

Observabilidad de infraestructuras y aplicaciones con *AWS*

UPC - Cloud Computing Architecture

Índice - ¿Qué servicios vamos a ver?

- CloudWatch
 - Alarms
 - Logs
 - Metrics
- EventBridge
- CloudTrail
 - Athena
- X-Ray

Monitorización vs Observabilidad

- **Monitorización**

- Supervisión de métricas predefinidas con el objetivo de detectar errores en servicios específicos
- Debemos conocer las métricas clave de antemano

- **Observabilidad**

- Indica la capacidad que tenemos de sacar conclusiones sobre el estado de un sistema a partir de los datos que devuelve
- Nos ayuda a comprender mejor cómo funciona nuestro sistema
- Nos permite anticiparnos a problemas desconocidos
- Significado más global de sistema
 - Sistemas complejos
 - Arquitectura de microservicios (+ Servicios)

Datos a monitorizar

- Métricas

- Medidas periódicas de KPIs (Key Performance Indicators)
 - Uso de CPU
- Información agregable

- Logs

- Registros de eventos + Timestamp
- Ejemplos
 - Access Log de un servidor web

- Trazas

- Orientado a arquitecturas de microservicios
- Comportamiento de las peticiones
- Ejemplos
 - Latencias específicas

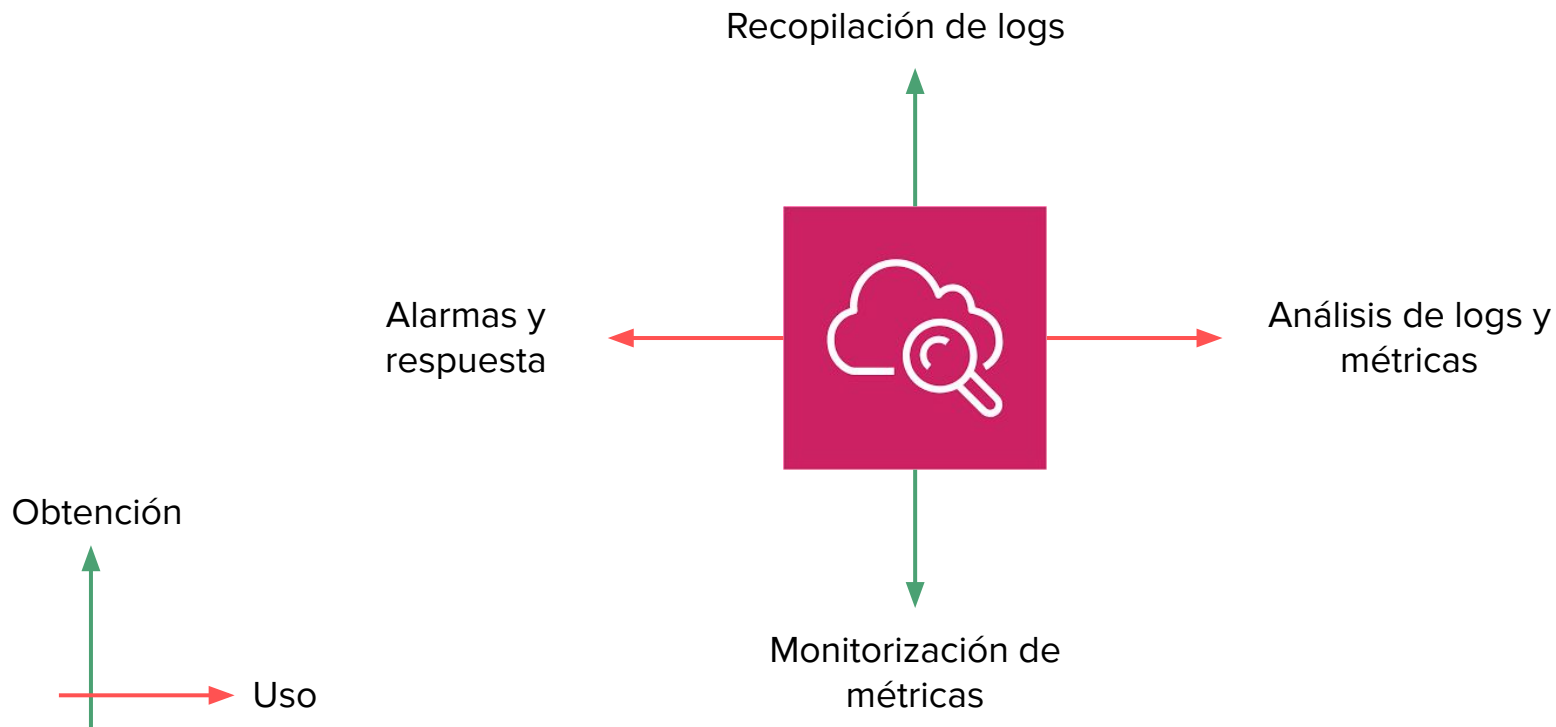
¿Qué métricas son importantes?

- Servicios centrados a usuarios (user-centric)
 - **Four Golden Signals**
 - **Latencia:** Tiempo que tarda en procesar una petición
 - **Tráfico:** ¿Cuánta demanda tiene el servicio? (req / s)
 - **Errores:** Ratio de peticiones que fallan (req / s)
 - **Saturación:** Carga del sistema (memoria en uso)
- Infraestructura
 - **USE**
 - **Utilización:** % CPU
 - **Saturación:** Trabajos (jobs) en cola
 - **Errores:** Número de eventos de error

CloudWatch

- Nos permite monitorizar recursos y aplicaciones en tiempo real
- Los servicios de *AWS* disponen de métricas propias
 - EC2 uso de CPU
 - RDS espacio libre de almacenamiento / IOPS
- Casos de uso:
 - Controlar recursos y rendimiento de aplicaciones (métricas)
 - Agrupar y monitorizar ficheros de logs
 - Crear y disparar alarmas
 - Basadas en métricas y logs

CloudWatch



CloudWatch

- **Conceptos clave:**

- Métricas
 - Estándar
 - Personalizadas
- Alarmas
 - Notificaciones

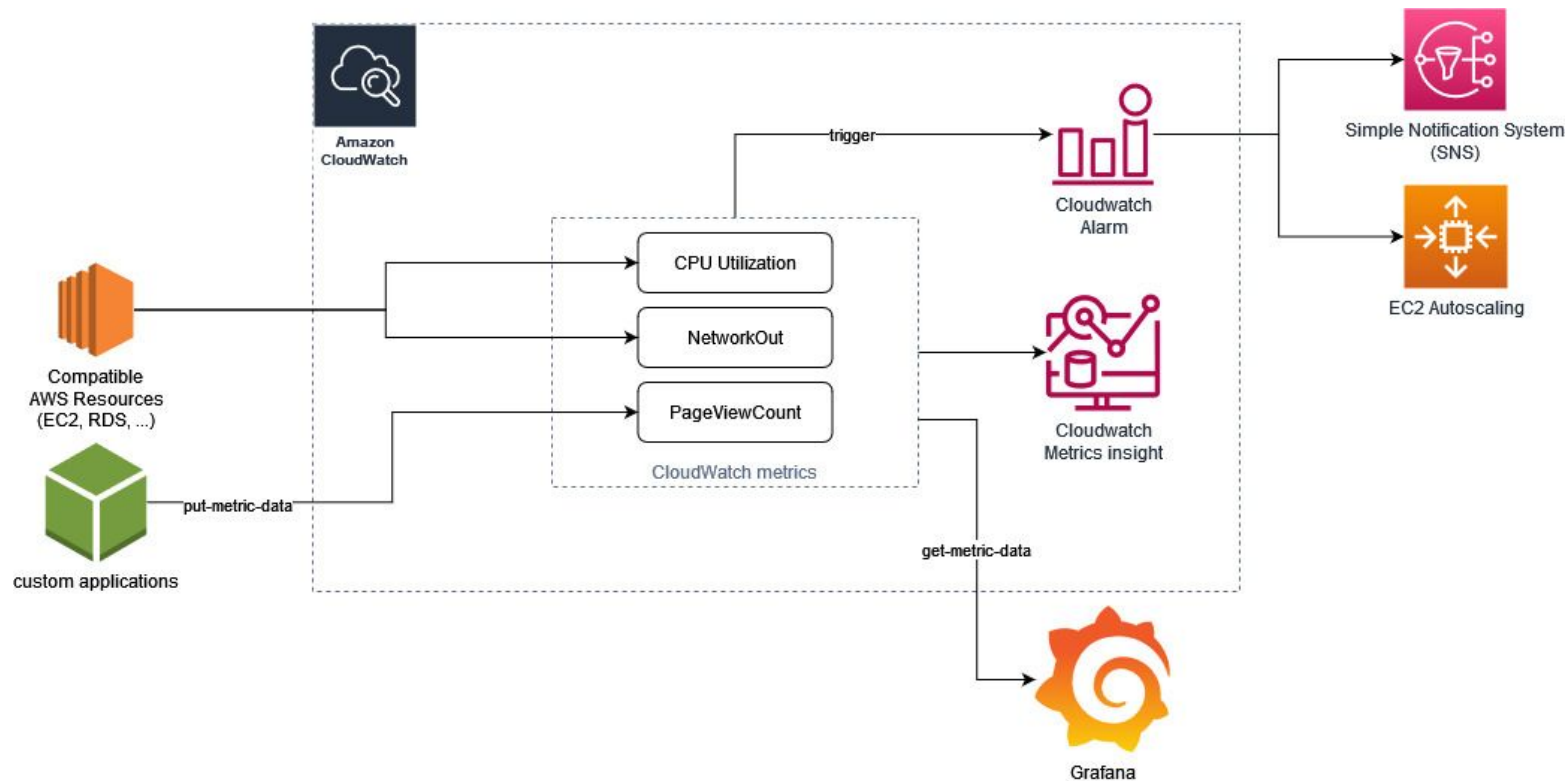
- **Limitaciones**

- CloudWatch por sí solo, no monitoriza a nivel de SO
 - Solo recolecta métricas estándar expuestas por los servicios (EC2, RDS, S3...)

- **CloudWatch agent:** recopila métricas a nivel de sistema

- Espacio de disco disponible
- Métricas custom
- etc...

CloudWatch - Ejemplo - Caso de uso



CloudWatch - Métricas

Metrica: Clave + Valor + Timestamp

Namespace: Grupo de métricas relacionadas

Dimensión: Par clave valor que categoriza la métrica

MetricName + Dimension = Nueva metrica!

```
1 {
2   "Metrics": [
3     {
4       "Namespace": "AWS/S3",
5       "MetricName": "BucketSizeBytes",
6       "Dimensions": [
7         {
8           "Name": "StorageType",
9           "Value": "StandardStorage"
10        },
11        {
12          "Name": "BucketName",
13          "Value": "mySuperBucket"
14        }
15      ]
16    }
17  ]
18 }
```

\$ aws cloudwatch list-metrics --namespace "AWS/S3"

CloudWatch - Metrics - Resolución

- **Estándar:** 5 minutos
- **Detallada:** 1 minuto
- **Alta resolución personalizada:** < 1 minuto

CloudWatch - Metrics - Retención de datos

A medida que pasa el tiempo, las métricas van perdiendo resolución, agregando los datos para ser almacenados:

- Periodo de **< 60 segundos** (métricas custom de alta resolución)
 - Durante **3 horas**
- Periodo de **60 segundos**
 - Durante **15 días**
- Periodo de **300 segundos (5 min)**
 - Durante **63 días (2 meses)**
- Periodo de **3600 segundos (1 hora)**
 - Durante **455 días (15 meses)**

Importante! Las métricas que no han tenido ninguna actualización no aparecen en la consola



get-metric-data / get-metric-statistics

CloudWatch - Metrics - Precio

- **Capa gratuita:**

- Todas las Métricas estándar (frecuencia de 5 minutos)
- 10 métricas de monitoreo detallado (frecuencia de 1 minuto)
- 1 millón de solicitudes API (no aplicable a GetMetricData ni GetMetricWidgetImage)

Capas	Costo (métrica/mes)
Primeras 10 000 métricas	0,30 USD
Siguientes 240 000 métricas	0,10 USD
Siguientes 750 000 métricas	0,05 USD
Más de 1 000 000 de métricas	0,02 USD

API

GetMetricData, GetInsightRuleReport	0,01 USD por 1000 métricas solicitadas
GetMetricWidgetImage	0,02 USD por 1000 métricas solicitadas
Solicitudes de GetMetricStatistics, ListMetrics, PutMetricData, GetDashboard, ListDashboards, PutDashboard y DeleteDashboards	0,01 USD por cada 1000 solicitudes

CloudWatch - Metrics - Precio

- **Ejemplo 1 - Monitorización detallada:**

- Suponemos 10 VM EC2
- Cada VM EC2 dispone de 7 métricas
- Métricas totales = $7 * 10 = 70$
- $0,3 \text{ USD} * 70 = \mathbf{21 \text{ USD al mes!}}$

CloudWatch - Metrics - Precio

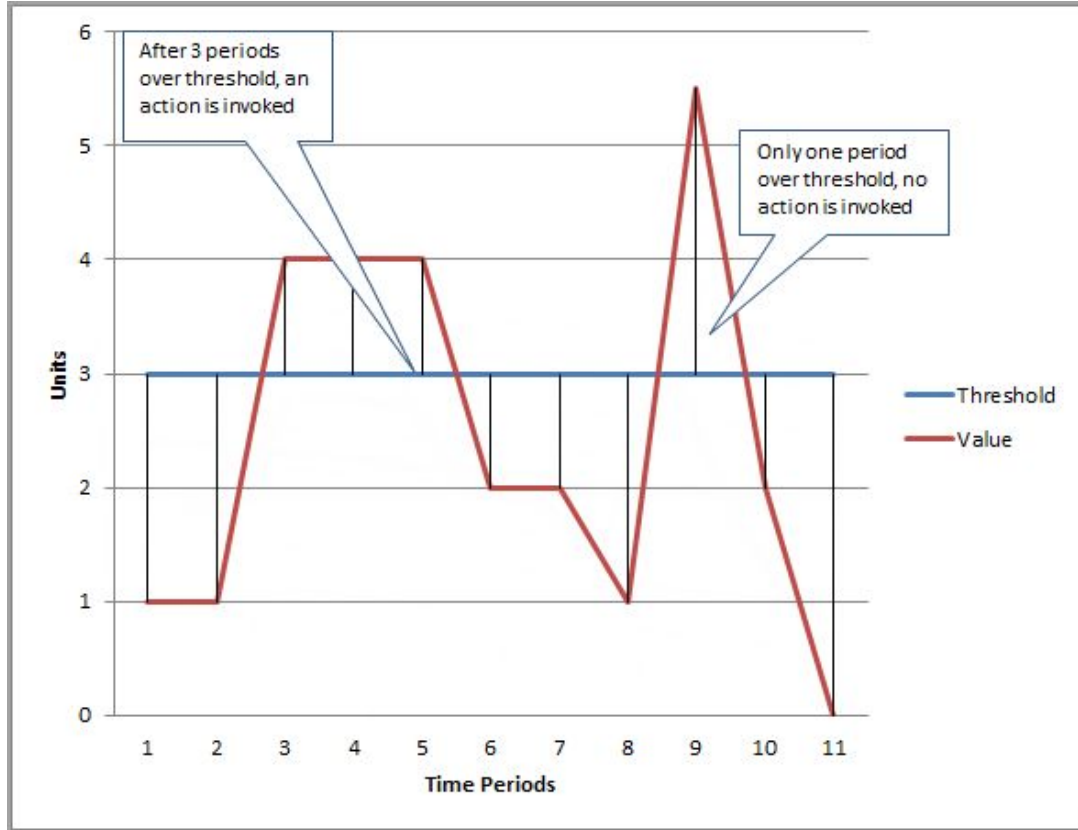
- **Ejemplo 2 - Métricas personalizadas:**

- Suponemos 100 VM EC2
- Publican 5 métricas personalizadas con el Agente CloudWatch cada minuto
- **Métricas**
 - $5 * 100 = 500$ métricas
 - $500 * 0,3 \text{ USD} = \mathbf{150\text{USD}}$
- **Peticiones**
 - $100 * (43200 \text{ minutos} / 1 \text{ minuto}) = \mathbf{4.320.000 \text{ peticiones al mes}}$
 - 1.000.000 entran dentro de la capa gratuita
 - $3.320.000 / 1000 * 0.01 = \mathbf{33 \text{ USD}}$
- **Total = 33 + 150 USD = 183 USD**

CloudWatch - Alarms

- 3 estados:
 - OK
 - No se ha excedido el umbral
 - Alarma
 - Se ha excedido el umbral
 - Datos insuficientes
 - La alarma se acaba de crear, o aún no hay datos para hacer una evaluación

Cloudwatch - Alarms - Ejemplo



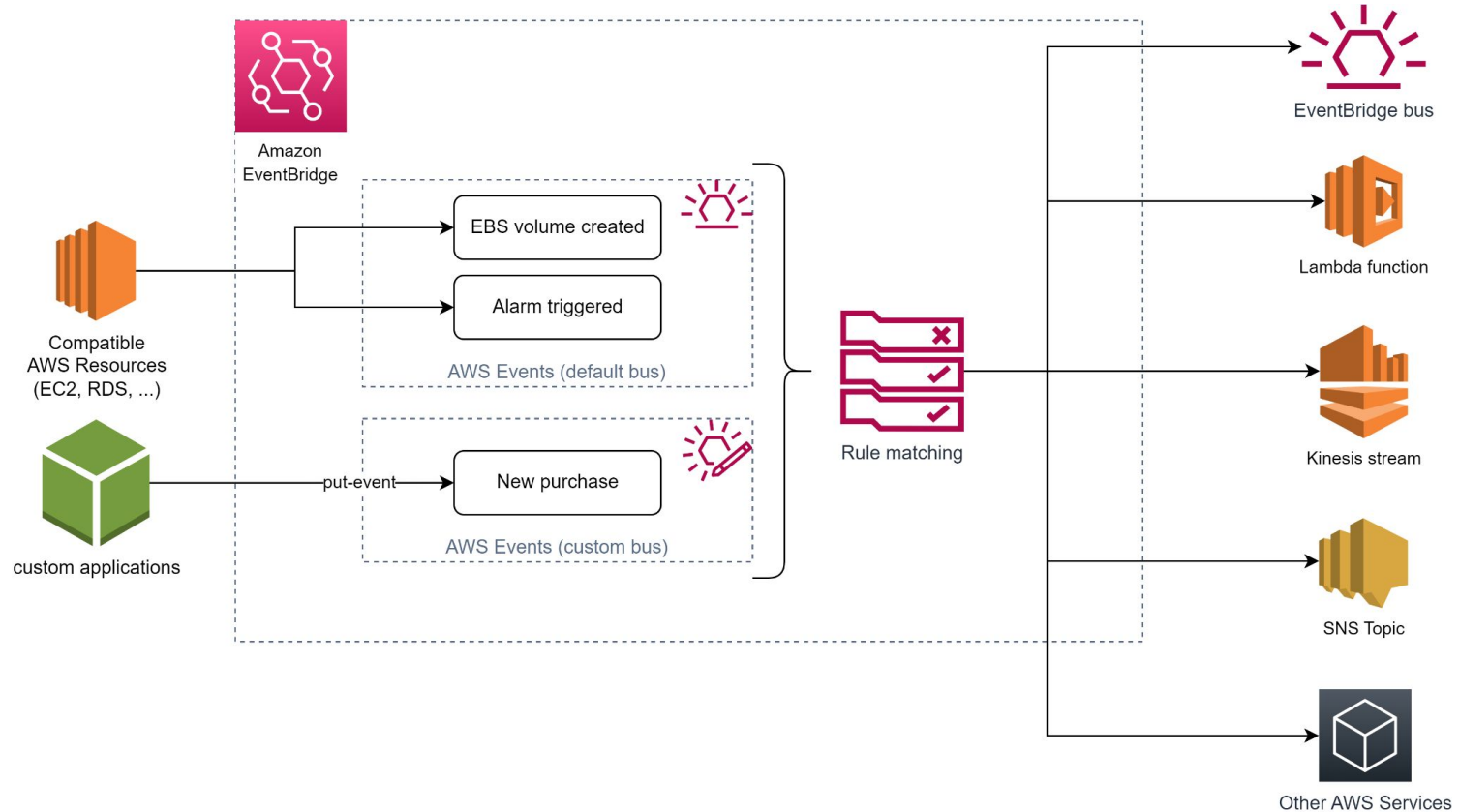
- **Evaluation period**
(Periodo de evaluacion)
 - Ejemplo: 3
- **Datapoints to Alarm**
(Puntos de datos para la alarma)
 - Ejemplo: 3

Events (EventBridge)

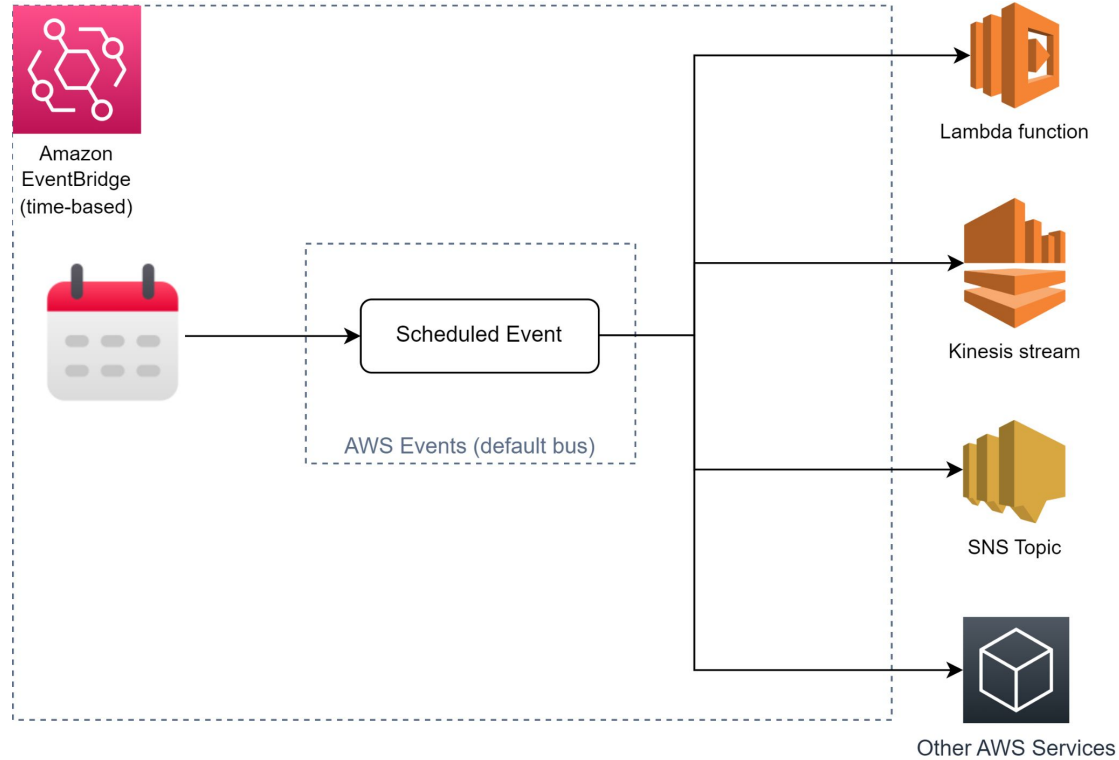


- Capturar eventos que ocurran en nuestra cuenta AWS y actuar acorde:
 - Creación de un volumen EBS
 - Activación de una alarma
- **EventBridge** es un bus de eventos sin servidor que facilita la creación de aplicaciones basadas en eventos

Events (EventBridge) - Ejemplo



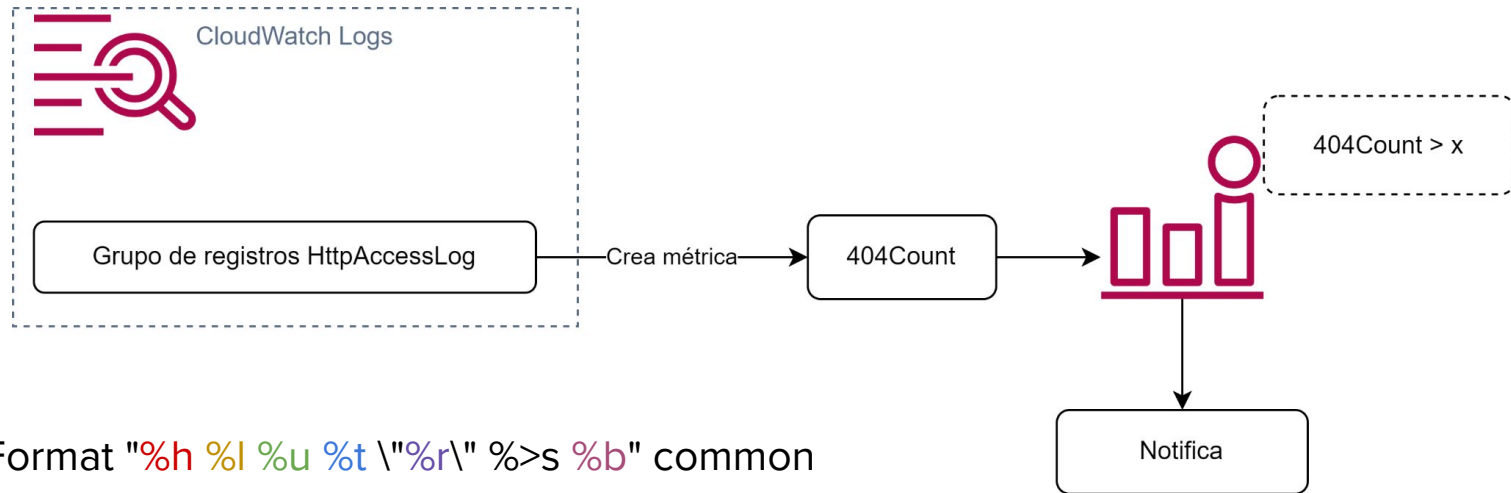
Events (EventBridge) - Ejemplo CRON



CloudWatch Logs

- CloudWatch logs nos permite monitorizar y almacenar archivos de registro de:
 - Instancias EC2
 - Servicios de AWS
 - AWS Route 53 / AWS Lambda / AWS CloudTrail
 - Otros
- Funcionalidades
 - Recopilación automática de registros
 - Agregación de datos en **grupos de registros**
 - Capacidad de configurar **filtros métricos**
 - Búsqueda de patrones (Errores en un Access Log de Apache)
 - Crear métricas en base a registros
 - Consulta de registros y creación de visualizaciones con **CloudWatch Logs Insights**

CloudWatch - Logs - Creación de métricas a partir de logs



LogFormat "%h %l %u %t \"%r\" %>s %b" common

127.0.0.1 - Marc [01/Feb/2023:13:55:36 +0100] "GET /hello_world HTTP/1.0" 200 2326

CloudWatch - Logs - Precio

- **Capa gratuita:**
 - 5GB (recopilación, almacenamiento, análisis...)

Recopilación (captura de datos)	0,50 USD por GB
Almacenamiento (archivado)	0,03 USD por GB
Análisis (consultas de Logs Insights)	0,005 USD por GB de datos escaneados
Detección y máscara (Data Protection)	0,12 USD por GB de datos analizados

CloudWatch - Logs - Precio

- **Ejemplo 1 - Logs HTTP:**

- Suponemos que registramos 1 GB diario durante 30 días = 30 GB al mes
- Captura de datos
 - 0 a 5GB = 0 (Capa gratuita)
 - 5 a 30 GB = $0,5 \text{ USD} * 25 = \mathbf{12,50 \text{ USD}}$
- Almacenamiento
 - 0 a 5GB = 0 (Capa gratuita)
 - 5 a 30 GB = $0,03 \text{ USD} * 25 = \mathbf{0,8 \text{ USD}}$
- Total
 - $\mathbf{12,50 + 0,8 = 13,3 \text{ USD}}$

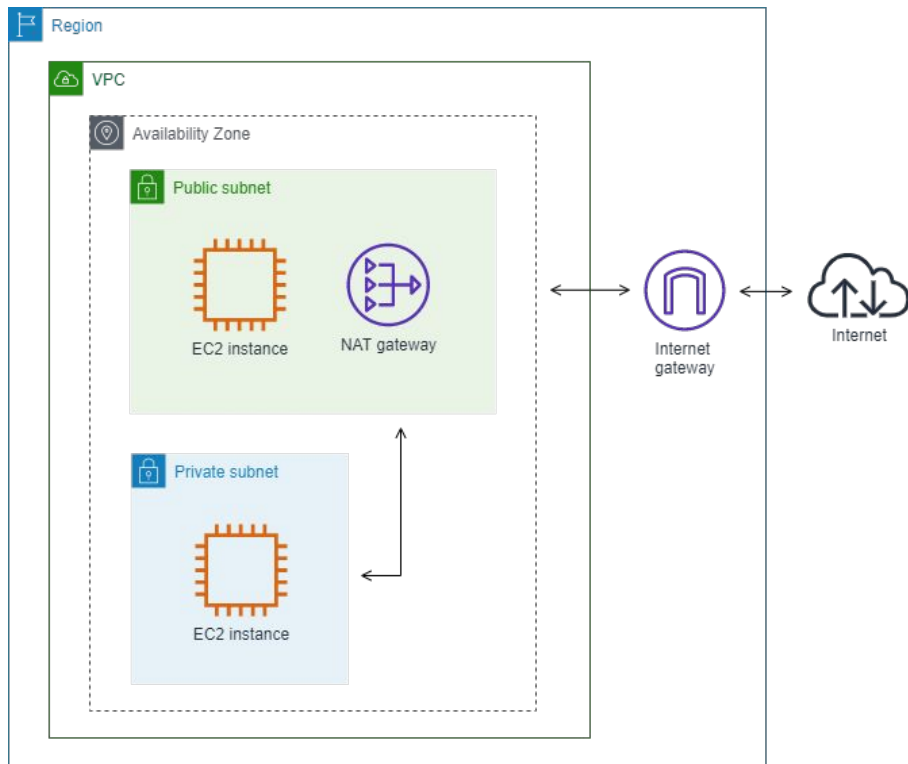
VPC Flow Logs

- Nos permite capturar información referente al **tráfico IP** que fluye a través de una o más interfaces de red
- Esta información puede distribuirse a
 - Bucket S3
 - Athena
 - CloudWatch Logs
 - Amazon Kinesis Data Firehose

VPC Flow Logs - Ejemplo

- Tráfico a través de un NAT Gateway

```
- eni-X 10.0.1.5 10.0.0.220 10.0.1.5 203.0.113.5
```



```
instance-id interface-id srcaddr dstaddr pkt-srcaddr pkt-dstaddr
```

VPC Flow Logs - Precio

	Envío a CloudWatch Logs	Envío a S3	Envío a Kinesis Data Firehose	
Datos recibidos				
Primeros 10 TB	0,50 USD por GB	0,25 USD por GB	0,25 USD por GB	
Siguientes 20 TB	0,25 USD por GB	0,15 USD por GB	0,15 USD por GB	
Siguientes 20 TB	0,10 USD por GB	0,075 USD por GB	0,075 USD por GB	
Más de 50 TB	0,05 USD por GB	0,05 USD por GB	0,05 USD por GB	
Datos almacenados	0,03 USD por GB	A partir de 0,023 USD/GB (Standard) hasta 0,00099 USD/GB (Glacier Deep Archive)		No se aplica
Formato convertido a Apache Parquet	N/A	0,035 USD por GB*	N/A	

Cloudwatch > Pricing > Logs > Vended Logs

Quizz!



Associate Question - 1

- 6) A company runs several production workloads on Amazon EC2 instances. A SysOps administrator discovered that a production EC2 instance failed a system health check. The SysOps administrator recovered the instance manually.

The SysOps administrator wants to automate the recovery task of EC2 instances and receive notifications whenever a system health check fails. Detailed monitoring is activated for all of the company's production EC2 instances.

Which of the following is the MOST operationally efficient solution that meets these requirements?

- A. For each production EC2 instance, create an Amazon CloudWatch alarm for Status Check Failed: System. Set the alarm action to recover the EC2 instance. Configure the alarm notification to be published to an Amazon Simple Notification Service (Amazon SNS) topic.
- B. On each production EC2 instance, create a script that monitors the system health by sending a heartbeat notification every minute to a central monitoring server. If an EC2 instance fails to send a heartbeat, run a script on the monitoring server to stop and start the EC2 instance and to publish a notification to an Amazon Simple Notification Service (Amazon SNS) topic.
- C. On each production EC2 instance, create a script that sends network pings to a highly available endpoint by way of a cron job. If the script detects a network response timeout, invoke a command to reboot the EC2 instance.
- D. On each production EC2 instance, configure an Amazon CloudWatch agent to collect and send logs to a log group in Amazon CloudWatch Logs. Create a CloudWatch alarm that is based on a metric filter that tracks errors. Configure the alarm to invoke an AWS Lambda function to reboot the EC2 instance and send a notification email.

Associate Question - 1

- EC2 Status checks
 - System
 - Comprueba el hardware (sistema de AWS) sobre el que corre nuestra instancia
 - Requiere de la intervención de AWS
 - Instance
 - Comprueba el software y la configuración de red de la instancia
 - El problema es nuestro

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Avail
<input checked="" type="checkbox"/>	-	i-0c0186a12aab3741d	Running	t2.large	1/2 checks ...	No alarms +	eu-w
<input type="checkbox"/>	-	i-0138edcaf722db475	Running	m4.large	2/2 checks ...	No alarms +	eu-w
<input type="checkbox"/>	-	i-02c65b735153975ec	Running	t3.medium	2/2 checks ...	No alarms +	eu-w

Instance: i-0c0186a12aab3741d

Details | Security | Networking | Storage | **Status checks** | Monitoring | Tags

Status checks [Info](#)

Status checks detect problems that may impair i-0c0186a12aab3741d from running your applications.

System status checks

✔ System reachability check passed

Instance status checks

✘ Instance reachability check failed

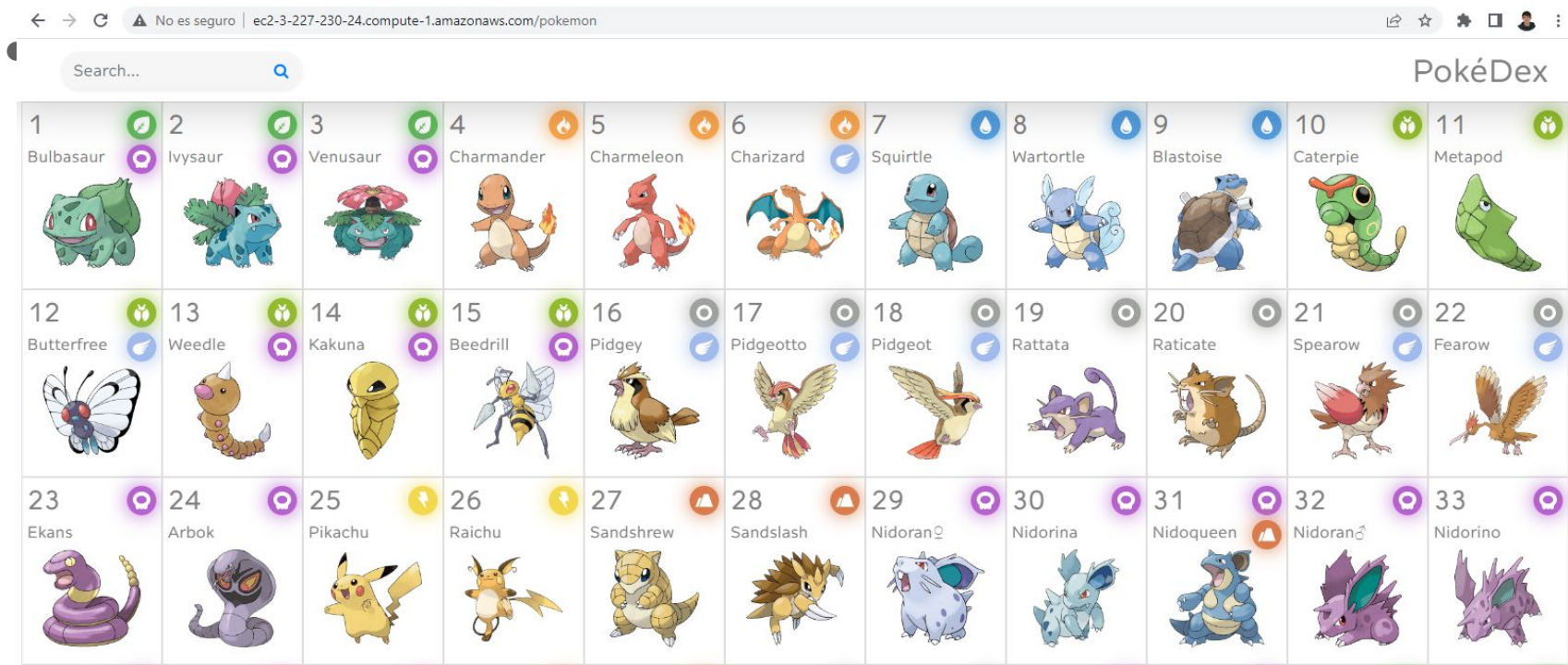
Check failure at

2020/12/16 17:30 GMT+2 (about 1 month)

Lab 1 - Pokedex

- Pokemon Go ha vuelto a hacerse viral
- El CEO ha decidido publicar una aplicación Web que ofrece el servicio de Pokedex
 - Objetivo: Conseguir aumentar la presencia de la marca en los círculos de jugadores
- La aplicación parece que está gustando aunque algunas personas se quejan de errores puntuales
- Problema! No la estamos monitorizando! ¿Errores?

Lab 1 - Pokedex




Lab 1 - La Pokedex

← → ↻ ⚠ No es seguro | ec2-3-227-230-24.compute-1.amazonaws.com/pokemon/94

← PokéDex

GENGAR

Shadow Pokémon



EVOLUTION CHAIN

ID #94

Height 1.5m (4'11")

Weight 40.5kg (89.3lbs.)

Abilities CURSED-BODY

Type Ghost Poison

Forms GENGAR MEGA GENGAR GIGANTAMAX GENGAR

Base Min Max

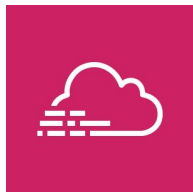
HP	60	
Attack	65	
Defence	60	
Sp. Attack		130
Sp. Defence	75	
Speed		110
Total	500	

Lab 1 - La Pokedex

- Despliegue con **Terraform**
- Servidor NGINX
 - App Web Angular



AWS CloudTrail



- Registra, monitoriza en tiempo real y retiene la actividad de la cuenta de AWS
 - Registra las llamadas a la API para la mayoría de los servicios de AWS
- Envía de manera automática los registros a Amazon S3
- CloudTrail nos permite dar respuesta a preguntas del estilo:
 - ¿Quién ha ejecutado esa instancia?
 - ¿Quién ha cambiado la configuración de un Security Group?
 - ¿Hay alguna actividad que provenga de una IP desconocida?
- Permite activar un mecanismo de validación de la integridad de los logs
- Importante:
 - No realiza el seguimiento de eventos dentro de una instancia de EC2, solo monitoriza la interacción con la API.

AWS CloudTrail - Log de ejemplo

Usuario que ha generado la petición

Información adicional

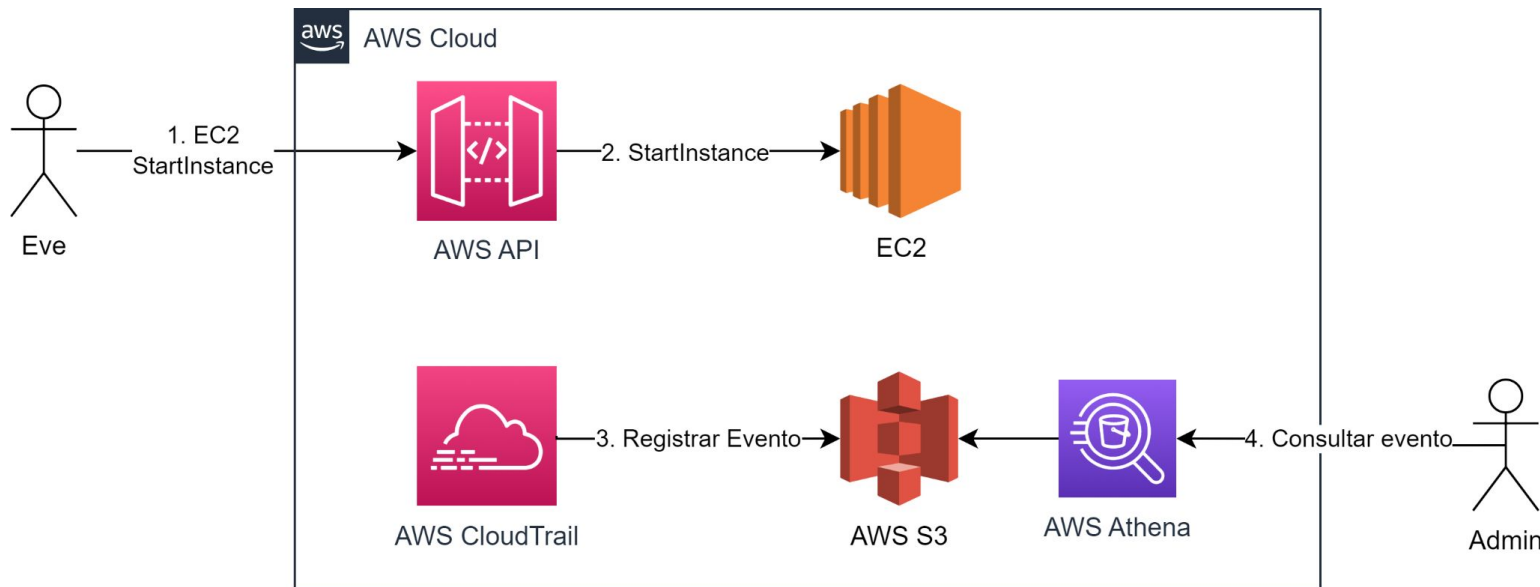
- Nombre de la petición
- IP origen
- Timestamp
- ... etc

Parámetros de la petición y respuesta

```
1  {"Records": [{
2    "eventVersion": "1.0",
3    "userIdentity": {
4      "type": "IAMUser",
5      "principalId": "EX_PRINCIPAL_ID",
6      "arn": "arn:aws:iam::123456789012:user/Alice",
7      "accessKeyId": "EXAMPLE_KEY_ID",
8      "accountId": "123456789012",
9      "userName": "Alice"
10   },
11   "eventTime": "2014-03-06T21:22:54Z",
12   "eventSource": "ec2.amazonaws.com",
13   "eventName": "StartInstances",
14   "awsRegion": "us-east-2",
15   "sourceIPAddress": "205.251.233.176",
16   "userAgent": "ec2-api-tools 1.6.12.2",
17   "requestParameters": {"instancesSet": {"items": [{"instanceId": "i-ebeaf9e2"}]}},
18   "responseElements": {"instancesSet": {"items": [{"instanceId": "i-ebeaf9e2",
19     "currentState": {
20       "code": 0,
21       "name": "pending"
22     },
23     "previousState": {
24       "code": 80,
25       "name": "stopped"
26     }
27   }]}]}]}
28 ]}]}
```

AWS Cloudtrail - Athena

- Almacena los eventos en ficheros en formato JSON que se almacenan en S3
- ¿Cómo los podemos consultar sin tener que descargarlos uno a uno?
 - Amazon Athena



AWS CloudTrail - Retención

- Por defecto, podemos acceder a la actividad de la cuenta de los **últimos 90 días**
 - via API
- Para obtener el registro completo, debemos configurar un seguimiento en CloudTrail
 - Almacenar los logs en S3

AWS CloudTrail - Insights

- CloudTrail Insights es un servicio que analiza los logs de CloudTrail de manera automática
 - Captura llamadas a la API inusuales
 - Detecta picos en el número de llamadas a la API
- Establece un patrón de referencia para detectar actividad inusual
- Si detecta una actividad anómala, almacena el evento para posterior análisis e integrarlo con el pipeline de CloudWatch Alerts.

AWS CloudTrail - Precio

- **Capa gratuita:**
 - Historial de 90 días gratuito (vía API)
 - Primera copia a S3 gratis

Característica

Precios

Eventos de administración enviados a Amazon S3

2,00 USD por cada 100 000 eventos de administración enviados (después de la primera copia gratuita; consulte el [nivel gratuito de AWS](#) para obtener más detalles)

Eventos de datos enviados a Amazon S3

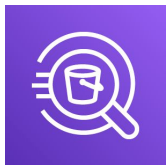
0,10 USD por cada 100 000 eventos de datos enviados

CloudTrail Insights

0,35 USD por cada 100 000 eventos analizados

+ Coste de S3

Amazon Athena



- Nos permite ejecutar consultas SQL sobre datos almacenados en S3
- Características
 - Maneja grandes conjuntos de datos con facilidad
 - Totalmente SaaS Serverless, evita el uso de ETL (Extracción, Transformación y Carga)
 - Pago por consulta
 - Paralelismo



Amazon Athena - Precio

- **5,00 USD por TB** de datos escaneados
- Si tenemos 5 TB de logs, cada consulta nos costará
 - $5,00 * 5,00 = 25 \text{ USD}$
- Nos puede interesar particionar la tabla
 - Reducimos los datos escaneados
 - Ej: (region, año, mes, dia)

Lab 2 - Cat as a Service (CaaS) API

- En la empresa hemos detectado un gran número de bajas por depresión y estrés
 - Los pizza-day de los viernes no están mejorando las estadísticas.
- Según un estudio de la Universidad de Leeds, ver videos e imágenes de animales puede ayudar a reducir el estrés hasta un 50%



Lab 2 - Cat as a Service (CaaS) API

- El departamento de RRHH ha pedido instalar una pantalla en la oficina que proyecte imágenes de gatitos 24/7
- Queremos desplegar una API de gatitos que nos proporcione imágenes aleatorias
- Nuestro compañero Juan ha encontrado una imagen Docker que parece que nos proporciona la funcionalidad
 - ¡El coste de AWS se ha duplicado!
 - ¿Qué ha pasado?

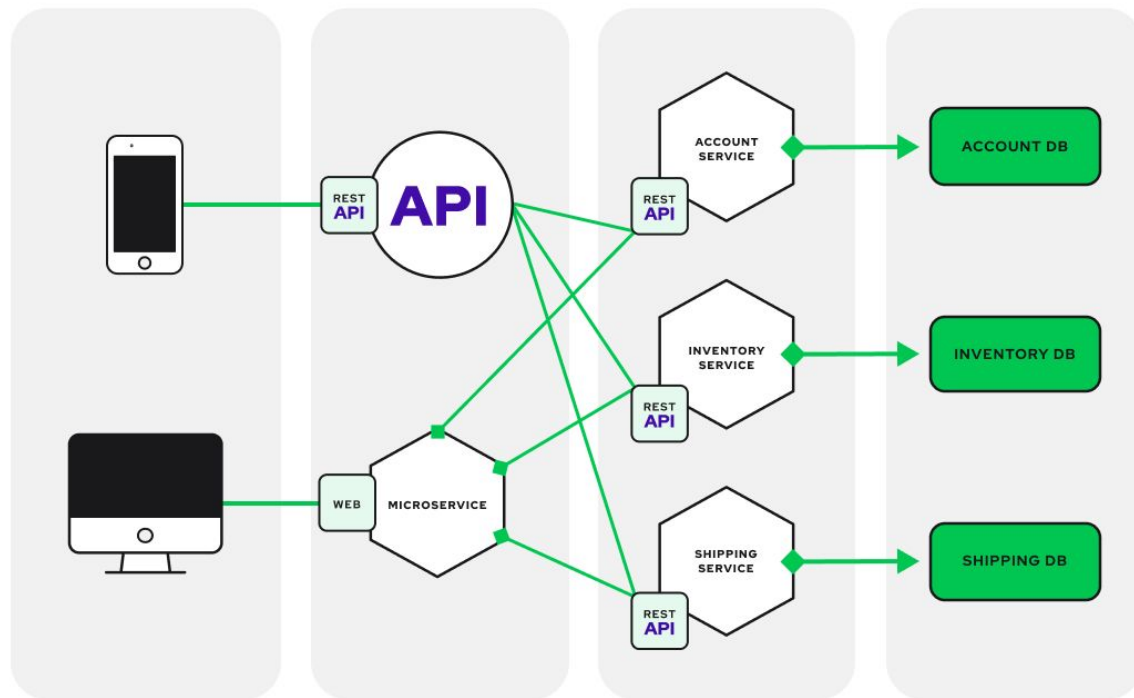


Lab 1 - La Pokedex

- Despliegue con **Terraform**
- API NodeJs
 - Express
- Docker



Monitorización de aplicaciones



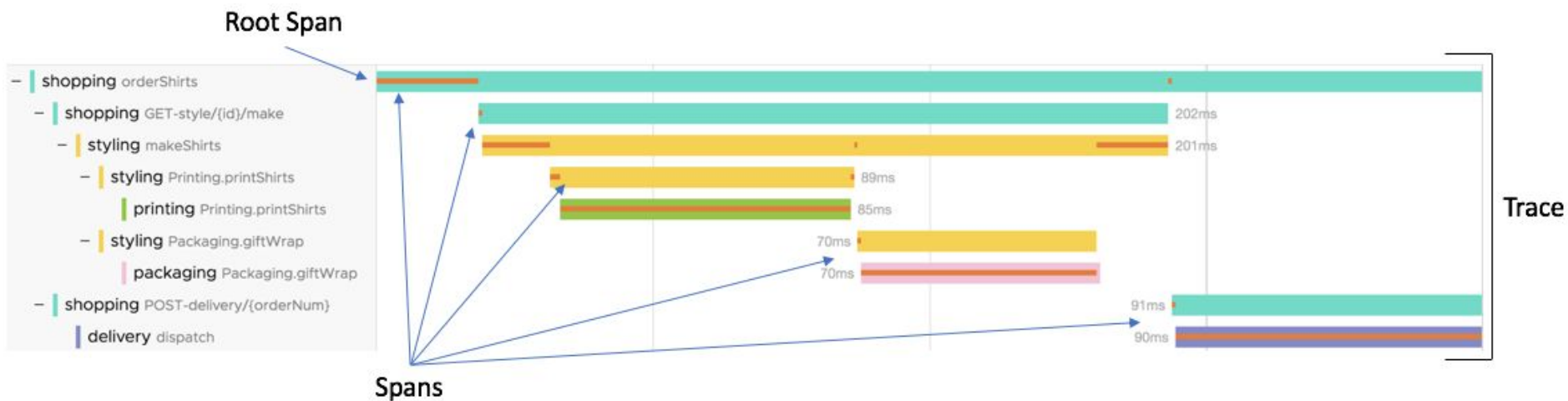
Monitorización de aplicaciones - RED

- Métricas **orientadas a peticiones y a la experiencia**
- **RED**
 - **Rate**: Número de peticiones por segundo
 - **Errors**: Número de errores por segundo
 - **Duration**: Distribuciones de la cantidad de tiempo que tarda cada petición

Monitorización de aplicaciones - Trazas

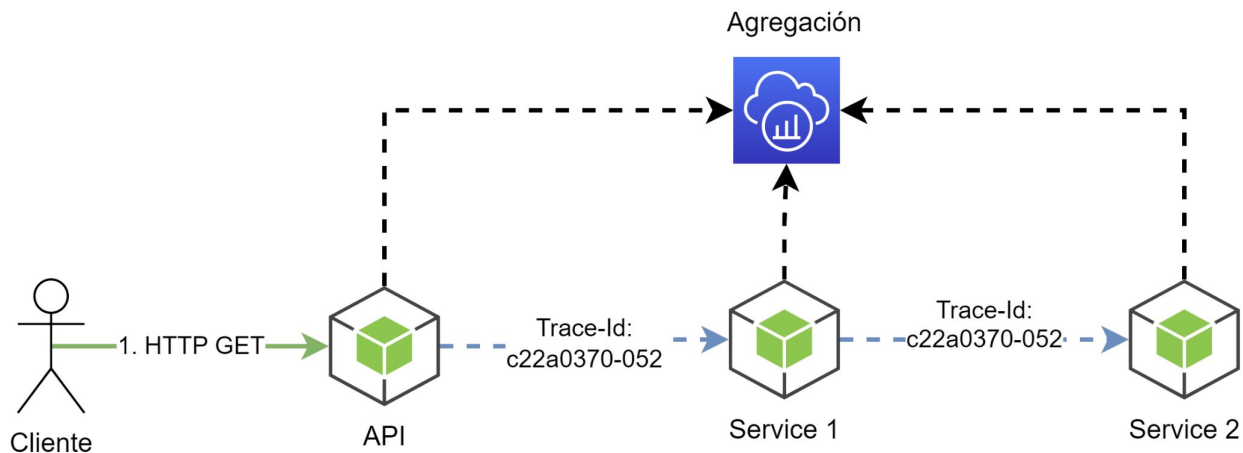
- **Trace** (Traza)
 - Muestra el comportamiento de las peticiones, y como se propagan de un microservicio a otro
 - Incluyen uno o más spans
- **Span** (Segmentos)
 - Segmentos de trabajo dentro de una traza

Monitorización de aplicaciones - Trazas



Monitorización de aplicaciones - Trazas

- ¿Cómo se hace el seguimiento de la petición en sistemas distribuidos?
 - Aplicaciones instrumentalizadas
 - Las peticiones incluyen un Header en las peticiones
 - Identifica la traza / segmento (span) de la que forman parte.

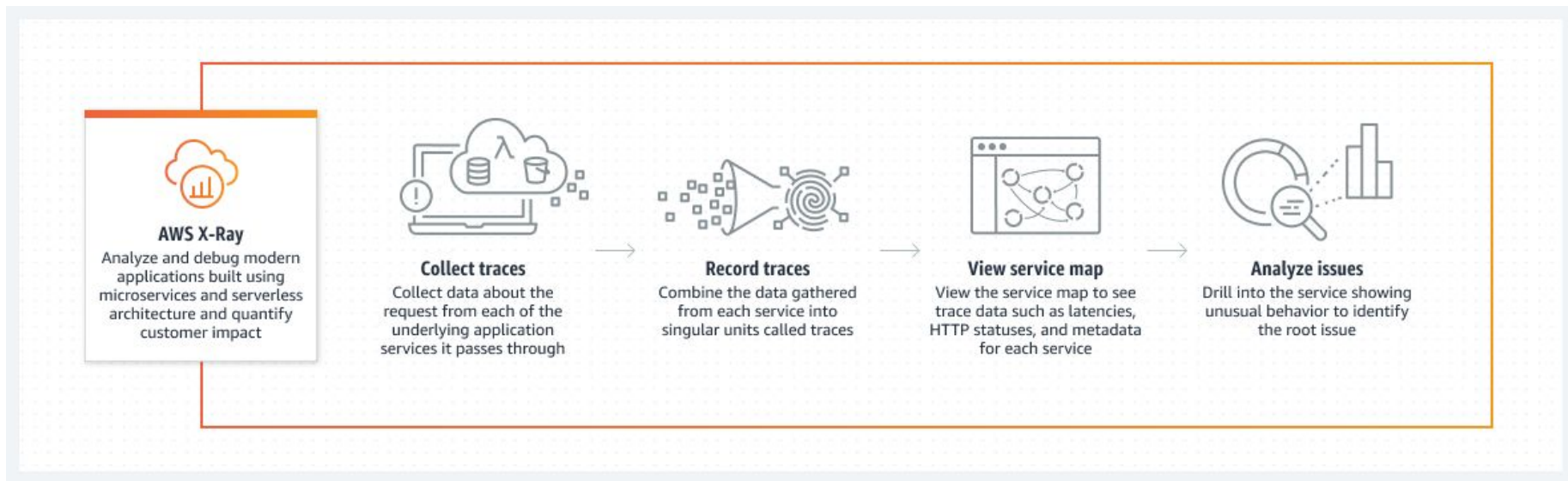


Monitorización de aplicaciones - Mapa de servicios

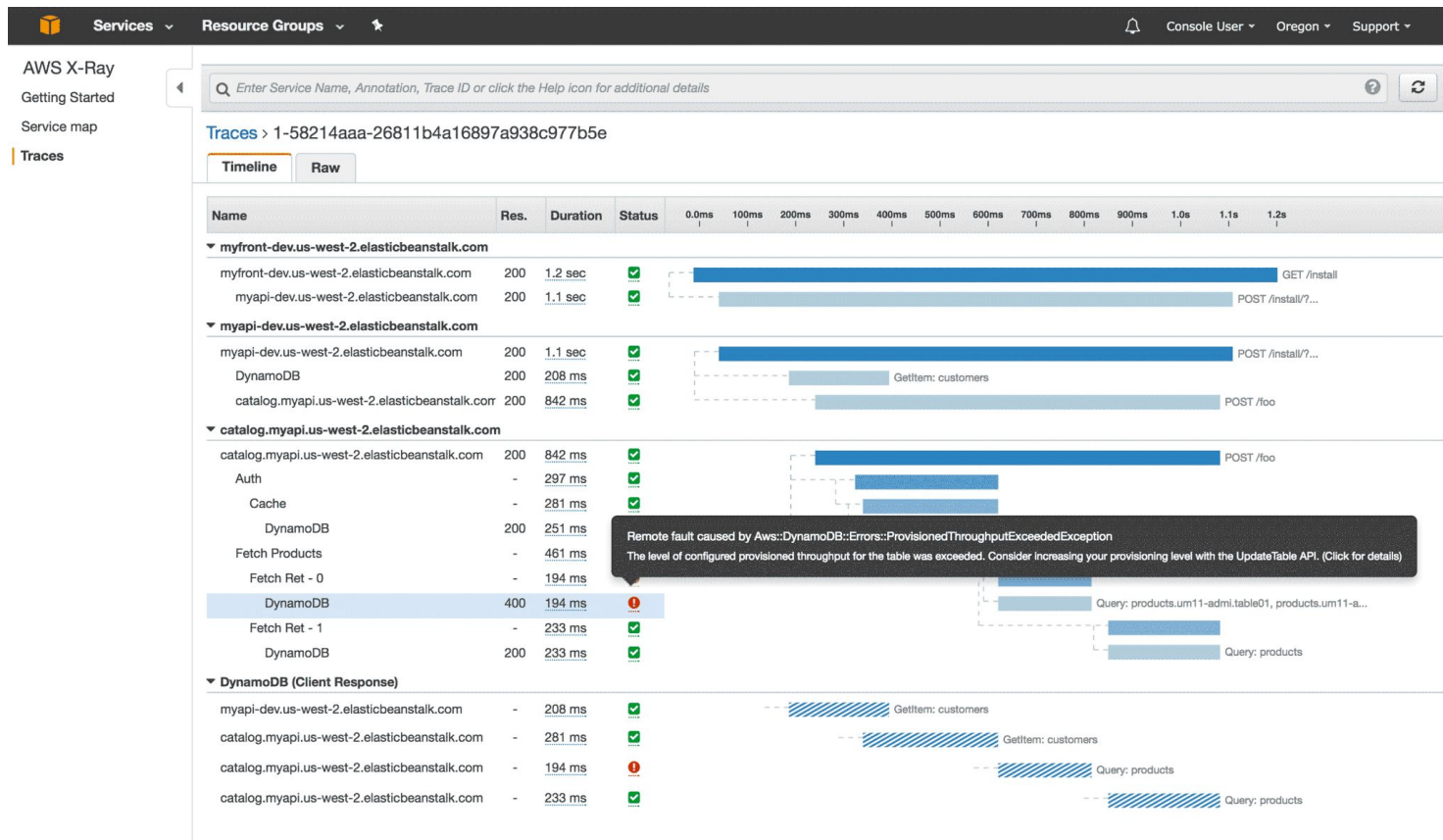


Monitorización de aplicaciones - X-Ray

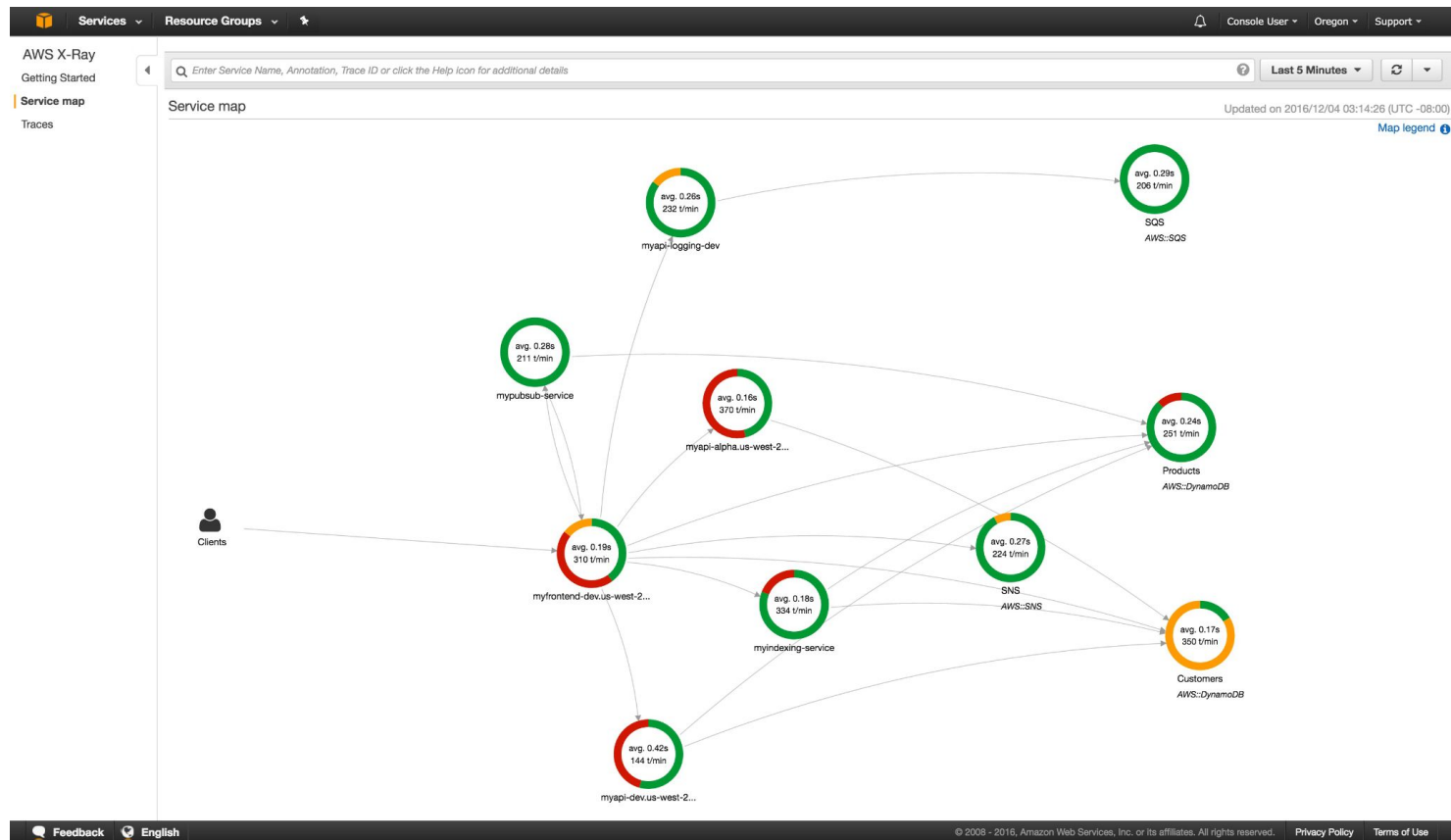
- Se trata de un servicio de monitorización que nos permite generar, recolectar y visualizar trazas de peticiones



Monitorización de aplicaciones - X-Ray



Monitorización de aplicaciones - X-Ray

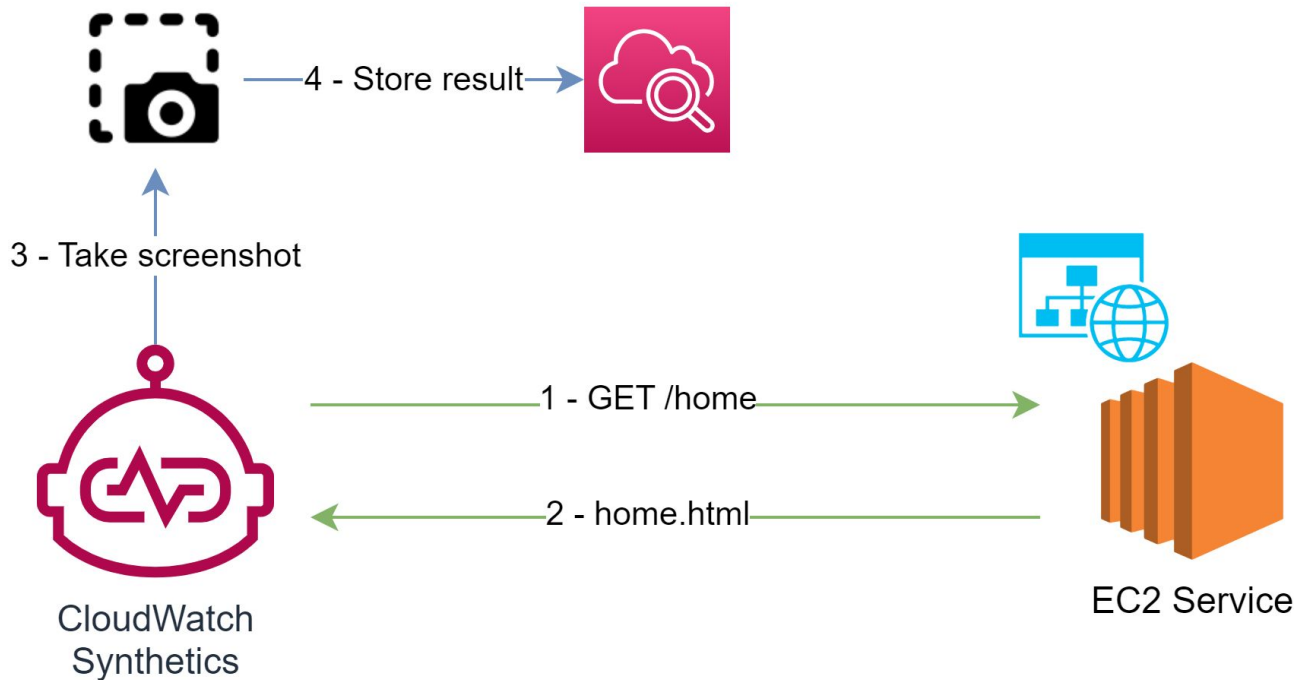


Monitorización de aplicaciones - Monitorización sintética

- En algunos casos de uso, nos puede interesar añadir un tipo de monitorización forzada
- Podemos detectar los problemas antes de que lo hagan los usuarios
- Utilización de *Canaries*
 - Scripts que se ejecutan siguiendo un calendario, monitorizando endpoints y APIs
 - Simulan el comportamiento de un usuario
 - Ejecución con Navegadores headless
- Ejemplos de *Canaries*
 - Spider en busca de links rotos
 - Comparación de la UI

Monitorización de aplicaciones - Monitorización sintética

- **Cloudwatch Synthetics**



Monitorización de aplicaciones - Monitorización sintética


- En algunos casos de uso, nos puede interesar añadir un tipo de monitorización forzada
- Podemos detectar los problemas antes de que lo hagan los usuarios
- Utilización de *Canaries*
 - Scripts que se ejecutan siguiendo un calendario, monitorizando endpoints y APIs
 - Simulan el comportamiento de un usuario
 - Ejecución con Navegadores headless
- Ejemplos de *Canaries*
 - Spider en busca de links rotos
 - Comparación de la UI


Monitorización de aplicaciones - Monitorización sintética


CloudWatch > Create a canary

Create canary [Info](#)

To get started, choose how you would like to create your canary.

**Use a blueprint**
Work from a template script

**Inline Editor**
Edit inline or upload your own scripts

**Import from S3**
Use existing scripts from S3

Blueprints

Heartbeat monitoring
Run a basic page load on a single URL.

API canary
Monitor your APIs as HTTP steps.

Broken link checker
Run a basic web crawler on designated URL.

Canary Recorder
Use the AWS Canary Recorder plugin.

GUI workflow builder
Create a GUI workflow with actions to perform.

Visual monitoring
Compare screenshots for every run

Monitorización de aplicaciones - Cloudwatch Synthetics - Precio

- Los *Canaries* se ejecutan como si fueran funciones *Lambda*, por lo que tenemos que tener en cuenta el coste de:
 - Lambda
 - S3
 - Cloudwatch logs
- + el coste de CloudWatch Synthethics
 - 0,0012 USD por ejecución de *Canary*

Bonus

AWS Config



- Analiza, audita y evalúa las configuraciones y las relaciones de los recursos en tiempo real
- Nos permite definir reglas custom y/o predefinidas para facilitar el cumplimiento de normas / políticas
 - VM EC2 con software no autorizado
 - Contenedor de ECS en modo privilegiado
 - Cloudtrail desactivado
- Packs de conformidad
 - *“Operational Best Practices for Amazon CloudWatch”*
 - *“Operational Best Practices for Amazon S3”*
- Nos permite crear snapshots de configuración
 - S3

AWS Config - Esquema

