Laboratorio #7

Monitoreo de Redes con Amazon CloudWatch

Proyecto:
Laboratorios Virtuales de Redes en AWS para el
Fortalecimiento de Competencias en Redes de Nueva Generación

Estudiantes:
Nicolás Carreño Tascón
Juan Manuel Canchala Jiménez

Director:
Carlos Olarte

Asignatura:
Redes de Nueva Generación

Duración Estimada: 90 minutos
Costo: \$0.00 (100 % Gratuito - Free Tier)

Diciembre 2025

Índice

Resumen

Este laboratorio aborda el **monitoreo de redes en AWS** utilizando Amazon CloudWatch como plataforma central para recopilar, visualizar y reaccionar ante métricas y logs relacionados con el tráfico de red. Se integra el uso de **métricas nativas**, **VPC Flow Logs**, **CloudWatch Logs Insights**, **alarmas** y **dashboards** para construir un entorno de observabilidad orientado a la capa de red.

El estudiante aprenderá a habilitar y analizar **VPC Flow Logs** para observar el comportamiento de la red (tráfico permitido/denegado), crear **alarmas** (hasta 10 gratuitas dentro del Free Tier) sobre métricas clave, construir un **dashboard de monitoreo** de red y ejecutar **consultas en CloudWatch Logs Insights** para investigar patrones de tráfico y posibles anomalías. Además, se presentan **mejores prácticas** de monitoreo que son aplicables en entornos de producción, como la selección de métricas relevantes, el uso de umbrales adecuados y la correlación de eventos.

El laboratorio está diseñado para ejecutarse dentro de los límites del **Free Tier** de AWS, aprovechando que CloudWatch proporciona de forma gratuita un subconjunto de métricas y hasta **10 alarmas** sin costo adicional, siempre que el uso de logs y consultas se mantenga en un nivel moderado.

Palabras clave: Amazon CloudWatch, Monitoreo de Redes, VPC Flow Logs, Alarmas, Dashboards, Logs Insights, Observabilidad, Free Tier.

1 Objetivos

1.1 Objetivo General

Diseñar e implementar un esquema básico de **monitoreo de red** en AWS apoyado en Amazon CloudWatch, utilizando métricas de red, VPC Flow Logs, alarmas y dashboards, aplicando mejores prácticas de observabilidad dentro del Free Tier.

1.2 Objetivos Específicos

- Comprender el rol de Amazon CloudWatch como servicio central de monitoreo en AWS.
- Habilitar y analizar **VPC Flow Logs** para observar tráfico aceptado y rechazado en una VPC.
- Crear y configurar **alarmas de CloudWatch** basadas en métricas de red, respetando el límite de 10 alarmas gratuitas.
- Construir un **dashboard de monitoreo** que incluya gráficos de métricas y paneles de estado relevantes para la red.
- Utilizar **CloudWatch Logs Insights** para ejecutar consultas sobre VPC Flow Logs y extraer información útil.
- Aplicar **mejores prácticas de monitoreo**, como la definición de umbrales, la selección de ventanas de tiempo y la priorización de señales.

1.3 Competencias a Desarrollar

- **Observabilidad en la nube:** Capacidad para instrumentar y supervisar recursos de red en AWS.
- **Análisis de logs de red:** Interpretación de VPC Flow Logs para entender patrones de tráfico y posibles problemas de conectividad o seguridad.
- **Diseño de tableros de monitoreo:** Creación de dashboards que resuman el estado de la red y faciliten la toma de decisiones.
- **Gestión de alarmas:** Configuración de alarmas efectivas que notifiquen condiciones anómalas sin generar ruido excesivo.
- **Buenas prácticas de operación:** Enfoque sistemático para monitorear, alertar y responder a eventos de red.

2 Marco Teórico

2.1 Introducción a Amazon CloudWatch

Amazon CloudWatch es el servicio de monitoreo y observabilidad nativo de AWS. Permite:

- Recopilar **métricas** (CPU, red, disco, etc.) de servicios como EC2, RDS, VPC, ELB, entre otros.
- Recibir y almacenar **logs** de aplicaciones, sistemas operativos y servicios administrados.
- Definir **alarmas** que se disparan cuando una métrica cruza un umbral.
- Construir **dashboards** con visualizaciones en tiempo casi real.
- Ejecutar **consultas** sobre logs mediante **CloudWatch Logs Insights**.

CloudWatch se organiza en diferentes componentes:

- **Métricas (Metrics):** Valores numéricos en el tiempo, organizados en *namespaces*.
- **Logs:** Flujos de eventos agrupados en *log groups* y *log streams*.
- **Alarmas:** Condiciones evaluadas sobre métricas que generan un cambio de estado (OK, ALARM, INSUFFICIENT_DATA).
- **Dashboards:** Paneles personalizados con widgets de métricas, logs y texto.

2.2 Métricas de Red en CloudWatch

Muchos servicios de AWS generan métricas relacionadas con la red, entre ellas:

- **EC2:** NetworkIn, NetworkOut, NetworkPacketsIn, NetworkPacketsOut.
- **Elastic Load Balancing:** RequestCount, HTTPCode_ELB_4XX, HTTPCode_ELB_5XX, etc.
- **NAT Gateway, VPN, Transit Gateway:** métricas específicas de tráfico y errores.

Aunque las **VPC** como tal no exponen muchas métricas de L3 en CloudWatch, la combinación de:

- Métricas de instancias y balanceadores.
- Logs de flujo de VPC (**VPC Flow Logs**).

permite construir una vista bastante completa del comportamiento de la red.

2.3 VPC Flow Logs

2.3.1. Definición

VPC Flow Logs es una característica que permite capturar información sobre el tráfico IP que entra y sale de:

- Interfaces de red de instancias (ENI).
- Subredes.
- VPCs completas.

Los logs contienen información como:

- **srcaddr, dstaddr**: IP de origen y destino.
- **srcport, dstport**: puertos de origen y destino.
- **protocol**: protocolo (TCP, UDP, ICMP).
- **action**: `ACCEPT` o `REJECT`.
- **bytes, packets**: volumen de datos y número de paquetes.

Estos logs pueden enviarse a:

- **CloudWatch Logs**.
- **Amazon S3**.

En este laboratorio utilizaremos **CloudWatch Logs** para:

- Visualizar los eventos.
- Consultarlos con **Logs Insights**.

2.3.2. Formato de un Registro de Flow Log (Versión Básica)

Un registro típico (simplificado) puede verse así:

```
version account-id interface-id srcaddr dstaddr srcport dstport protocol
packets bytes start end action log-status
2 123456789012 eni-0abc123def456 10.10.1.10 10.20.1.20 443 51532 6
10 840 1699980000 1699980060 ACCEPT OK
```

2.4 CloudWatch Logs y Logs Insights

2.4.1. CloudWatch Logs

CloudWatch Logs almacena flujos de logs en:

- **Log groups:** contenedores lógicos (por ejemplo, `/aws/vpc/flow-logs/lab7`).
- **Log streams:** secuencias de eventos (por ejemplo, por ENI o por instancia).

Permite:

- Definir **retención** de logs (días).
- Crear **métricas derivadas** (metric filters).

2.4.2. CloudWatch Logs Insights

CloudWatch Logs Insights es un motor de consulta interactivo para logs. Permite:

- Escribir consultas tipo SQL simplificado.
- Filtrar por campos, agrupar, contar, hacer sumas, etc.
- Visualizar resultados en tablas y gráficos.

Ejemplo de consulta sobre VPC Flow Logs para contar conexiones rechazadas por IP de origen:

```
fields srcaddr, dstaddr, action
| filter action = 'REJECT'
| stats count(*) as rechazos by srcaddr
| sort rechazos desc
| limit 20
```

Listing 1: Ejemplo de consulta en CloudWatch Logs Insights

2.5 Alarmas de CloudWatch

Una **alarma de CloudWatch**:

- Se basa en una métrica (o métrica compuesta).
- Evalúa una condición (por ejemplo, *NetworkIn* mayor a cierto umbral durante N períodos).
- Cambia de estado: `OK`, `ALARM` o `INSUFFICIENT_DATA`.
- Puede ejecutar acciones: enviar notificaciones (SNS), ejecutar acciones automáticas, etc.

En el contexto de este laboratorio:

- Aprovecharemos que el Free Tier ofrece hasta **10 alarmas** sin costo.
- Crearemos alarmas **sencillas pero significativas** relacionadas con:
  - Volumen de tráfico inusual.
  - Número de paquetes rechazados (vía métrica derivada de logs).

2.6 Dashboards de Monitoreo

Un **CloudWatch Dashboard** es una vista personalizable que puede incluir:

- Gráficos de métricas de red.
- Widgets de texto explicativo.
- Gráficos de resultados de consultas de Logs Insights (a través de métricas derivadas).

En este laboratorio crearemos un dashboard básico llamado `Lab7-Network-Monitoring` con:

- Panel de tráfico de red (NetworkIn/NetworkOut).
- Panel de recuento de conexiones rechazadas.
- Panel de estado de alarmas.

2.7 Mejores Prácticas de Monitoreo de Redes

Algunas recomendaciones clave:

- **Medir antes de optimizar:** No se puede mejorar lo que no se mide.
- **Elegir métricas representativas:** Volumen de tráfico, errores, latencia (cuando aplique), rechazos de firewall, etc.
- **Evitar el ruido de alarmas:** Pocos umbrales bien pensados son mejores que decenas de alarmas que nadie mira.
- **Correlacionar señales:** Un pico de tráfico *y* un aumento de `REJECT` puede indicar un ataque o una mala configuración.
- **Definir ventanas de tiempo adecuadas:** Evaluar promedios o sumas en intervalos de 5-15 minutos suele ser más estable que mirar cada minuto.
- **Respetar el Free Tier:** Configurar solo las alarmas y logs necesarios, evitando ingestión masiva innecesaria.

3 Requisitos Previos

3.1 Conocimientos Necesarios

- Conceptos básicos de:
  - VPC, subredes, tablas de ruteo.
  - Instancias EC2 y Security Groups.
- Familiaridad con:
  - Consola de AWS.
  - Conceptos de métrica, log y alerta.

3.2 Infraestructura Base

Idealmente, el estudiante debe disponer de:

- Al menos una VPC de laboratorios anteriores (por ejemplo, `VPC-A-Lab6`).
- Una o dos instancias EC2 de prueba que generen algo de tráfico de red.

Si no existen, se pueden reutilizar los pasos de otros labs para lanzar una instancia simple de prueba.

3.3 Recursos Técnicos

- Cuenta AWS activa (Free Tier).
- Usuario IAM con permisos sobre:
  - CloudWatch (métricas, logs, alarmas, dashboards).
  - VPC (para crear VPC Flow Logs).
  - SNS (para notificaciones de alarmas).
- Navegador web moderno.

3.4 Costos Estimados

Concepto	**Costo Estimado**
Hasta 10 métricas personalizadas	\$0.00 (Free Tier)
Hasta 10 alarmas CloudWatch	\$0.00 (Free Tier)
VPC Flow Logs (bajo volumen de tráfico)	\$0.00 – costo muy bajo (lab corto)
CloudWatch Logs Insights (pocas consultas)	\$0.00 – Free Tier / costo despreciable
TOTAL	**\$0.00**

Cuadro 1: Costos del Laboratorio 7

Nota: Aunque VPC Flow Logs y Logs Insights pueden generar costos en escenarios de alta carga, en este laboratorio se limita el uso a una ventana de tiempo acotada y a pocas consultas, manteniéndose en la práctica dentro del **Free Tier**.

3.5 Tiempo Estimado

- Lectura del marco teórico: 20 minutos.
- Habilitar VPC Flow Logs y CloudWatch Logs: 20 minutos.
- Crear métricas y alarmas: 25 minutos.
- Construir dashboard y ejecutar consultas en Logs Insights: 25 minutos.
- **TOTAL ESTIMADO:** 90 minutos.

4 Procedimiento Paso a Paso

El siguiente procedimiento construye un sistema básico de monitoreo de red usando CloudWatch en torno a una VPC de laboratorio.

4.1 Paso 1: Identificar la VPC a Monitorear

Objetivo: Seleccionar la VPC sobre la cual habilitaremos VPC Flow Logs.

1. En la consola de AWS, ir a **VPC**.
2. En el panel izquierdo, hacer clic en **Your VPCs**.
3. Identificar la VPC que se usará para el laboratorio. Por ejemplo:
   - **Name**: `VPC-A-Lab6`.
   - **CIDR**: `10.10.0.0/16`.
4. Anotar el **VPC ID**, por ejemplo `vpc-0abc123def456`.

4.2 Paso 2: Crear un Log Group para VPC Flow Logs

Objetivo: Definir el destino en CloudWatch Logs donde llegarán los VPC Flow Logs.

1. Ir al servicio **CloudWatch**.
2. En el panel izquierdo, seleccionar **Logs** → **Log groups**.
3. Hacer clic en **Create log group**.
4. Completar:
   - **Log group name:** `/aws/vpc/flow-logs/lab7`.
   - **Retention:** por ejemplo, 7 días (para laboratorio).
5. Crear el log group.

4.3 Paso 3: Crear un Role IAM (si es necesario)

En muchas cuentas, el asistente de VPC Flow Logs crea el role automáticamente. Si no existe:

1. Ir al servicio **IAM**.
2. Crear un **role** para el servicio **VPC Flow Logs** con permisos para escribir en CloudWatch Logs.
3. Nombre sugerido: `VPCFlowLogs-CloudWatchRole-Lab7`.

(En caso de usar el wizard, AWS puede crear una política y role por defecto.)

4.4 Paso 4: Habilitar VPC Flow Logs para la VPC

Objetivo: Activar el registro de tráfico para la VPC seleccionada.

1. Volver al servicio **VPC**.
2. En **Your VPCs**, seleccionar la VPC (ej. `VPC-A-Lab6`).
3. En el panel inferior, ir a la pestaña **Flow Logs**.
4. Hacer clic en **Create flow log**.
5. Parámetros recomendados:
   - **Filter:** `ALL` (captura ACCEPT y REJECT).
   - **Destination:** `Send to CloudWatch Logs`.
   - **Destination log group:** seleccionar `/aws/vpc/flow-logs/lab7`.
   - **IAM role:** seleccionar el role creado o sugerido.
   - **Maximum aggregation interval:** 1 minuto (para ver eventos más pronto).
6. Crear el Flow Log.

4.5 Paso 5: Generar Tráfico de Red de Prueba

Objetivo: Asegurar que el Flow Log capture tráfico útil.

1. Conectarse a una instancia EC2 dentro de la VPC mediante SSH.
2. Generar tráfico:
   - **Tráfico permitido:**
     - Hacer ping a otra instancia dentro de la VPC o a un servidor de prueba permitido.
   - **Tráfico rechazado:**
     - Intentar conectarse a un puerto bloqueado o a un destino no permitido por el Security Group.
3. Esperar unos minutos para que los eventos aparezcan en CloudWatch Logs.

4.6 Paso 6: Visualizar VPC Flow Logs en CloudWatch Logs

Objetivo: Confirmar que los eventos están siendo registrados.

1. Ir a **CloudWatch** → **Logs** → **Log groups**.
2. Seleccionar `/aws/vpc/flow-logs/lab7`.
3. Abrir uno de los **log streams**.
4. Verificar que aparecen líneas con campos como `srcaddr`, `dstaddr`, `action`, `bytes`, etc.

4.7 Paso 7: Crear una Métrica Derivada (Metric Filter) para Tráfico REJECT

Objetivo: Contar conexiones rechazadas usando una métrica de CloudWatch.

1. Dentro del log group `/aws/vpc/flow-logs/lab7`, ir a la pestaña **Metric filters**.
2. Hacer clic en **Create metric filter**.
3. En **Filter pattern**, usar un patrón simple, por ejemplo:

```
REJECT
```

4. Hacer clic en **Next**.
5. Asignar:
   - **Filter name:** `Lab7-Rejected-Connections`.
   - **Metric namespace:** `Lab7/Network`.
   - **Metric name:** `RejectedConnections`.
   - **Metric value:** `1`.
   - **Default value:** `0` (opcional).
6. Guardar el metric filter.

Esta métrica incrementará en 1 por cada línea de log que contenga `REJECT`.

4.8 Paso 8: Crear una Alarma para Conexiones Rechazadas

Objetivo: Generar una alerta cuando haya un número inusual de rechazos (potencialmente indicando un problema de configuración o intento de ataque).

1. Ir a **CloudWatch** → **Alarms** → **All alarms**.
2. Hacer clic en **Create alarm**.
3. Hacer clic en **Select metric**.
4. Navegar a:
   - **Custom namespaces** → `Lab7/Network`.
   - Seleccionar `RejectedConnections`.
5. Hacer clic en **Select metric**.
6. Configurar la alarma:
   - **Statistic:** Sum.
   - **Period:** 5 minutes.

- **Condition:** `Greater than or equal to`.
- **Threshold value:** por ejemplo, `10` (10 rechazos en 5 minutos).

7. Hacer clic en **Next**.
8. Configurar notificación:
   - Crear o usar un **SNS topic** (por ejemplo, `Lab7-Alerts`).
   - Ingresar un correo electrónico para recibir notificaciones.
9. Asignar nombre a la alarma:
   - **Alarm name:** `Lab7-HighRejectedConnections`.
10. Revisar y crear la alarma.

4.9 Paso 9: Crear un Dashboard de Monitoreo de Red

Objetivo: Visualizar en un solo lugar métricas relevantes del laboratorio.

1. Ir a **CloudWatch** → **Dashboards**.
2. Hacer clic en **Create dashboard**.
3. Nombre sugerido: `Lab7-Network-Monitoring`.
4. Seleccionar tipo de widget **Line** para gráficos de serie temporal.
5. Agregar:
   - **Widget 1:** Tráfico de red de una instancia:
     - Namespace: `AWS/EC2`.
     - Métricas: `NetworkIn` y `NetworkOut` de la instancia de prueba.
     - Título: `EC2 Network In/Out`.
   - **Widget 2:** Conexiones rechazadas:
     - Namespace: `Lab7/Network`.
     - Métrica: `RejectedConnections`.
     - Título: `Conexiones REJECT por período`.
   - **Widget 3:** Estado de alarmas:
     - Tipo de widget: **Alarm status**.
     - Seleccionar `Lab7-HighRejectedConnections`.
6. Ajustar el rango de tiempo del dashboard (por ejemplo, última 1 hora).

4.10 Paso 10: Consultas Básicas en CloudWatch Logs Insights

Objetivo: Usar Logs Insights para analizar los VPC Flow Logs.

1. Ir a **CloudWatch** → **Logs** → **Logs Insights**.
2. Seleccionar el log group `/aws/vpc/flow-logs/lab7`.
3. Establecer el rango de tiempo (últimos 30 minutos, por ejemplo).
4. Ejecutar las siguientes consultas como ejemplo:

Consulta 1: Conteo de tráfico ACCEPT vs REJECT

```
fields action
| stats count(*) as total by action
```

Listing 2: ACCEPT vs REJECT

Consulta 2: IPs con más tráfico rechazado

```
fields srcaddr, action
| filter action = 'REJECT'
| stats count(*) as rechazos by srcaddr
| sort rechazos desc
| limit 10
```

Listing 3: IPs con mayor número de REJECT

Consulta 3: Puertos de destino más usados

```
fields dstport
| stats count(*) as total by dstport
| sort total desc
| limit 10
```

Listing 4: Puertos más frecuentes

5. Observar los resultados y relacionarlos con el tráfico generado en el laboratorio.

5 Tablas de Configuración

5.1 Resumen de Recursos de Monitoreo

Recurso	Nombre	Descripción
Log Group	/aws/vpc/flow-logs/lab7	Almacena VPC Flow Logs de la VPC del lab
Metric Filter	Lab7-Rejected-Connections	Cuenta líneas con "REJECT" en los flow logs
Namespace Métrica	Lab7/Network	Namespace personalizado para métricas del lab
Métrica	RejectedConnections	Número de conexiones rechazadas por período
Alarma	Lab7-HighRejectedConnections	Alerta si hay muchos REJECT en poco tiempo
Dashboard	Lab7-Network-Monitoring	Panel de monitoreo de red del lab

Cuadro 2: Recursos principales de CloudWatch en el Laboratorio 7

5.2 Parámetros de Flow Logs

Parámetro	Valor
Resource type	VPC
Filter	ALL
Destination	CloudWatch Logs
Log group	/aws/vpc/flow-logs/lab7
Aggregation interval	1 minute

Cuadro 3: Parámetros recomendados para VPC Flow Logs

5.3 Configuración de la Alarma Principal

Propiedad	Valor
Namespace	Lab7/Network
Métrica	RejectedConnections
Statistic	Sum
Period	300 s (5 minutos)
Condición	Greater or equal
Umbral	10
Acción	Enviar notificación a SNS (Lab7-Alerts)

Cuadro 4: Configuración de la alarma Lab7-HighRejectedConnections

6 Verificación

6.1 Verificación de VPC Flow Logs

1. Confirmar que en el panel **Flow Logs** de la VPC el estado del flow log aparece como **Active**.
2. Revisar algún log stream en `/aws/vpc/flow-logs/lab7` para verificar que se siguen generando eventos.

6.2 Verificación de la Métrica RejectedConnections

1. Ir a **CloudWatch** → **Metrics**.
2. Seleccionar el namespace `Lab7/Network`.
3. Localizar `RejectedConnections`.
4. Visualizar la gráfica en los últimos 15–30 minutos.
5. Generar de nuevo algunos intentos de conexión que resulten en `REJECT` y observar el incremento de la métrica.

6.3 Verificación de la Alarma

1. Revisar la alarma `Lab7-HighRejectedConnections`.
2. Verificar que su estado es:
   - **OK**, si no se ha superado el umbral.
   - **ALARM**, si se han generado suficientes rechazos.
3. Si se ha configurado SNS, revisar el correo para confirmar la recepción de la notificación cuando la alarma entra en estado `ALARM`.

6.4 Verificación del Dashboard

1. Abrir el dashboard `Lab7-Network-Monitoring`.
2. Comprobar que:
   - Se ven las métricas de red de la instancia (NetworkIn/NetworkOut).
   - Se ve la curva de `RejectedConnections`.
   - El widget de estado de alarmas refleja correctamente el estado de `Lab7-HighRejectedConnectio`

7 Limpieza de Recursos

Objetivo: Evitar costos innecesarios y dejar el entorno ordenado después del laboratorio.

7.1 Pasos de Limpieza

1. **Instancias EC2 de prueba**
   - Si se crearon instancias sólo para este laboratorio, detenerlas y terminarlas.
2. **Alarmas**
   - Ir a **CloudWatch** → **Alarms**.
   - Seleccionar `Lab7-HighRejectedConnections` (y cualquier otra alarma creada).
   - Hacer clic en **Actions** → **Delete**.
3. **Metric filters y Log groups**
   - En **CloudWatch** → **Logs**, seleccionar el log group `/aws/vpc/flow-logs/lab7`.
   - Eliminar metric filters asociados si el entorno no se va a reutilizar.
   - Si el laboratorio ha terminado y no se requiere retener logs, eliminar el log group.
4. **Flow Logs**
   - Volver al servicio **VPC**.
   - En la VPC monitoreada, ir a la pestaña **Flow Logs**.
   - Seleccionar el flow log creado y eliminarlo para detener la captura de tráfico.
5. **SNS Topic (opcional)**
   - En **SNS**, eliminar el topic `Lab7-Alerts` si fue creado solo para este laboratorio.

7.2 Comando CLI Opcional para Ver Alarmas

```
aws cloudwatch describe-alarms \
  --query 'MetricAlarms[*].[AlarmName,StateValue,MetricName,
     Namespace]' \
  --output table
```

Listing 5: Listar alarmas de CloudWatch

8 Cuestionario de Evaluación

Instrucciones: Responde las siguientes preguntas. Las respuestas sugeridas se encuentran al final de la sección.

8.1 Preguntas de Selección Múltiple

1. **¿Cuál es el propósito principal de Amazon CloudWatch?**

   a) Almacenar objetos estáticos como imágenes y videos.

   b) Ofrecer bases de datos relacionales escalables.

   c) Proveer monitoreo, métricas y logs de recursos en AWS.

   d) Gestionar identidades y accesos de usuarios.

2. **¿Qué información típica proveen los VPC Flow Logs?**

   a) Solo el uso de CPU en instancias EC2.

   b) IP de origen y destino, puertos, protocolo, acción (ACCEPT/REJECT), bytes y paquetes.

   c) Únicamente latencias de red entre regiones.

   d) Número total de instancias activas en la VPC.

3. **¿Hacia dónde se pueden enviar los VPC Flow Logs?**

   a) Únicamente a Amazon S3.

   b) Únicamente a CloudWatch Logs.

   c) A CloudWatch Logs o a Amazon S3.

   d) Solo a una base de datos RDS.

4. **En el contexto de este laboratorio, la métrica personalizada `RejectedConnections` se obtiene a partir de:**

   a) El número de conexiones exitosas aceptadas por la VPC.

   b) Un metric filter que cuenta líneas con “REJECT” en los VPC Flow Logs.

   c) El tráfico total (en bytes) de la VPC.

   d) El promedio de latencia de red entre instancias.

5. **¿Cuál de las siguientes afirmaciones sobre las alarmas de CloudWatch es correcta?**

   a) No pueden enviar notificaciones, solo registrar el estado.

   b) Solo pueden basarse en métricas de CPU y memoria.

   c) Cambian de estado (OK/ALARM/INSUFFICIENT_DATA) según la evaluación de una métrica.

d) Solo funcionan con servicios de red.

6. **En el Free Tier de CloudWatch, típicamente se dispone de:**

   a) Hasta 10 alarmas y 10 métricas personalizadas sin costo adicional.

   b) Alarmas ilimitadas sin costo.

   c) Cero métricas gratuitas, todo se cobra desde el inicio.

   d) Sólo 1 alarma gratuita.

7. **¿Qué componente de CloudWatch permite construir paneles visuales personalizados?**

   a) CloudWatch Logs Insights.

   b) CloudWatch Dashboards.

   c) CloudWatch Metrics Engine.

   d) VPC Flow Logs.

8. **¿Qué permite hacer CloudWatch Logs Insights?**

   a) Crear y administrar VPCs.

   b) Ejecutar consultas sobre logs para analizarlos, filtrarlos y agregarlos.

   c) Configurar usuarios IAM.

   d) Crear buckets de S3 automáticamente.

9. **Si se desea detectar un posible ataque mediante un gran número de conexiones rechazadas, una estrategia razonable es:**

   a) Ignorar los VPC Flow Logs y monitorear solo CPU.

   b) Crear una métrica derivada de logs que cuente las entradas REJECT y una alarma sobre esa métrica.

   c) Deshabilitar todos los Security Groups.

   d) Asignar IPs públicas a todas las instancias.

10. **¿Cuál de las siguientes es una buena práctica de monitoreo?**

    a) Crear cientos de alarmas con umbrales poco claros.

    b) No monitorear nada mientras el sistema "parezca" funcionar.

    c) Elegir un conjunto limitado de métricas clave y definir umbrales significativos.

    d) Usar únicamente métricas de CPU para evaluar la salud de la red.

8.2 Preguntas Verdadero/Falso

VF1. Los VPC Flow Logs permiten ver tanto tráfico aceptado como rechazado, dependiendo del filtro configurado.

VF2. Las alarmas de CloudWatch pueden enviarse a un tópico de SNS para generar notificaciones por correo.

VF3. CloudWatch Logs Insights solo funciona con logs de aplicaciones, pero no con VPC Flow Logs.

8.3 Escenarios Prácticos

E1. Escenario 1: Pico inusual de tráfico rechazado

Durante la noche, la alarma `Lab7-HighRejectedConnections` se dispara varias veces. En el dashboard se observa un pico en `RejectedConnections`, pero el tráfico `NetworkIn/Out` no ha aumentado de forma proporcional.

Pregunta: ¿Qué pasos seguirías para investigar este comportamiento usando VPC Flow Logs y Logs Insights? Menciona al menos dos consultas o análisis que realizarías.

E2. Escenario 2: Diseño de un dashboard de red para producción

Una empresa despliega aplicaciones críticas en múltiples instancias EC2 detrás de un Load Balancer. Desean un dashboard de red que les permita detectar rápidamente:

- Picos de tráfico.
- Errores de red (4xx/5xx).
- Aumentos en conexiones rechazadas.

Pregunta: ¿Qué métricas y widgets incluirías en el dashboard? ¿Qué alarmas clave definirías para complementar ese dashboard?

8.4 Respuestas del Cuestionario

Selección Múltiple

1. **c)** Proveer monitoreo, métricas y logs de recursos en AWS.
2. **b)** IP de origen y destino, puertos, protocolo, acción (ACCEPT/REJECT), bytes y paquetes.
3. **c)** A CloudWatch Logs o a Amazon S3.
4. **b)** Un metric filter que cuenta líneas con "REJECT" en los VPC Flow Logs.
5. **c)** Cambian de estado (OK/ALARM/INSUFFICIENT_DATA) según la evaluación de una métrica.
6. **a)** Hasta 10 alarmas y 10 métricas personalizadas sin costo adicional (según condiciones del Free Tier).
7. **b)** CloudWatch Dashboards.
8. **b)** Ejecutar consultas sobre logs para analizarlos, filtrarlos y agregarlos.
9. **b)** Crear una métrica derivada de logs que cuente las entradas REJECT y una alarma sobre esa métrica.
10. **c)** Elegir un conjunto limitado de métricas clave y definir umbrales significativos.

Verdadero/Falso

1. **Verdadero.** El filtro `ALL` captura ACCEPT y REJECT; también existen filtros `ACCEPT` y `REJECT`.
2. **Verdadero.** SNS es una de las integraciones más comunes para notificaciones de alarmas.
3. **Falso.** Logs Insights puede trabajar con cualquier log group de CloudWatch, incluyendo VPC Flow Logs.

Guía para Escenarios

Escenario 1:

- Revisar en Logs Insights cuáles IP de origen generan más `REJECT`:

```
fields srcaddr, action
| filter action = 'REJECT'
| stats count(*) as rechazos by srcaddr
| sort rechazos desc
| limit 20
```

- Analizar puertos de destino para ver si se trata de escaneo de puertos:

```
fields dstport, action
| filter action = 'REJECT'
| stats count(*) as rechazos by dstport
| sort rechazos desc
| limit 20
```

- Correlacionar el horario de los picos con cambios en Security Groups o despliegues recientes.

Escenario 2:

- Métricas en el dashboard:
  - `NetworkIn/NetworkOut` de instancias y/o Load Balancer.
  - `RequestCount`, `HTTPCode_ELB_4XX`, `HTTPCode_ELB_5XX`.
  - Métrica derivada de `RejectedConnections` vía VPC Flow Logs.
- Widgets:
  - Gráficas de línea de tráfico total.
  - Gráficas de errores HTTP 4xx/5xx.
  - Gráfico de `RejectedConnections`.
  - Widget de estado de alarmas críticas.
- Alarmas:
  - Alarma por tasa alta de errores 5xx.
  - Alarma por pico de tráfico fuera de rango esperado.
  - Alarma por aumento significativo de `RejectedConnections`.

9 Conclusiones

En este laboratorio, el estudiante ha construido un esquema básico pero completo de **monitoreo de redes** utilizando Amazon CloudWatch como plataforma central de observabilidad. A través de la combinación de **VPC Flow Logs**, **métricas derivadas**, **alarmas** y **dashboards**, se ha demostrado cómo pasar de una red "opaca" a una red **instrumentada** y **observable**.

Los principales logros incluyen:

- Comprender el rol de CloudWatch en la arquitectura de AWS.
- Habilitar y analizar VPC Flow Logs para entender el tráfico permitido y rechazado.
- Derivar métricas significativas (como `RejectedConnections`) a partir de logs de bajo nivel.
- Configurar alarmas efectivas que permiten reaccionar ante comportamientos de red anómalos, manteniéndose dentro de los límites del **Free Tier**.
- Construir un dashboard que integra diferentes señales y facilita la detección visual de problemas.
- Utilizar CloudWatch Logs Insights como herramienta de investigación para responder preguntas específicas sobre el comportamiento de la red.

Más allá de los detalles de implementación, el laboratorio enfatiza la importancia de:

- **Medir constantemente**: la red debe ser observable, no una caja negra.
- **Filtrar el ruido**: las alarmas deben ser pocas pero relevantes.
- **Correlacionar datos**: métricas y logs se complementan para contar la historia completa de lo que ocurre en la infraestructura.

Estas habilidades y buenas prácticas son fundamentales para operar entornos de producción en la nube de forma segura, eficiente y resiliente. El trabajo realizado en este laboratorio prepara el camino para arquitecturas más avanzadas donde el monitoreo se integra con automatización, respuesta a incidentes y prácticas de observabilidad de nivel empresarial.

10 Referencias

10.1 Documentación Oficial de AWS

1. Amazon CloudWatch Documentation
   https://docs.aws.amazon.com/cloudwatch/

2. Using Amazon CloudWatch Metrics
   https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/working_with_metrics.html

3. VPC Flow Logs
   https://docs.aws.amazon.com/vpc/latest/userguide/flow-logs.html

4. CloudWatch Logs Insights
   https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AnalyzingLogData.html

5. Creating Amazon CloudWatch Alarms
   https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/AlarmThatSendsEmail.html

6. CloudWatch Free Tier
   https://aws.amazon.com/cloudwatch/pricing/

10.2 Recursos de Mejores Prácticas

1. AWS Well-Architected Framework - Operational Excellence and Reliability Pillars
   https://aws.amazon.com/architecture/well-architected/

2. AWS Architecture Center
   https://aws.amazon.com/architecture/

10.3 Bibliografía General

1. Jones, S. (2020). *Monitoring and Observability in the Cloud*. Cloud Native Press.

2. Wittig, A., & Wittig, M. (2018). *Amazon Web Services in Action* (2nd ed.). Manning Publications.

Nota: Las URLs fueron verificadas al momento de elaboración de este laboratorio. Se recomienda revisar periódicamente la documentación oficial de AWS para cambios o nuevas funcionalidades relacionadas con CloudWatch y VPC Flow Logs.