[Publish] - 8. Asegurar que las funciones de Amazon Lambda no compartan el mismo rol de ejecución de IAM

El hallazgo de seguridad **"Asegurar que las funciones de Amazon Lambda no compartan el mismo rol de ejecución de IAM"** es una aplicación directa del principio de mínimo privilegio en un entorno sin servidor (serverless). Cuando varias funciones Lambda, cada una con una tarea distinta, comparten un mismo rol de IAM, ese rol tiende a acumular los permisos necesarios para *todas* esas funciones.

Esto crea un riesgo de seguridad significativo:

- **Escalada de Privilegios:** Si una de las funciones Lambda (por ejemplo, una que procesa imágenes) tiene una vulnerabilidad y es comprometida, el atacante obtiene los permisos del rol compartido. Si ese rol también tiene permisos para interactuar con una base de datos (porque otra función lo necesita), el atacante puede usar esos permisos para acceder o modificar datos sensibles, aunque la función comprometida originalmente no tuviera esa intención.
- **Dificultad para Auditar:** Es mucho más difícil determinar qué función fue responsable de una acción específica en CloudTrail si todas comparten la misma identidad (rol).

La mejor práctica es que **cada función Lambda tenga su propio rol de ejecución de IAM** con una política de permisos ajustada específicamente a las necesidades de esa única función.

Análisis de las Funciones Lambda Reportadas

Las siguientes funciones Lambda han sido identificadas, y es probable que algunas de ellas estén compartiendo roles, especialmente aquellas con nombres similares que operan en diferentes entornos (DEV, QA, CAR).

Tipo	Nombre de la Función Lambda	Número de Cuenta
Lambda	dev-gruposanchezagent2025pocgenia-InvokeAgentLambda	174771279728
Lambda	dev-gruposanchezagent2025pocgenia-	174771279728

	InvokeAgentLambda_py11	
Lambda	ebs-snapper-[...]	398984081329
Lambda	Encender_EC2	613703092215
Lambda	Encender_Instancias	002670977573
Lambda	Encender_Produccion1	002670977573
Lambda	Modificar_Instancia	398984081329
Lambda	Modify-Instance	398984081329
Lambda	Start_EC2_BO_CAR	398984081329
Lambda	Start_EC2_BO_CAR	613703092215
Lambda	Start_EC2_DEV	613703092215
Lambda	start_EC2_DEV-CAR-ECC	613703092215
Lambda	start_ec2_instance	398984081329
Lambda	Start_EC2_QA	613703092215
Lambda	Stop_EC2_BO_CAR	398984081329
Lambda	Stop_EC2_BO_CAR	613703092215
Lambda	Stop_EC2_DEV	613703092215
Lambda	Stop_EC2_DEV-CAR-ECC	613703092215
Lambda	Stop_EC2_QA	613703092215
Lambda	Stop_EC2_SLT_PO_DEV	398984081329
Lambda	Stop_EC2_SLT_PO_DEV	613703092215

Observación Clave: Es muy probable que funciones como `Start_EC2_DEV`, `Start_EC2_QA` y `Start_EC2_BO_CAR` estén utilizando un único rol genérico como

"rol-para-iniciar-instancias", a pesar de operar en entornos distintos.

Preguntas para tu Equipo (Acción Requerida)

Para remediar este hallazgo, necesitamos entender las necesidades específicas de cada función. Por favor, trabaja con los equipos de desarrollo para responder:

1. **Confirmar Roles Compartidos:** ¿Pueden revisar la configuración de estas funciones (especialmente las de `Start_EC2`, `Stop_EC2`, `Encender_Instancias`) y confirmar cuáles están compartiendo el mismo rol de ejecución?
2. **Definir Permisos Específicos:** Para cada función, ¿cuáles son exactamente los recursos de AWS con los que necesita interactuar? Por ejemplo:
   - La función `Start_EC2_DEV` solo debería tener permiso para `ec2:StartInstances` sobre las instancias EC2 específicas del entorno de desarrollo (identificadas por etiquetas o ARN), y no sobre las de QA o producción.
   - La función `ebs-snapper-CreateSnapshotFunction` solo necesita permisos para `ec2:CreateSnapshot` y quizás `ec2:DescribeVolumes`.

Consideraciones de Costo

No hay **ningún costo directo** asociado con la creación de roles o políticas de IAM adicionales. El costo de no seguir esta práctica es un **aumento del riesgo de seguridad**. Aislar los permisos de las funciones Lambda es una de las formas más efectivas y económicas de mejorar la seguridad de tus aplicaciones sin servidor.

Cómo Solucionarlo: Crear y Asignar Roles Dedicados

El proceso implica crear un nuevo rol para cada función (o grupo lógico muy pequeño de funciones) y luego actualizar la configuración de la función Lambda.

1. **Crear un Nuevo Rol de IAM Dedicado:**
   - Navega a la consola de **IAM** y selecciona **Roles**.
   - Haz clic en **Crear rol**.
   - Como caso de uso, selecciona **Lambda**.
   - Adjunta una nueva política de permisos. En lugar de usar una política administrada por AWS, haz clic en **Crear política** para abrir el editor visual o JSON.
   - Define los permisos de mínimo privilegio. Sé lo más específico posible con las acciones y los recursos. Por ejemplo, para la función `Start_EC2_DEV`:

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Action": "ec2:StartInstances",
            "Resource": "arn:aws:ec2:region:ACCOUNT_ID:instance/i-12345dev"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:*"
        }
    ]
}


```

- Dale un nombre descriptivo al rol (ej. `Start_EC2_DEV_Lambda_ExecutionRole`) y créalo.

2. **Asignar el Nuevo Rol a la Función Lambda:**
   - Navega a la consola de **Lambda**.
   - Selecciona la función que deseas actualizar (ej. `Start_EC2_DEV`).
   - Ve a la pestaña **Configuración** y luego a la sección **Permisos**.
   - Haz clic en **Editar**.
   - En **Rol de ejecución**, selecciona **Usar un rol existente** y elige el nuevo rol dedicado que creaste en el paso anterior.
   - Guarda los cambios.
3. **Probar y Eliminar Roles Antiguos:**
   - Invoca la función Lambda para asegurarte de que sigue funcionando correctamente con los nuevos permisos restringidos.
   - Una vez que todas las funciones hayan sido migradas a sus nuevos roles dedicados, puedes eliminar el rol compartido antiguo si ya no está en uso.

Recomendación

Se recomienda que se adopte la práctica de "un rol por función" como un estándar para todos los desarrollos de Lambda. Inicia el proceso de refactorización para las funciones existentes, priorizando aquellas en entornos de producción y las que manejan los datos más sensibles. Esto fortalecerá significativamente la seguridad de tu arquitectura sin servidor al limitar el impacto potencial de una función comprometida.

Referencias

Defining Lambda function permissions with an execution role - AWS Lambda