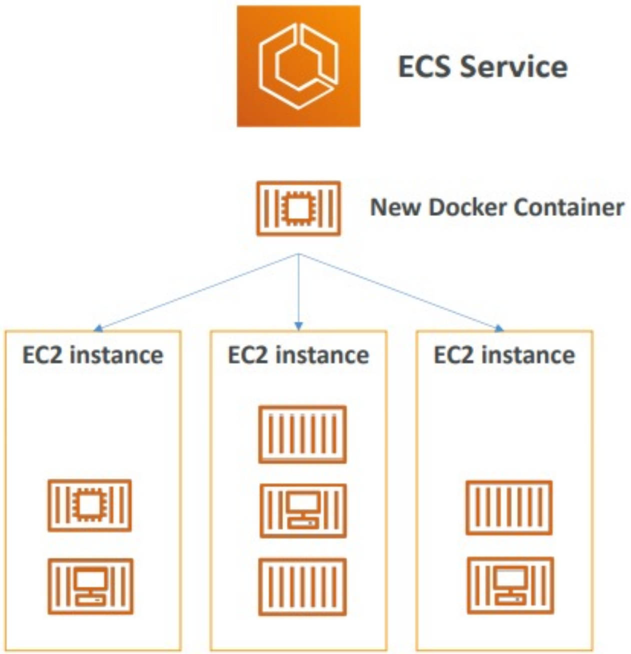
ECS, Fargate, EKS

- To manage containers, we need a container management platform
- **Three choices:**
- **ECS:** Amazon’s own container platform
- **Fargate:** Amazon’s own Serverless container platform
- **EKS:** Amazon’s managed Kubernetes (open source)

ECS

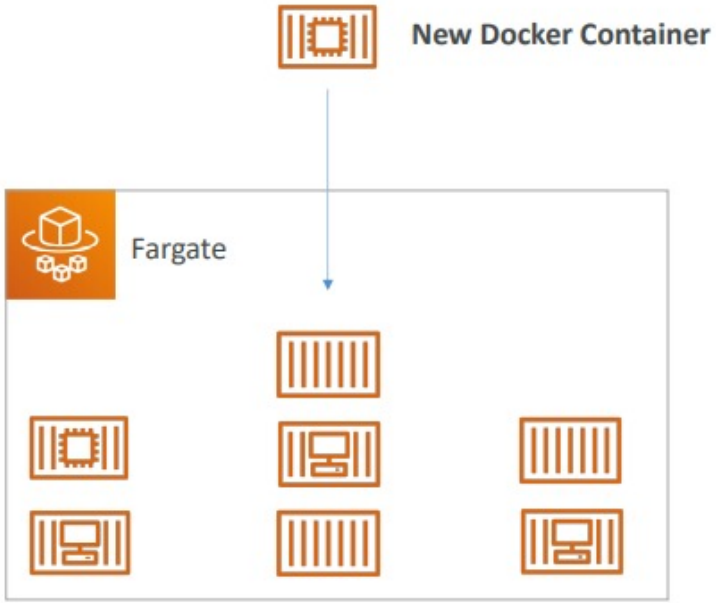
What is ECS?

- ECS = Elastic Container Service
- Launch Docker containers on AWS
- **You must provision & maintain the infrastructure (the EC2 instances)**
- AWS takes care of starting / stopping containers
- Has integrations with the Application Load Balancer

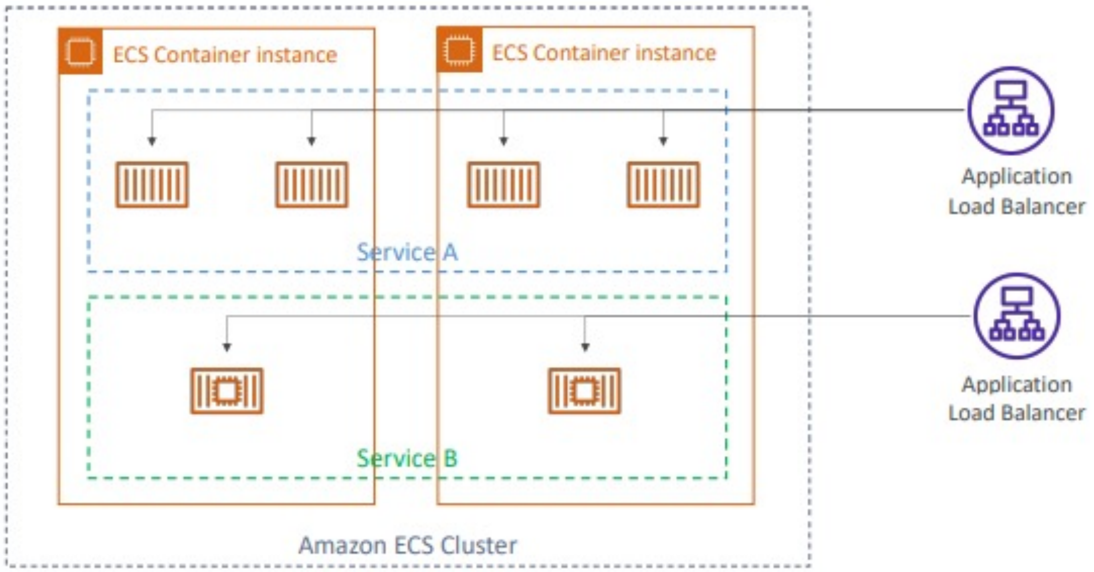


What is Fargate?

- Launch Docker containers on AWS
- **You do not provision the infrastructure (no EC2 instances to manage) – simpler!**
- **Serverless offering**
- AWS just runs containers for you based on the CPU / RAM you need



ECS Services/Tasks & Load Balancing



Load Balancing for EC2 Launch Type

- We get a **dynamic port** mapping
- The ALB supports finding the right port on your EC2 Instances
- **You must allow on the EC2 instance's security group any port from the ALB security group**

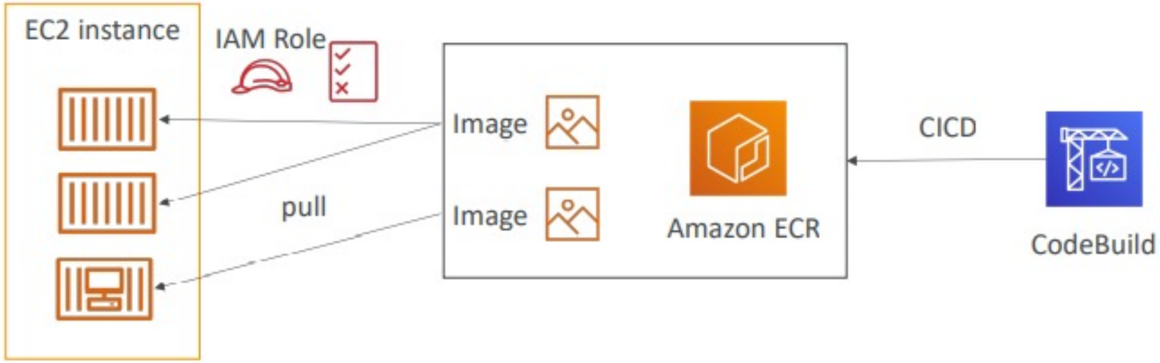
Load Balancing for Fargate

- Each task has a **unique IP**
- You must allow on the ENI's security group the task port from the ALB security group

ECR & EKS

ECR – Elastic Container Registry

- Store, manage and deploy containers on AWS, pay for what you use
- Fully integrated with ECS & IAM for security, backed by Amazon S3
- Supports image vulnerability scanning, version, tag, image lifecycle

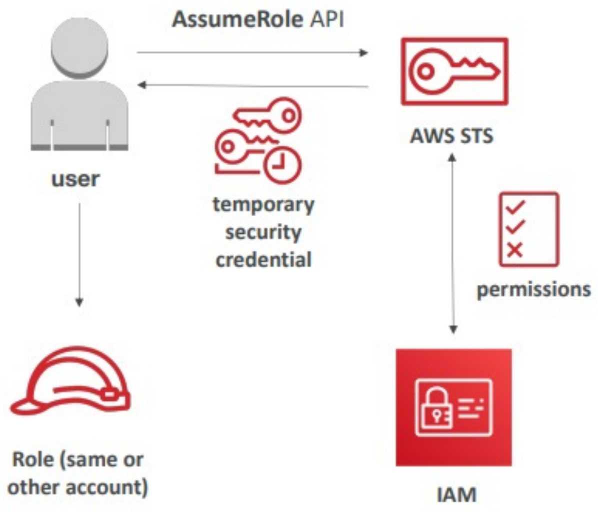


EKS – Elastic Kubernetes Service

- It is a way to launch **managed Kubernetes clusters on AWS**
- Kubernetes is an open-source system for automatic deployment, scaling and management of containerized (usually Docker) application
- It’s an alternative to ECS, similar goal but different API
- EKS supports **EC2** if you want to to deploy worker nodes or **Fargate** to deploy serverless containers
- **Use case:** if your company is already using Kubernetes on-premises or in another cloud, and wants to migrate to AWS using Kubernetes

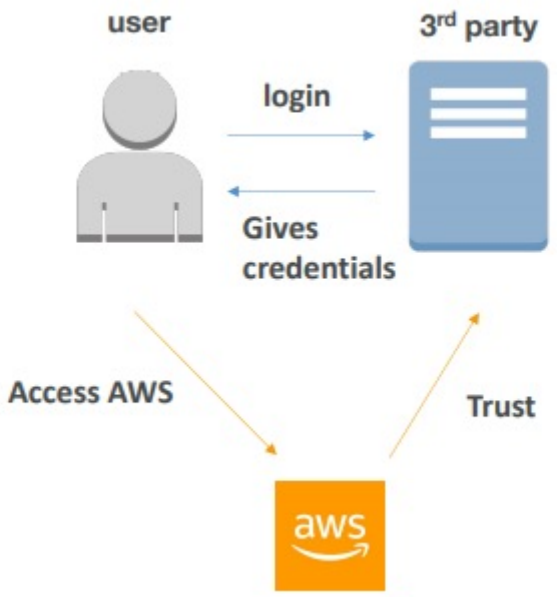
AWS STS – Security Token Service

- **Allows to grant limited and temporary access to AWS resources.**
- Token is valid for up to one hour (must be refreshed)
- **AssumeRole:**
  - Within your own account: for enhanced security
  - Cross Account Access: assume role in target account to perform actions there
- **AssumeRoleWithSAML:**
  - return credentials for users logged with SAML
- **AssumeRoleWithWebIdentity:**
  - return creds for users logged with an IdP (Facebook Login, Google Login, OIDC compatible…)
  - AWS recommends against using this, and using **Cognito** instead
- **GetSessionToken:**
  - for MFA, from a user or AWS account root user



Identity Federation in AWS

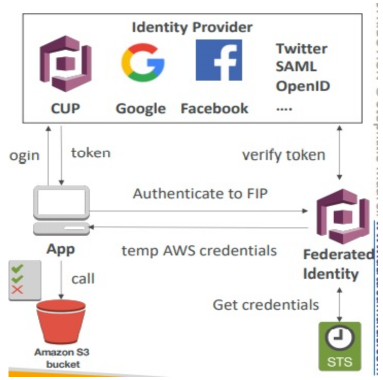
- Federation lets users outside of AWS to assume temporary role for accessing AWS resources.
- These users assume identity provided access role.
- Federations can have many flavors:
  - SAML 2.0
  - Custom Identity Broker
  - Web Identity Federation with Amazon Cognito
  - Web Identity Federation without Amazon Cognito
  - Single Sign On
  - Non-SAML with AWS Microsoft AD
- **Using federation, you don’t need to create IAM users (user management is outside of AWS)**



Cognito & Directory Services

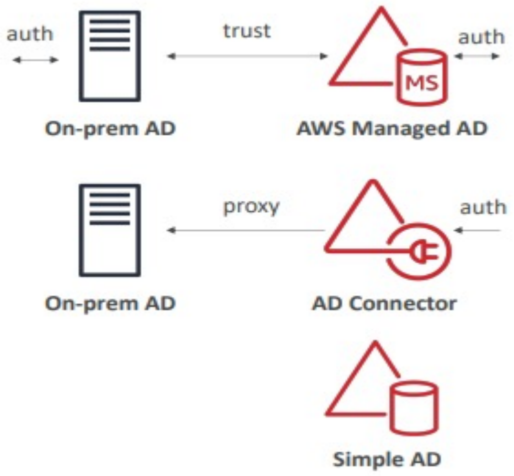
AWS Cognito

- **Goal:**
  - Provide direct access to AWS Resources from the Client Side (mobile, web app)
- **Example:**
  - provide (temporary) access to write to S3 bucket using Facebook Login
- **Problem:**
  - We don't want to create IAM users for our app users
- **How:**
  - Log in to federated identity provider – or remain anonymous
  - Get temporary AWS credentials back from the Federated Identity Pool
  - These credentials come with a pre-defined IAM policy stating their permissions



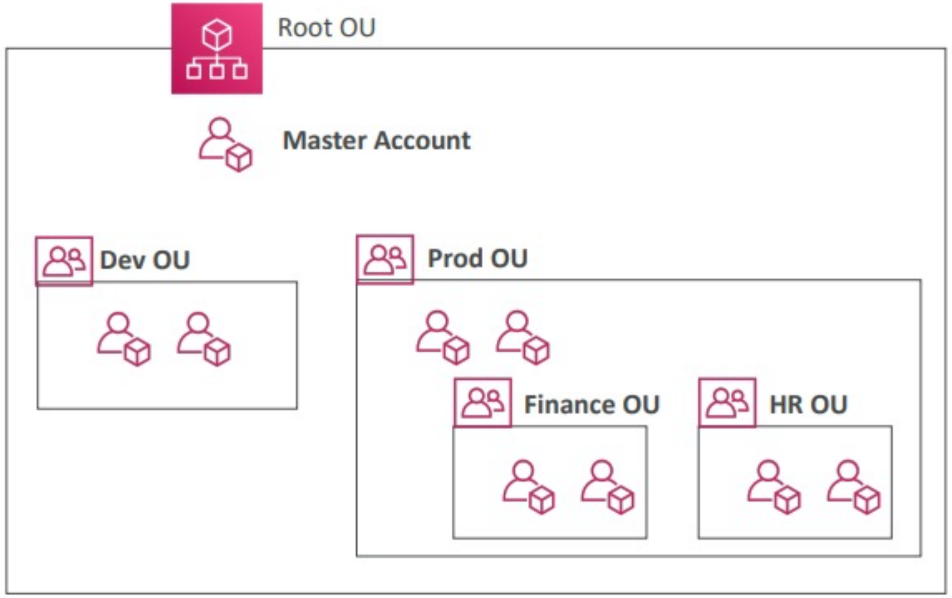
AWS Directory Services

- **AWS Managed Microsoft AD:**
  - Create your own AD in AWS, manage users locally, supports MFA
  - Establish "trust" connections with your on- premise AD
- **AD Connector:**
  - Directory Gateway (proxy) to redirect to on- premise AD
  - Users are managed on the on-premise AD
- **Simple AD:**
  - AD-compatible managed directory on AWS
  - Cannot be joined with on-premise AD



AWS Organizations

- Global service
- Allows to manage multiple AWS accounts
- The main account is the master account – you can't change it
- Other accounts are member accounts
- Member accounts can only be part of one organization
- Consolidated Billing across all accounts - single payment method
- Pricing benefits from aggregated usage (volume discount for EC2, S3…)
- API is available to automate AWS account creation

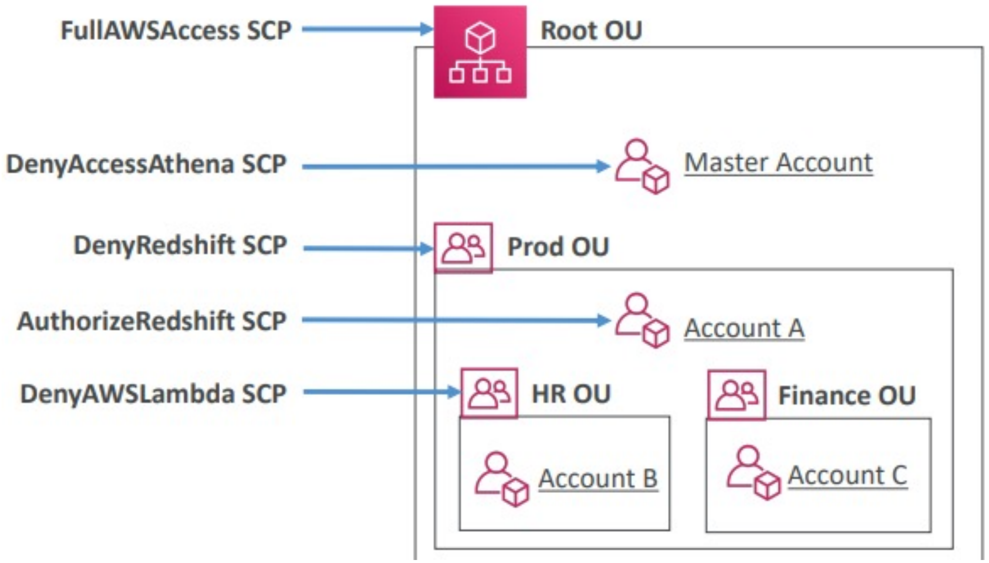


Multi Account Strategies

- Create accounts per department, per cost center, per dev/test/prod, based on regulatory restrictions (using SCP), for better resource isolation (ex: VPC), to have separate per-account service limits, isolated account for logging
- Multi Account vs One Account Multi VPC
- Use tagging standards for billing purposes
- Enable CloudTrail on all accounts, send logs to central S3 account
- Send CloudWatch Logs to central logging account

AWS Organization - SCP

- Whitelist or blacklist IAM actions
- Applied at the **OU** or **Account** level
- Does not apply to the Master Account
- SCP is applied to all the **Users and Roles** of the Account, including Root user
- The SCP does not affect service-linked roles
  - Service-linked roles enable other AWS services to integrate with AWS Organizations and can't be restricted by SCPs.
- SCP must have an explicit Allow (does not allow anything by default)
- Use cases:
  - Restrict access to certain services (for example: can’t use EMR)
  - Enforce PCI compliance by explicitly disabling services

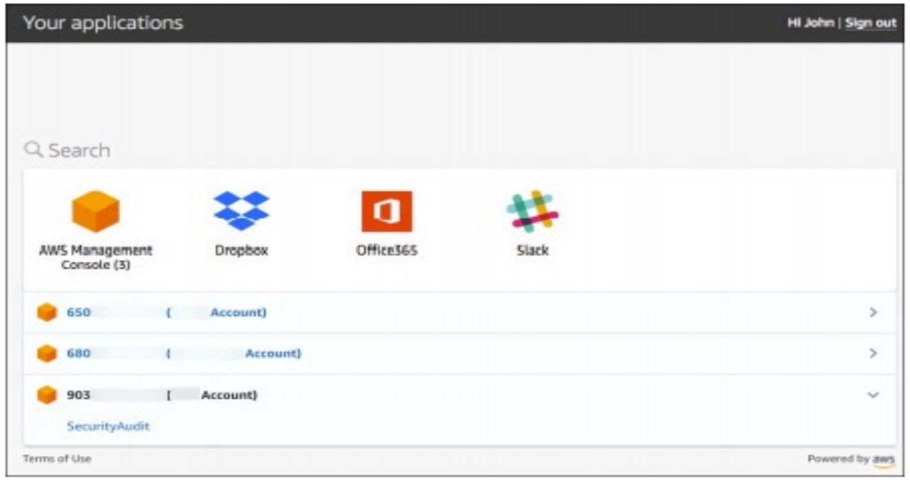


SCP – Blacklist Strategy

SCP – Whitelist Strategy

AWS Single Sign-On (SSO)

- Centrally manage Single Sign-On to access **multiple accounts** and **3rd-party business applications**.
- Integrated with **AWS Organizations**
- **Supports SAML 2.0** markup
- Integration with on-premise **Active Directory**
- Centralized permission management
- Centralized auditing with CloudTrail



Your applications

Hi John | Sign out

Search

AWS Management Console (3)

Dropbox

Office365

Slack

650 (Account)

680 (Account)

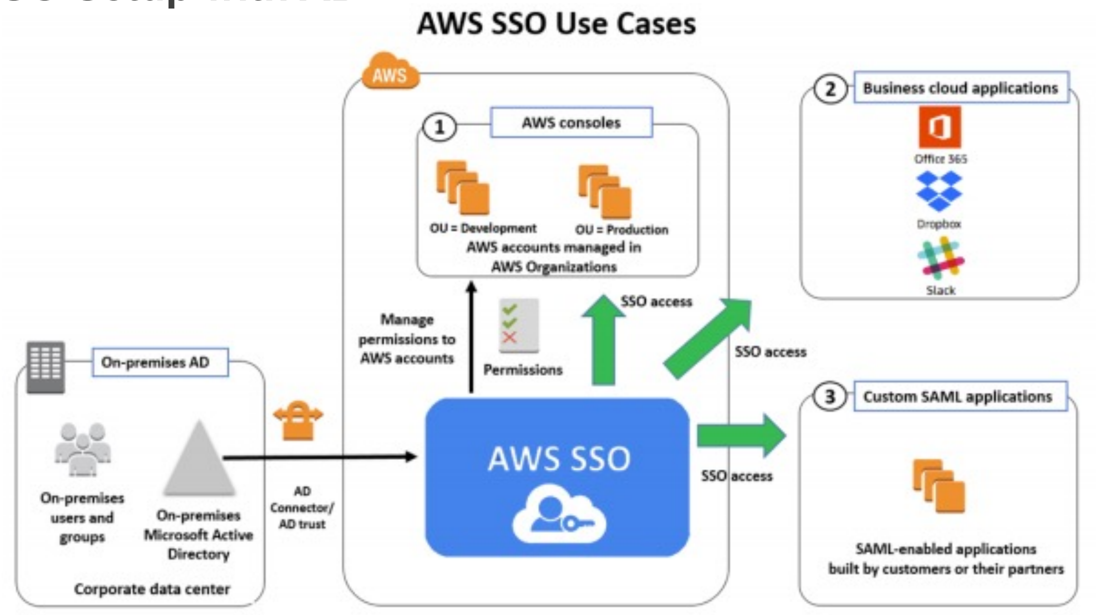
903 (Account)

SecurityAudit

Terms of Use

Powered by aws

SSO Setup with AD


AWS SSO Use Cases
AWS
1
AWS consoles
OU = Development
OU = Production
AWS accounts managed in AWS Organizations
Manage permissions to AWS accounts
Permissions
SSO access
SSO access
SSO access
2
Business cloud applications
Office 365
Dropbox
Slack
3
Custom SAML applications
SAML-enabled applications built by customers or their partners
AWS SSO
On-premises AD
On-premises users and groups
On-premises Microsoft Active Directory
Corporate data center
AD Connector/ AD trust

AWS Monitoring & Audit: CloudWatch

- CloudWatch provides metrics for every services in AWS
- **Metric** is a variable to monitor (CPUUtilization, NetworkIn...)
- Metrics belong to **namespaces**
- **Dimension** is an attribute of a metric (instance id, environment, etc...)
- Up to 10 dimensions per metric
- Metrics have **timestamps**

CloudWatch

CloudWatch Custom Metrics

- Possibility to define and send your own custom metrics to CloudWatch
- Example: memory (RAM) usage, disk space, number of logged in users ...
- Ability to use dimensions (attributes) to segment metrics
  - Instance.id
  - Environment.name

Metric resolution (**StorageResolution** API parameter – two possible value):

- Standard: 1 minute (60 seconds)
- High Resolution: 1/5/10/30 second(s) – Higher cost

Important: Accepts metric data points two weeks in the past and two hours in the future (make sure to configure your EC2 instance time correctly)

AWS Monitoring & Audit: CloudTrail

- **Provides governance, compliance and audit for your AWS Account**
- CloudTrail is enabled by default!
- Get **an history of events / API calls made within your AWS Account** by:
  - Console, SDK, CLI, AWS Services
- Can put logs from CloudTrail into CloudWatch Logs or S3
- **A trail can be applied to All Regions (default) or a single Region.**
- If a resource is deleted in AWS, investigate CloudTrail first!

CloudTrail Events

- **Management Events:**
  - Operations that are performed on resources in your AWS account
- **Data Events:**
  - By default, data events are not logged (because high volume operations)
- **CloudTrail Insights:**
  - Enable CloudTrail Insights to detect unusual activity in your account

CloudTrail Events Retention

- Events are stored for 90 days in CloudTrail
- To keep events beyond this period, log them to S3 and use Athena

CloudTrail

AWS Lambda

- Virtual functions – no servers to manage!
- Limited by time - short executions
- Run on-demand
- Scaling is automated!

Benefits of AWS Lambda

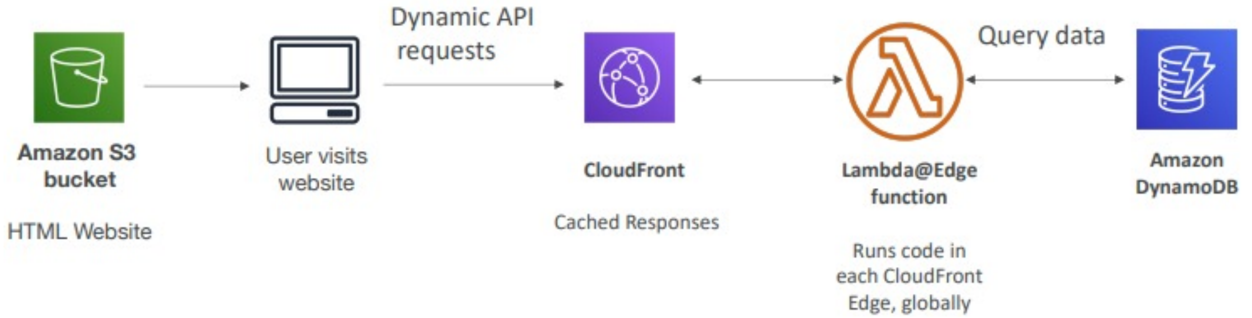
- Easy Pricing:
  - Pay per request and compute time
  - Free tier of 1,000,000 AWS Lambda requests and 400,000 GBs of compute time
- Integrated with the whole AWS suite of services
- Integrated with many programming languages
- Easy monitoring through AWS CloudWatch
- Easy to get more resources per functions (up to 10GB of RAM!)
- Increasing RAM will also improve CPU and network!

AWS Lambda language support

- Node.js (JavaScript), Python, Java (Java 8 compatible)
- C# (.NET Core), Golang
- C#, Powershell, Ruby, Custom Runtime API (community supported, example Rust)

Lambda@Edge

- You have deployed a CDN using CloudFront
- What if you wanted to run a global AWS Lambda alongside?
- Or how to implement request filtering before reaching your application?
- For this, you can use **Lambda@Edge**: deploy Lambda functions alongside your CloudFront CDN
  - Build more responsive applications
  - You don't manage servers, Lambda is deployed globally
  - Customize the CDN content
  - Pay only for what you use



Lambda@Edge: Use Cases

- Website Security and Privacy
- Dynamic Web Application at the Edge
- Search Engine Optimization (SEO)
- Intelligently Route Across Origins and Data Centers
- Bot Mitigation at the Edge
- Real-time Image Transformation
- User Authentication and Authorization

AWS DynamoDB

- Fully Managed, Highly available with replication across 3 AZ
- NoSQL database - not a relational database
- Scales to massive workloads, distributed database
- Millions of requests per seconds, trillions of row, 100s of TB of storage
- Fast and consistent in performance (low latency on retrieval)
- Integrated with IAM for security, authorization and administration
- Low cost and auto scaling capabilities

DynamoDB Basics:

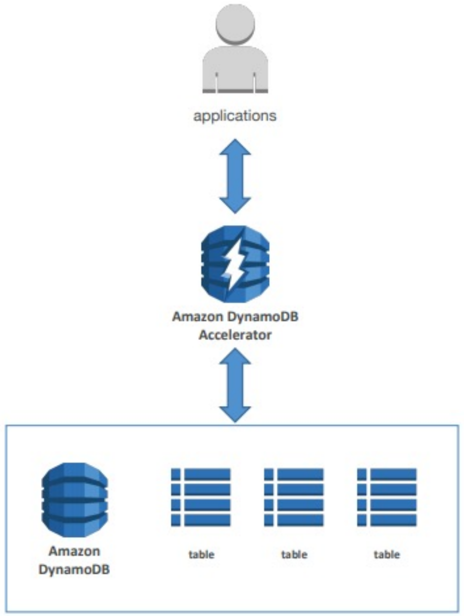
- DynamoDB is made of **tables**
- Each table has a **primary key** (must be decided at creation time)
- Each table can have an infinite number of items (= rows)
- Each item has **attributes** (can be added over time – can be null)
- Maximum size of an item is **400KB**
- Table must have provisioned read and write capacity units
- **Read Capacity Units (RCU) & Write Capacity Units (WCU)**

DynamoDB

DynamoDB – DAX & Streams

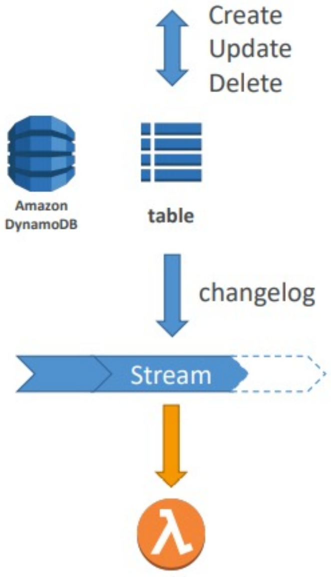
DAX - DynamoDB Accelerator

- Seamless cache for DynamoDB, no application re-write
- Writes go through DAX to DynamoDB
- Microsecond latency for cached reads & queries
- Solves the Hot Key problem (too many reads)
- 5 minutes TTL for cache by default
- Up to 10 nodes in the cluster
- Multi AZ (3 nodes minimum recommended for production)
- Secure (Encryption at rest with KMS,VPC, IAM, CloudTrail...)



DynamoDB Streams

- Changes in DynamoDB (Create, Update, Delete) can end up in a DynamoDB Stream
- This stream can be read by AWS Lambda, and we can then do:
  - React to changes in real time (welcome email to new users)
  - Analytics
  - Create derivative tables / views
  - Insert into ElasticSearch
- **Could implement cross region replication using Streams**
- Stream has 24 hours of data retention



AWS API Gateway

- AWS Lambda + API Gateway: No infrastructure to manage
- Support for the WebSocket Protocol
- Handle API versioning (v1, v2...)
- Handle different environments (dev, test, prod...)
- Handle security (Authentication and Authorization)
- Transform and validate requests and responses
- Generate SDK and API specifications
- Cache API responses

API Gateway – Integrations High Level

- **Lambda Function**
  - Invoke Lambda function
  - Easy way to expose REST API backed by AWS Lambda
- **HTTP**
  - Expose HTTP endpoints in the backend
  - Example: internal HTTP API on premise, Application Load Balancer...
  - Why? Add rate limiting, caching, user authentications, API keys, etc...

API Gateway

Topics we’ll cover in week - 4

- AWS Security & Encryption
- Networking – VPC
- Disaster Recovery & Migrations

Q & A

Thank you

CONNECT WITH US

EMAIL INFO@INFOSTRETCH.COM

CALL +1-408-727-1100

infostretch