

# Cuando la nube se defiende sola

Remediación automática en **AWS** con eventos y  
funciones Lambda

Luis Lunar (Luis Eduardo)  
Noviembre 19, 2025

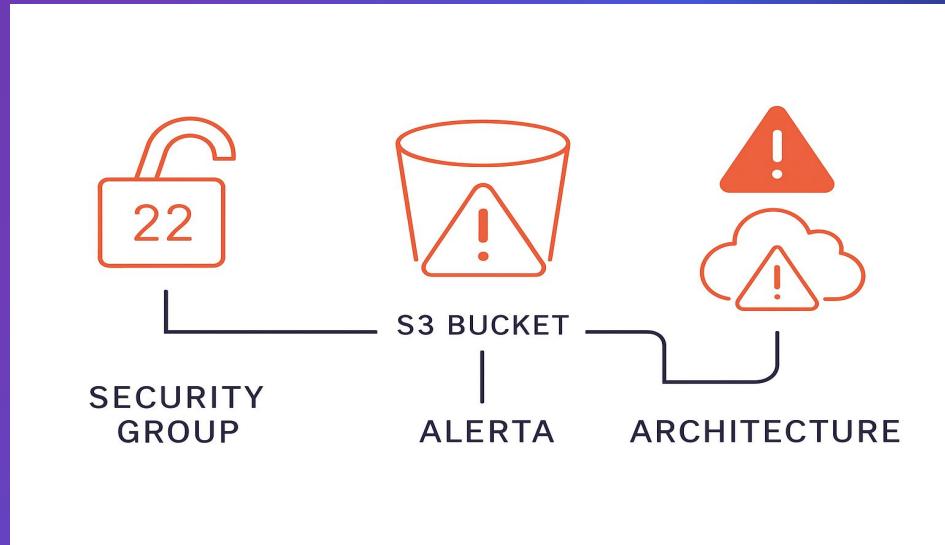


# El conflicto: cuando la nube NO se defiende sola

## El conflicto, cuando una mala configuración abre la puerta

- Un puerto 22 expuesto sin querer
- Un bucket S3 creado sin autorización
- Una regla insegura que nadie ve
- Un cambio que pasa “debajo del radar”

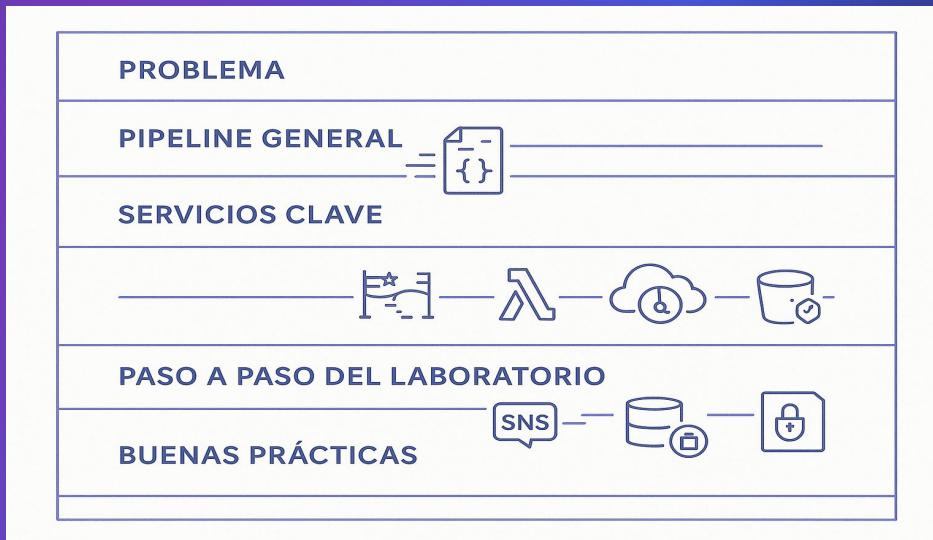
“¿Qué pasa si nadie corrige este error... y nadie se da cuenta?”



# Qué vas a aprender hoy (Objetivos + Mapa de ruta)

## Objetivos de aprendizaje

- Entender cómo AWS detecta acciones sensibles en tiempo real.
- Aprender a diseñar un pipeline de remediación automática.
- Conocer el rol de CloudTrail, EventBridge, Lambda, S3, SNS y CloudWatch.
- Revisar un laboratorio paso a paso sin hacer demo en vivo.
- Llevarte un modelo replicable para tu propia cuenta de AWS.



# Visión global del pipeline de remediación automática

## Pipeline de seguridad automatizada flujo completo:

1. CloudTrail registra la acción.
2. EventBridge filtra el evento
3. Lambda ejecuta la remediación.
4. CloudWatch Logs registra la evidencia.
5. SNS envía la alerta.
6. S3 centraliza la auditoría.



**Resultado:** La nube detecta, corrige y notifica sin intervención manual.

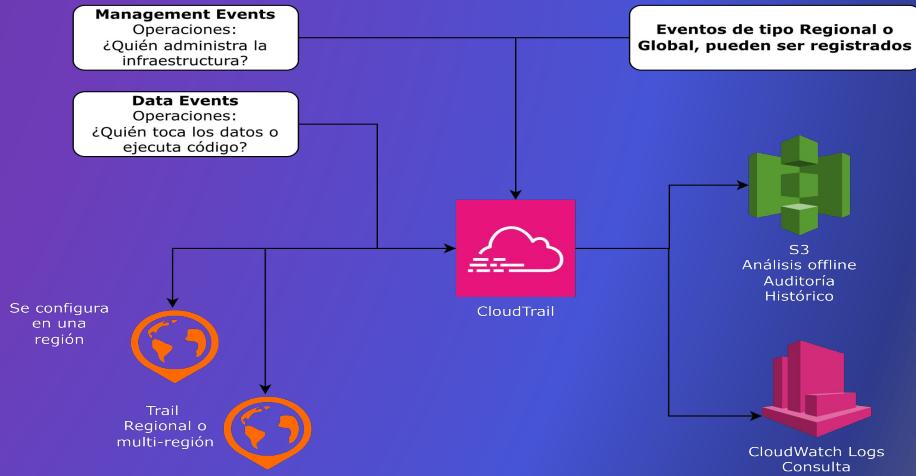


# CloudTrail: ojos y memoria de la nube

## CloudTrail: registro y trazabilidad total

- Registra cada acción API en tu cuenta.
- Captura *quién, cuándo, desde dónde y qué hizo*.
- Fuente primaria de eventos para seguridad y auditoría.
- Alimenta a EventBridge en tiempo real.

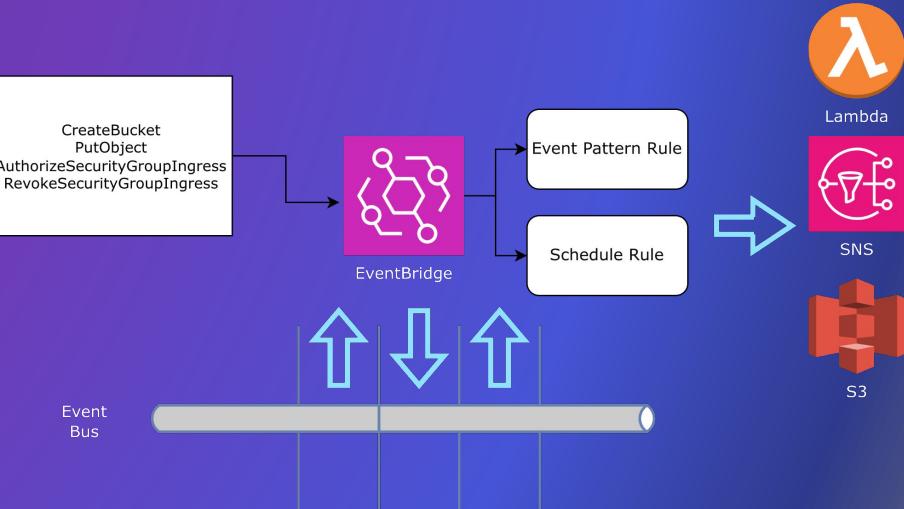
**“Si pasó en tu cuenta,  
CloudTrail lo sabe.”**



# EventBridge: el motor que detecta y dispara acciones

## EventBridge: detección inteligente en tiempo real

- Detecta eventos de CloudTrail en tiempo real.
- Filtra acciones sensibles mediante *Event Patterns*.
- Ejecuta automatizaciones vía Lambda.
- Permite reglas por servicio, acción, usuario, IP, región, etc.
- Base del modelo event-driven en seguridad.



**“EventBridge convierte eventos en acciones.”**



# Lambda: el cerebro de la remediación

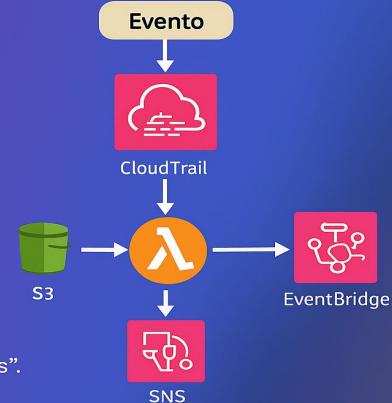
**Lambda toma el evento que detectó EventBridge y aplica la corrección necesaria.**

- EventBridge, no corrige nada.
- SNS, no corrige nada.
- CloudWatch Logs, no corrige nada.

**Lambda: el cerebro de la remediación**



- Ejecuta acciones correctivas automáticamente.
- Dos funciones en tu laboratorio:
  - Elimina bucket no autorizado tras un “CreateBucket”.
  - Revoca regla SSH 22 insegura tras un “AuthorizeSecurityGroupIngress”.



**“Solo Lambda ejecuta la  
remediación”**



# ¿Qué tipo de remediaciones hace Lambda en tu laboratorio?

Lambda realiza dos acciones críticas:

a) **Remediación 1:** Bucket no autorizado

Ante un CreateBucket:

- Verifica si el bucket cumple las reglas esperadas.
- Si NO cumple → elimina automáticamente el bucket.
- Registra evidencia.
- Envía notificación vía SNS.

“Este es un control **preventivo + correctivo** inmediato.”

b) **Remediación 2:** Regla SSH 22 abierta

Ante un AuthorizeSecurityGroupIngress:

- Analiza el contenido del evento.
- Si detecta algo como 0.0.0.0/0 + puerto 22.
- Elimina la regla insegura.
- Registra evidencia en CloudWatch Logs.
- Envía alerta vía SNS.

“Aquí Lambda te protege de un riesgo de **alto impacto**.”



# ¿Qué contiene el evento que recibe Lambda?

El evento que recibe Lambda es un JSON 100% igual al evento original de CloudTrail filtrado por EventBridge.

Contiene:

- Nombre del servicio: "eventSource": "s3.amazonaws.com".
- Operación: "eventName": "CreateBucket".
- Identidad del usuario: userIdentity.
- Dirección IP: sourceIPAddress.
- Recurso afectado: requestParameters.
- Hora exacta, región y cuenta.

Este JSON es suficiente para:

- Identificar el bucket creado.
- Identificar la regla de seguridad agregada.
- Decidir la acción automática.

"Lambda no adivina nada,  
simplemente lee el evento."



# Lambda necesita permisos (y por eso el rol IAM es clave)

**Lambda NO puede corregir nada si no tiene permisos, debes darle un rol que permita:**

Para S3:

- s3>DeleteBucket
- s3>ListBucket
- s3:GetBucketPolicy
- etc.

Para Security Groups:

- ec2:RevokeSecurityGroupIngress
- ec2:DescribeSecurityGroups

Para logs:

- logs>CreateLogGroup
- logs:PutLogEvents.

Y para SNS:

- sns:Publish.



**“Lambda es la pieza que convierte un evento en una acción concreta, es donde ocurre la remediación automática. Sin Lambda, detectaríamos el problema... pero no haríamos nada al respecto.”**



# CloudWatch Logs, S3 y SNS: evidencias y alertas

**CloudWatch Logs es donde Lambda deja todo registrado:**

- Qué acción ejecutó.
- Qué recurso afectó.
- Qué decisión tomó.
- Qué valores encontró en el evento JSON.
- El resultado de la remediación.
- Errores, advertencias y detalles internos.

**“CloudWatch Logs es donde queda la evidencia técnica del proceso.**

**Si quieres saber qué hizo Lambda exactamente, lo ves aquí.”**



# S3 → “El repositorio de auditoría”

S3 funciona como **almacenamiento histórico**.

Mientras CloudWatch Logs guarda evidencia técnica temporal, S3 guarda evidencia permanente.

En el laboratorio, S3 sirve como:

- Repositorio de evidencia.
- Archivos históricos.
- Punto de control para auditoría

“CloudWatch Logs te dice qué pasó. S3 te **guarda** el registro histórico para auditoría.”



# SNS → “La alerta inmediata”

**SNS no guarda nada.**

**SNS no analiza nada.**

**SNS no ejecuta nada.**

**SNS notifica lo que pasó.**

En el pipeline:

- Lambda remedia.
- Lambda publica en SNS.
- SNS envía un correo al equipo técnico o de seguridad.

“SNS no forma parte de la  
remediación, forma parte de la  
**comunicación**, es la voz del  
pipeline.”



# Laboratorio - Pasos



# PASO 1 – Crear bucket S3 de evidencias

## PASO 1 – Crear bucket de evidencias en S3

Ir a S3 → Buckets - Clic en Create bucket

**Configurar:**

**Bucket name:** audit-evidence-<randomkey>

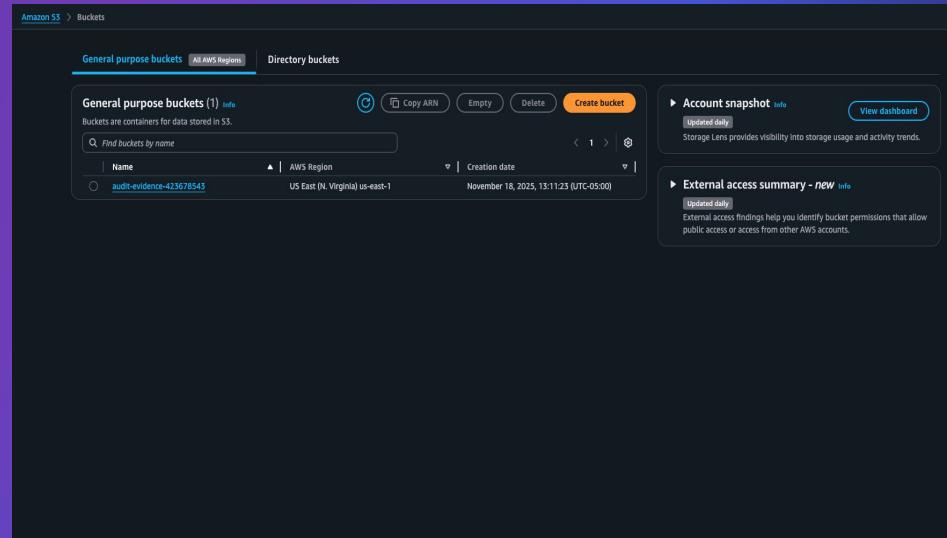
**Región:** us-east-1 (N. Virginia)

**Block Public Access:** Activado (por defecto)

**Object Ownership:** ACLs disabled

Ejemplo:audit-evidence-a92fbd

**Nota:** No cambiar encryption, versioning ni configuraciones avanzadas.



# PASO 2 — Crear Trail en CloudTrail

## PASO 2 — Habilitar el registro de eventos con CloudTrail

Ir a CloudTrail → Trails – Clic en Create trail

**Configurar:**

**Trail name:** AuditTrail

**S3 bucket:** Crear nuevo:

cloudtrail-logs-<randomkey>

**Management events:** Read/Write (activado)

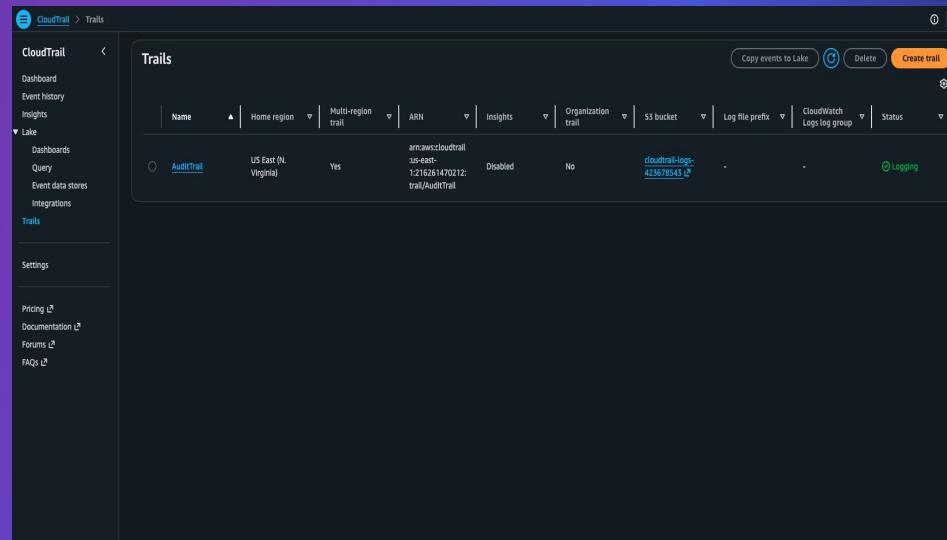
**Data events:** Opcional (dejar apagado para este lab)

**Log file SSE-KMS Encryption:** Dejar por defecto (opcional)

Clic en Create trail

**Ejemplo de bucket:**

cloudtrail-logs-a4ldcb



Name	Home region	Multi-region trail	ARN	Insights	Organization trail	S3 bucket	Log file prefix	CloudWatch Logs log group	Status
AuditTrail	US East (N. Virginia)	Yes	arn:aws:cloudtrail:us-east-1:216261470212:trail/AuditTrail	Disabled	No	cloudtrail-logs-423678541			Logging



# PASO 3 – Crear SNS Topic + Suscripción por Email

## PASO 2 – Habilitar el registro de eventos con CloudTrail

Ir a SNS (Simple Notification Service)

**Configurar:**

\*Menú izquierdo → Topics

Clic en Create topic

**Tipo:** Standard

**Nombre:** AuditNotificationsTopic

Dejar el resto por defecto → Clic en Create topic

**Configurar:**

\*Crear suscripción por correo

Dentro del Topic → Subscriptions

Clic en Create subscription

**Protocol:** Email – **Endpoint:** (tu correo real, ej: [miemail@gmail.com](mailto:miemail@gmail.com))

Clic en Create subscription



# PASO 3 – Crear SNS Topic + Suscripción por Email

Amazon SNS > Topics > Create topic

### Create topic

**Details**

Type [Info](#)  
Topic type cannot be modified after topic is created

FIFO (first-in, first-out)  

- Strictly-preserved message ordering
- Exactly-once message delivery
- Delivery semantics: SQS

Standard  

- Best-effort message delivery
- At-least once message delivery
- Delivery protocols: SNS, Lambda, Data Firehose, HTTP, SMS, email, mobile application endpoints

Name [AuditNotificationsTopic](#)  
Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (\_).

Display name - optional [Info](#)  
To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message.

AuditNotificationsTopic  
Maximum 100 characters.

► Encryption - optional [Info](#)  
Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.

► Access policy - optional [Info](#)  
This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic.

► Data protection policy - optional [Info](#)  
This policy defines which sensitive data to monitor and to prevent from being exchanged via your topic.

Amazon SNS > Subscriptions > Create subscription

### Create subscription

**Details**

Topic ARN [Q emawsnsus-e9](#)

Protocol [Email](#)  
The type of endpoint to subscribe

Endpoint An email address that can receive notifications from Amazon SNS.  
llunorg@example.com

After your subscription is created, you must confirm it. [Info](#)

► Subscription filter policy - optional [Info](#)  
This policy filters the messages that a subscriber receives.

► Redrive policy (dead-letter queue) - optional [Info](#)  
Send undeliverable messages to a dead-letter queue.

[Cancel](#) [Create subscription](#)



# PASO 4 – Crear la función Lambda de remediación – Parte 1

## PASO 2 – Habilitar el registro de eventos con CloudTrail

Ir a Lambda → Create function

### Configurar:

Name: ProcessAuditEvent

Runtime: Python 3.12

Architecture: x86\_64 (o ARM si lo prefieres)

Execution role:

Usar rol existente:

ProcessAuditEventExecutionRole

### Verificar permisos del rol

El rol debe incluir (previamente creado):

- AWSLambdaBasicExecutionRole
- AmazonS3FullAccess
- AmazonEC2FullAccess
- AmazonSNSFullAccess
- CloudWatchLogsFullAccess

### Configurar variables de entorno

EVIDENCE\_BUCKET = audit-evidence-<id>

SNS\_TOPIC\_ARN =

arn:aws:sns:us-east-1:<id>:AuditNotificationsTopic

Finalmente **pegar el código** compartido por la comunidad.



# PASO 4 – Crear la función Lambda de remediación – Parte 2

Lambda > Functions > Create function

### Create function Info

Choose one of the following options to create your function.

Author from scratch  
Start with a simple Hello World example.

Use a blueprint  
Build a Lambda application from sample code and configuration presets for common use cases.

Container image  
Select a container image to deploy for your function.

#### Basic information

**Function name**  
Enter a name that describes the purpose of your function.

Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, hyphens (-), and underscores (\_).

**Runtime** Info  
Choose the language to use for your function. Note that the console code editor supports only Node.js, Python, and Ruby.  
 

**Architecture** Info  
Choose the instruction set architecture you want for your function code.  
 arm64  
 x86\_64

#### Permissions Info

By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

▶ Change default execution role

#### Additional configurations

Use additional configurations to set up networking, security, and governance for your function. These settings help secure and customize your Lambda function deployment.

Cancel Create function

Info Tutorials

Learn how to implement common use cases in AWS Lambda.

Create a simple web app ^

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

Learn more 

Start tutorial



# PASO 4 – Crear la función Lambda de remediación – Parte 3

Lambda > Functions > ProcessAuditEvent > Edit environment variables

## Edit environment variables

Environment variables

You can define environment variables as key-value pairs that are accessible from your function code. These are useful to store configuration settings without the need to change function code. [Learn more](#)

Key	Value	Remove
EVIDENCE_BUCKET	audit-evidence-423678543	Remove
SNS_TOPIC_ARN	arn:aws:sns:us-east-1:123456789012:audit-evidence-topic	Remove

Add environment variable

Encryption configuration

Cancel Save

Tutorials

Learn how to implement common cases in AWS Lambda.

Create a simple web app

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

[Learn more](#)

[Start tutorial](#)

The screenshot displays the AWS Lambda console interface. The main area is titled 'Edit environment variables' and contains a table with two rows. The first row has 'EVIDENCE\_BUCKET' as the key and 'audit-evidence-423678543' as the value. The second row has 'SNS\_TOPIC\_ARN' as the key and 'arn:aws:sns:us-east-1:123456789012:audit-evidence-topic' as the value. Each row has a 'Remove' button to its right. Below the table is a blue 'Save' button. To the right of the main content is a sidebar titled 'Tutorials' which includes a section for 'Create a simple web app' with a brief description and a 'Start tutorial' button.



# PASO 5 – Crear reglas en EventBridge –

## Parte 1

### PASO 5 – Configurar reglas EventBridge para activar Lambda

#### Regla: CreateBucket

Detecta cuando se crea un bucket sin autorización.

- **Rule name:** DetectCreateBucket
- **Service:** S3
- **Event type:** CreateBucket
- **Target:** ProcessAuditEvent
- **Event Pattern JSON:**

```
{  
  "source": ["aws.s3"],  
  "detail-type": ["AWS API Call via  
CloudTrail"],  
  "detail": {  
    "eventSource": ["s3.amazonaws.com"],  
    "eventName": ["CreateBucket"]  
  }  
}
```



# PASO 5 – Crear reglas en EventBridge – Parte 2

## PASO 5 – Configurar reglas EventBridge para activar Lambda

### Regla: CreateBucket

Detecta cuando se crea un bucket sin autorización.

- **Rule name:** DetectInsecureSSH
- **Service:** EC2
- **Event:** Reglas de SG modificadas
- **Target:** ProcessAuditEvent
- **Event Pattern JSON:**

```
{  
    "source": ["aws.ec2"],  
    "detail": {  
        "eventSource":  
        ["ec2.amazonaws.com"],  
        "eventName": [  
            "AuthorizeSecurityGroupIngress",  
            "RevokeSecurityGroupIngress",  
            "ModifySecurityGroupRules"  
        ]  
    }  
}
```



# PASO 5 – Crear reglas en EventBridge – Parte 1

The screenshot shows the Amazon EventBridge console with the path: Amazon EventBridge > Rules > DetectCreateBucket > Edit rule.

**Event pattern** (Info)

**Creation method**

- Use schema: Use an Amazon EventBridge schema to generate the event pattern.
- Use pattern form: Use a template provided by EventBridge to create an event pattern.
- Custom pattern (JSON editor): Write an event pattern in JSON.

**Event source**  
AWS service or EventBridge partner as source

**AWS services** dropdown: AWS services

**AWS service**  
The name of the AWS service as the event source

**S3 (Simple Storage Service)** dropdown: S3 (Simple Storage Service)

**Event type**  
The type of events as the source of the matching pattern

**Bucket-Level API Call via CloudTrail** dropdown: Bucket-Level API Call via CloudTrail

**Event pattern**  
Event pattern, or filter to match the events

```
1 {  
2   "source": ["aws.s3"],  
3   "detail-type": ["AWS API Call via CloudTrail"],  
4   "detail": {  
5     "eventSource": ["s3.amazonaws.com"],  
6     "eventName": ["CreateBucket"]  
7   }  
8 }
```

**Event Type Specification 1**

Any operation

Specific operation(s)

Specific operation(s)

CreateBucket

Copy Test pattern Edit pattern



# PASO 5 – Crear reglas en EventBridge – Parte 2

The screenshot shows the 'Edit rule' page in the Amazon EventBridge console. The left sidebar navigation includes 'Amazon EventBridge', 'Dashboard', 'Developer resources' (Learn, Sandbox, Quick starts), 'Buses' (Event buses, Rules Updated, Global endpoints, Archives, Replays), 'Pipes' (Pipes), 'Scheduler' (Schedules, Schedule groups), 'Integration' (Partner event sources, API destinations, Connections), and 'Schema registry' (Schemas). The main content area is titled 'DetectSSHOpenToWorld' under 'Rules'. It features a 'Sample event - optional' section with a note about referencing it for event patterns or filter criteria. The 'Event pattern' section has three creation methods: 'Use schema' (selected), 'Use pattern form', and 'Custom pattern (JSON editor)' (highlighted with a blue border). The 'Event pattern' input field contains the following JSON code:

```
1 {
2   "source": ["aws.ec2"],
3   "detail": [
4     "eventSource": ["ec2.amazonaws.com"],
5     "eventName": ["AuthorizeSecurityGroupIngress", "RevokeSecurityGroupIngress", "ModifySecurityGroupRules"]
6   ]
7 }
```

Below the input field are buttons for 'Prefix matching' (radio button), 'Content-based filter syntax' (radio button selected), 'Insert' (button), and 'JSON is valid' (radio button). At the bottom are buttons for 'Copy', 'Pretty', 'Event pattern form', and 'Test pattern'.



# PASO 6 — Pruebas Reales del Pipeline

## Escenario S3: Bucket no autorizado

Crear un bucket temporal (cualquier nombre). Esperar 5–10 segundos mientras EventBridge procesa el evento.

Confirmar que el bucket se eliminó automáticamente.

Revisar CloudWatch Logs → buscar log de Lambda:

- Acción detectada: CreateBucket
- Bucket marcado como no autorizado
- Acción realizada: DeleteBucket

## Verificar correo SNS confirmando la remediación.

## Escenario Security Group: SSH inseguro

Crear o seleccionar un Security Group de prueba. Agregar la regla:

- Port 22
- Source: 0.0.0.0/0

Esperar 5–10 s: Lambda debe eliminar automáticamente la regla insegura.

Validar en CloudWatch Logs:

- Evento: AuthorizeSecurityGroupIngress
- Evaluación de riesgo
- Remediación ejecutada: RevokeSecurityGroupIngress



# PASO FINAL — Limpieza del entorno

## Limpieza del laboratorio

- Eliminar Lambda
- Eliminar reglas EventBridge
- Eliminar SG y buckets de prueba
- Eliminar SNS Topic si ya no se usa



# Referencias y Documentación Oficial

## AWS — Servicios utilizados en el laboratorio

CloudTrail (Registro y auditoría)

<https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudtrail-user-guide.html>

EventBridge (Reglas y automatización)

<https://docs.aws.amazon.com/eventbridge/latest/userguide/what-is-amazon-eventbridge.html>

AWS Lambda (Remediación automática)

<https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>

Amazon S3 (Evidencias y almacenamiento)

<https://docs.aws.amazon.com/AmazonS3/latest/userguide>Welcome.html>

Amazon SNS (Notificaciones por correo)

<https://docs.aws.amazon.com/sns/latest/dg/welcome.html>

