Session 8

**CloudWatch**

Amazon CloudWatch is a monitoring service for AWS cloud resources and the applications you run on AWS.

You can use Amazon CloudWatch to collect and track metrics, collect and monitor log files, and set alarms. Amazon CloudWatch can monitor AWS resources such as Amazon EC2 instances, Amazon DynamoDB tables, and Amazon RDS DB instances, as well as custom metrics generated by your applications and services, and any log files your applications generate.

**Cloud Watch Logs**

Amazon CloudWatch Logs lets you monitor and troubleshoot your systems and applications using your existing system, application and custom log files.

With CloudWatch Logs, you can monitor your logs, in near real time, for specific phrases, values or patterns. For example, you could set an alarm on the number of errors that occur in your system logs or view graphs of latency of web requests from your application logs.

Log data can be stored and accessed indefinitely in highly durable, low-cost storage so you don’t have to worry about filling up hard drives.

What can I measure with Amazon CloudWatch Metrics?

Amazon CloudWatch allows you to monitor AWS cloud resources and the applications you run on AWS.

**Metrics are provided automatically for a number of AWS products and services, including Amazon EC2 instances, EBS volumes, Elastic Load Balancers, Auto Scaling groups, EMR job flows, RDS DB instances, DynamoDB tables, ElastiCache clusters, RedShift clusters, OpsWorks stacks, Route 53 health checks, SNS topics, SQS queues, SWF workflows, and Storage Gateways.**

You can also monitor custom metrics generated by your own applications and services.

**What types of CloudWatch Alarms can be created?**

You can create an alarm to monitor any Amazon CloudWatch metric in your account. For example, you can create alarms on an Amazon EC2 instance CPU utilization, Amazon ELB request latency, Amazon DynamoDB table throughput, Amazon SQS queue length, or even the charges on your AWS bill.

You can also create an alarm on custom metrics that are specific to your custom applications or infrastructure. If the custom metric is a high-resolution metric, you have the option of creating high-resolution alarms that alert as soon as 10-second or 30-second periods.

**What is AWS CloudTrail?**

**AWS CloudTrail is a web service that records activity made on your account and delivers log files to your Amazon S3 bucket**.

**CloudTrail provides visibility into user activity by recording actions taken on your account**. CloudTrail records important information about each action, including who made the request, the services used, the actions performed, parameters for the actions, and the response elements returned by the AWS service. This information helps you to track changes made to your AWS resources and to troubleshoot operational issues. CloudTrail makes it easier to ensure compliance with internal policies and regulatory standards

Cloud Watch – Demo

while true; do echo; done

Standard Monitoring – 5 minute

Detailed Monitoring – 1 minute

<https://aws.amazon.com/cloudwatch/pricing/>

Cloud Trail- Monitors API Calls, what or who is doing what to the instance.

Cloud Formation -Demo

AWS CloudFormation provisions your resources in a safe, repeatable manner, allowing you to build and rebuild your infrastructure and applications, without having to perform manual actions or write custom scripts. CloudFormation takes care of determining the right operations to perform when managing your stack, and rolls back changes automatically if errors are detected.

SQS --ppt

**SWF**

Amazon Simple Workflow Service (SWF) is a web service that makes it easy to coordinate work across distributed application components. Amazon 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.

Amazon SWF can be used to address many challenges that arise while building applications with distributed components. For example, you can use Amazon SWF and the accompanying AWS Flow Framework for:

* Writing your applications as asynchronous programs using simple programming constructs that abstract details such as initiating tasks to run remotely and tracking the program’s runtime state.
* Maintaining your application’s execution state (e.g. which steps have completed, which ones are running, etc.). You do not have to use databases, custom systems, or ad hoc solutions to keep execution state.
* Communicating and managing the flow of work between your application components. With Amazon SWF, you do not need to design a messaging protocol or worry about lost and duplicated tasks.
* Centralizing the coordination of steps in your application. Your coordination logic does not have to be scattered across different components, but can be encapsulated in a single program.
* Integrating a range of programs and components, including legacy systems and 3rd party cloud services, into your applications. By allowing your application flexibility in where and in what combination the application components are deployed, Amazon SWF helps you gradually migrate application components from private data centers to public cloud infrastructure without disrupting the application availability or performance.

**SNS**

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

**Amazon SNS follows the “**[**publish-subscribe” (pub-sub) messaging paradigm**](https://aws.amazon.com/pub-sub-messaging/)**, with notifications being delivered to clients using a “push” mechanism that eliminates the need to periodically check or “poll” for new information and updates**.

With simple APIs requiring minimal up-front development effort, no maintenance or management overhead and pay-as-you-go pricing, Amazon SNS gives developers an easy mechanism to incorporate a powerful notification system with their applications.

Amazon SNS offers several benefits making it a versatile option for building and integrating loosely-coupled, distributed applications:

* **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**

The Amazon SNS service can support a wide variety of needs including event notification, monitoring applications, workflow systems, time-sensitive information updates, mobile applications, and any other application that generates or consumes notifications.

Amazon SNS can be used in workflow systems to relay events among distributed computer applications, move data between data stores or update records in business systems. Event updates and notifications concerning validation, approval, inventory changes and shipment status are immediately delivered to relevant system components as well as end-users.

**A common pattern is to use SNS to publish messages to**[**Amazon SQS**](https://aws.amazon.com/sqs/)**message queues to reliably send messages to one or many system components asynchronously**.

Another example use for Amazon SNS is to relay time-critical events to mobile applications and devices. Since Amazon SNS is both highly reliable and scalable, it provides significant advantages to developers who build applications that rely on real-time events.

Amazon Simple Queue Service (SQS) and Amazon SNS are both messaging services within AWS, which provide different benefits for developers.

.Amazon SNS allows applications to send time-critical messages to multiple subscribers through a “push” mechanism, eliminating the need to periodically check or “poll” for updates. Amazon SQS is a [message queue service](https://aws.amazon.com/message-queue/) used by distributed applications to exchange messages through a polling model, and can be used to decouple sending and receiving components. Amazon SQS provides flexibility for distributed components of applications to send and receive messages without requiring each component to be concurrently available

A common pattern is to use SNS to publish messages to Amazon SQS queues to reliably send messages to one or many system components asynchronously.

Amazon MQ, Amazon SQS, and Amazon SNS are messaging services that are suitable for anyone from startups to enterprises.

If you're using messaging with existing applications, and want to move your messaging to the cloud quickly and easily, we recommend you consider [Amazon MQ](https://aws.amazon.com/amazon-mq/). It supports industry-standard APIs and protocols **so you can switch from any standards-based message broker to Amazon MQ without rewriting the messaging code in your applications**.

**If you are building brand new applications in the cloud, we recommend you consider Amazon SQS and Amazon SNS**. Amazon SQS and SNS are lightweight, fully managed message queue and topic services that scale almost infinitely and provide simple, easy-to-use APIs. You can use Amazon SQS and SNS to decouple and scale microservices, distributed systems, and serverless applications, and improve reliability.

**Elastic Transcoder**

Amazon Elastic Transcoder is media transcoding in [the cloud](https://aws.amazon.com/what-is-cloud-computing/). It is designed to be a highly scalable, easy to use and a cost effective way for developers and businesses to convert (or “transcode”) media files from their source format into versions that will playback on devices like smartphones, tablets and PCs.

You can use Amazon Elastic Transcoder to convert video and audio files into supported output formats optimized for playback on desktops, mobile devices, tablets, and televisions. In addition to supporting a wide range of input and output formats, resolutions, bitrates, and frame rates, Amazon Elastic Transcoder also offers features for automatic video bit rate optimization, generation of thumbnails, overlay of visual watermarks, caption support, DRM packaging, progressive downloads, encryption and more

**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), Amazon Elastic Container Service (Amazon ECS) or AWS Elastic Beanstalk, code running on AWS Lambda, or any web application.

Amazon API Gateway handles all of the tasks involved in accepting and processing up to hundreds of **thousands of concurrent API calls**, including **traffic management**, authorization and access control, monitoring, and API version management.

**Why use Amazon API Gateway?**

**Amazon API Gateway provides developers with a simple, flexible, fully managed, pay-as-you-go service that handles all aspects of creating and operating robust APIs for application back ends.**

**1) Metering. API Gateway helps you define plans that meter and restrict third-party developer access to your APIs.**

**2) Security. API Gateway provides you with multiple tools to authorize access to your APIs and control service operation access.**

**3) Resiliency. API Gateway helps you manage traffic with throttling so that backend operations can withstand traffic spikes.**

**4) Operations Monitoring. After an API is published and in use, API Gateway provides you with a metrics dashboard to monitor calls to your services.**

**5) Lifecycle Management. After an API has been published, you often need to build and test new versions that enhance or add new functionality. API Gateway lets you operate multiple API versions and multiple stages for each version simultaneously so that existing applications can continue to call previous versions after new API versions are published.**

6) Designed for Developers. API Gateway allows you to quickly create APIs and assign static content for their responses to reduce cross-team development effort and time-to-market for your applications. Teams who depend on your APIs can begin development while you build your backend processes.

7) Real-Time Two-Way Communication. Build real-time two-way communication applications such as chat apps, streaming dashboards, and notifications without having to run or manage any servers. API Gateway maintains a persistent connection between connected users and enables message transfer between them.