



MÁSTER EN CIBERSEGURIDAD CON DELOITTE

# CIBERSEGURIDAD EN LA NUBE: ESQUEMA SIEM DISTRIBUÍDO EN AWS

TFM elaborado por: Santiago García Arango

Tutor de TFM: Jonay Arón Puente Luis

Septiembre 2024

## **DEDICATORIA**

A mi familia, por su apoyo incondicional. Adriana y George, con su amor, acompañamiento y ejemplo, me han inspirado a esforzarme cada día y luchar por mis sueños.

A Mariana y Óscar, por su disciplina, madurez y valiosas enseñanzas. Gracias por demostrarme que, con determinación y constancia, todo lo que uno se propone es posible.

A Kike, Papo y Lilo, por su experiencia y apoyo constante durante estos dos años de estudio, desvelos y esfuerzo.

A Moni, por su amor y compromiso. Gracias por ser mi compañera de vida y de viaje. Estos dos años de estudio no habrían sido los mismos sin tu inspiración y aliento para dar lo mejor de mí cada día.

Y a todas las personas apasionadas por aprender algo nuevo cada día. Que la chispa de curiosidad siga intacta y nunca se apague, pues es el motor que realmente mueve al mundo.

## RESUMEN

Este proyecto de máster tiene como objetivo implementar un sistema SIEM distribuido sobre arquitecturas en la nube de Amazon Web Services (AWS), empleando Wazuh como herramienta principal de monitoreo, protección y seguridad. A través de la adopción de las mejores prácticas de seguridad en la nube, se diseñó un esquema organizacional de múltiples cuentas en AWS para separar áreas críticas del negocio, respaldado por una infraestructura de red segura y adaptable. El proyecto incluyó el despliegue de múltiples flujos de trabajo simulando cargas de trabajo empresariales y su monitoreo mediante el SIEM, con configuraciones personalizadas y reglas avanzadas para detectar y responder a ciberataques. La solución se implementó siguiendo el enfoque de "Infraestructura como Código" usando AWS CDK, garantizando una administración eficiente y escalable. Se llevaron a cabo múltiples escenarios que validaron la capacidad del sistema para identificar vulnerabilidades y generar respuestas proactivas ante amenazas, cumpliendo así con los objetivos establecidos.

**Palabras clave:** AWS, SIEM, Wazuh, seguridad en la nube, AWS CDK, ciberseguridad, Infraestructura como Código.

## ABSTRACT

This master's project aims to implement a distributed SIEM system on Amazon Web Services (AWS) cloud architectures, using Wazuh as the primary tool for monitoring, protection, and security. Through the adoption of cloud security best practices, a multi-account organizational setup in AWS was designed to separate critical business areas, supported by a secure and adaptable network infrastructure. The project included deploying multiple workflows simulating business workloads and monitoring them through the SIEM, with customized configurations and advanced rules to detect and respond to cyberattacks. The solution was implemented following an "Infrastructure as Code" approach using AWS CDK, ensuring efficient and scalable management. Multiple scenarios were conducted to validate the system's ability to identify vulnerabilities and generate proactive responses to threats, thus fulfilling the established objectives.

**Keywords:** AWS, SIEM, Wazuh, cloud security, AWS CDK, cybersecurity, Infrastructure as Code.

# CONTENIDO

DEDICATORIA.....	2
RESUMEN .....	3
ABSTRACT.....	4
CONTENIDO.....	5
LISTA DE FIGURAS.....	6
LISTA DE TABLAS .....	10
CONTEXTO Y PROBLEMÁTICA POR RESOLVER .....	11
OBJETIVOS.....	12
1.    Objetivo General .....	12
2.    Objetivos Específicos .....	12
MARCO DE REFERENCIA.....	13
1.    Computación en la Nube .....	13
2.    Seguridad en la Nube.....	15
3.    Tecnologías SIEM .....	17
METODOLOGÍA .....	19
1.    Identificación de Requisitos .....	19
2.    Casa de la Calidad.....	19
3.    Matriz Morfológica.....	21
4.    Arquitectura General del Sistema .....	25
DESARROLLO DEL PROYECTO .....	27
1.    Esquema Organizacional en AWS.....	27
2.    Creación de Redes e Infraestructura de Comunicaciones.....	39
3.    Despliegue de Flujos de Trabajo (Servidores) .....	45
4.    Implementación y Configuración del SIEM .....	48
CONCLUSIONES.....	62
REFERENCIAS .....	63

## LISTA DE FIGURAS

Ilustración 1. Mercado de Computación en la Nube en Estados Unidos. Tomado de (Fortune BI, 2024).....	13
Ilustración 2. Cuadrante Mágico de Proveedores de Nube 2023. Tomado de (Gartner, 2024) .....	14
Ilustración 3. Modelo de Responsabilidad Compartida para la Nube de AWS. Tomado de (Amazon Web Services, 2024). .....	15
Ilustración 4. Ejemplo del uso de AWS Organizations para control de múltiples cuentas de AWS. Tomado de (Amazon Web Services, 2024) .....	16
Ilustración 5. Ciberataques promedio por empresa entre 2021 y 2024. Tomado de (Check Point, 2024).....	17
Ilustración 6. Casa de la Calidad para el sistema SIEM. Elaboración propia.....	20
Ilustración 7. AWS Logo. Tomado de (AWS, 2024).....	21
Ilustración 8. Azure Logo. Tomado de (Microsoft, 2024).....	21
Ilustración 9. GCP Logo. Tomado de (Google, 2024).....	22
Ilustración 10. Splunk Logo. Tomado de (Splunk, 2024).....	22
Ilustración 11. IBM QRadar Logo. Tomado de (IBM, s.f.) .....	22
Ilustración 12. Wazuh Logo. Tomado de (Wazuh, 2024).....	23
Ilustración 13. Terraform Logo. Tomado de (Hashicorp, 2024) .....	23
Ilustración 14. CDK Logo. Tomado de (AWS, 2024). .....	23
Ilustración 15. CloudFormation Logo. Tomado de (AWS, 2024).....	24
Ilustración 16. Arquitectura general del sistema SIEM en AWS. Elaboración propia. ....	26
Ilustración 17. Imagen de mi proyecto Open-Source para la elaboración del trabajo de grado. Elaboración propia. .....	27
Ilustración 18. Ejemplo de múltiples cuentas de AWS sin ningún tipo de jerarquía ni esquema organizacional. Elaboración propia.....	28
Ilustración 19. Ejemplo de AWS Organizations con los componentes principales. Tomado de (AWS, 2024).....	29
Ilustración 20. Esquema organizacional en AWS para mi proyecto de máster. Elaboración propia. .....	30
Ilustración 21. Repositorio propio para el esquema organizacional del proyecto de máster. Elaboración propia. .....	31

Ilustración 22. Proceso de despliegue con Cloud Development Kit (CDK). Elaboración propia.....	32
Ilustración 23. Explicación detallada del repositorio propio para el despliegue organizacional. Elaboración propia. ....	32
Ilustración 24. Pipeline de Despliegue Automático para el esquema organizacional. Elaboración propia. ....	33
Ilustración 25. Despliegue automatizado de estructura organizacional propia. Elaboración propia. ....	33
Ilustración 26. Resultado del Single Sign On de mi organización. Elaboración propia. ....	34
Ilustración 27. Estructura organizacional para usuario admin con acceso a todas las cuentas. Elaboración propia.....	34
Ilustración 28. Proceso de despliegue de instancia permitida "t2.micro". Elaboración propia. ....	35
Ilustración 29. Resultado despliegue de instancia permitida. Elaboración propia....	36
Ilustración 30. Proceso de despliegue de instancia permitida "t2.2xlarge". Elaboración propia .....	36
Ilustración 31. Resultado de despliegue fallido para instancias no permitidas en la estructura organizacional. Elaboración propia. ....	37
Ilustración 32. Despliegue fallido de RDS debido a control preventivo en regiones no permitidas. Elaboración propia. ....	37
Ilustración 33. Esquema organizacional exitoso desde el panel de control de AWS Identity Center. Elaboración propia.....	38
Ilustración 34. Ejemplo básico de estructura de red con VPC, Subnets y Local Zone. Tomado de (AWS, 2024).....	40
Ilustración 35. Esquema de red en AWS con VPC para proyecto de máster. Elaboración propia. ....	41
Ilustración 36. Explicación detallada del repositorio propio para el despliegue de Redes. Elaboración propia.....	42
Ilustración 37. Configuraciones para esquema de Red con CDK. Elaboración propia. ....	42
Ilustración 38. Resultados de la red desplegada en AWS con IaC. Elaboración propia. ....	43
Ilustración 39. Despliegue de servidor en red pública. Elaboración propia.....	43

Ilustración 40. Respuesta correcta de conectividad a internet (google.com) desde red pública. Elaboración propia.....	44
Ilustración 41. Esquema de servidores simulando flujo de trabajo en esquema híbrido con múltiples cuentas de AWS y On-Premise. Elaboración propia.....	45
Ilustración 42. Explicación detallada del repositorio propio para el despliegue de servidores demo. Elaboración propia.....	46
Ilustración 43. Configuraciones para servidores demo con CDK. Elaboración propia.....	47
Ilustración 44. Servidores demo para el trabajo de máster simulando flujos productivos en AWS. Elaboración propia.....	47
Ilustración 45. Componentes del SIEM Wazuh. Tomado de (Wazuh, 2024).....	48
Ilustración 46. Imagen de mi proyecto SIEM Open-Source para la elaboración del trabajo de grado. Elaboración propia.....	49
Ilustración 47. Configuración de DNS multi cuenta para AWS con mi dominio san99tiago.com . Elaboración propia.....	51
Ilustración 48. Arquitectura en AWS del sistema SIEM basado en Wazuh con DNS personalizado. Elaboración propia.....	51
Ilustración 49. Explicación detallada del repositorio propio para el despliegue del SIEM en AWS. Elaboración propia.....	52
Ilustración 50. Configuraciones para despliegue del SIEM con CDK. Elaboración propia.....	53
Ilustración 51. Resultado del despliegue del Stack de CloudFormation del SIEM en AWS. Elaboración propia.....	53
Ilustración 52. Resultado del Dashboard de Wazuh (SIEM) en mi endpoint personalizado siem.san99tiago.com . Elaboración propia.....	54
Ilustración 53. Comandos de instalación de agentes para servidores Windows. Elaboración propia.....	55
Ilustración 54. Ejemplo de comandos de instalación de agentes Windows en servidor productivo. Elaboración propia.....	55
Ilustración 55. Scripts de instalación de agentes de Wazuh en servidores Linux de EC2. Elaboración propia.....	56
Ilustración 56. Resultado de sistema SIEM con servidores detectados y protegidos. Elaboración propia.....	56
Ilustración 57. Detección de vulnerabilidades de forma activa en el Dashboard de Wazuh. Elaboración propia.....	57

Ilustración 58. Simulación de ataque a servidor Windows productivo en AWS protegido con SIEM Wazuh. Elaboración propia.....	58
Ilustración 59. Visualización del Dashboard de Wazuh ANTES del ataque de fuerza bruta hacia el servicios Windows. Elaboración propia.....	59
Ilustración 60. Ejecución del ataque (simulado) con herramienta Hydra al servidor Windows productivo. Elaboración propia.....	59
Ilustración 61. Dashboard de Wazuh durante el ataque al servidor Windows. Elaboración propia.....	60
Ilustración 62. Eventos asociados al ataque de fuerza bruta hacia el servidor Windows. Elaboración propia.....	60
Ilustración 63. Medida reactiva al ataque al servidor Windows a través de actualización de Security Groups. Elaboración propia.....	61
Ilustración 64. Dashboard de Wazuh luego de la contención del ataque exitosa. Elaboración propia.....	61

## **LISTA DE TABLAS**

Tabla 1. Requisitos del Sistema SIEM en la Nube. Elaboración propia.....	19
Tabla 2. Alternativas de proveedor de nube. Elaboración propia. ....	22
Tabla 3. Alternativas de herramienta SIEM. Elaboración propia.....	23
Tabla 4. Alternativas de Infraestructura como Código. Elaboración propia. ....	24
Tabla 5. Matriz morfológica para solución SIEM con opciones A, B, C. Elaboración propia.....	24
Tabla 6. Matriz de decisión PUGH. Elaboración propia. ....	25
Tabla 7. Repositorio de GitHub del proyecto (aws-cybersecurity-siem). Elaboración propia. ....	27
Tabla 8. Repositorio de GitHub del proyecto (aws-cdk-organizations-demo). Elaboración propia. ....	31
Tabla 9. Repositorio de GitHub del proyecto SIEM (aws-cybersecurity-siem). Elaboración propia. ....	49

## CONTEXTO Y PROBLEMÁTICA POR RESOLVER

La tecnología ha avanzado en los últimos años de una forma exponencial, convirtiéndose en el pilar más importante que las empresas deben adoptar para mantenerse en la vanguardia. Entre las tendencias más valiosas de los últimos tiempos se encuentran la computación en la nube, la inteligencia artificial, y la ciberseguridad (McKinsey Technology Council, 2024).

Según el “Reporte de Ciberseguridad 2024”, desde el año 2023 se ha registrado un incremento del 90% en los ciberataques, y, al menos 4 de cada 10 empresas no están preparadas para soportar un ataque sofisticado (World Economic Forum, 2024). Esto evidencia una alarmante brecha en las defensas de seguridad cibernética que es crucial cerrar lo antes posible. Sin embargo, el panorama actual se complica aún más debido a la creciente sofisticación de los ciberataques, los cuales emplean tecnologías avanzadas y estrategias cada vez más complejas.

La evolución constante de estas amenazas exige una rápida y eficiente adaptación de las estrategias de protección cibernética en contextos distribuidos para salvaguardar los activos críticos de las organizaciones, como, por ejemplo, sus datos.

Además, según el Cuadrante Mágico de Gartner de 2023 para “Infraestructuras Cloud y Plataformas de Servicios”, Amazon Web Services ha sido la nube líder durante más de diez años consecutivos (Gartner, 2023). Esto permite concluir que es la nube pública más utilizada actualmente para despliegues distribuidos, tanto para grandes empresas, como startups. Por esta razón, es fundamental estudiarla y considerarla en los esquemas de seguridad modernos.

Este trabajo de grado busca proponer un esquema SIEM distribuido enfocado en arquitecturas en la nube de “Amazon Web Services”, que sea resiliente a ataques cibernéticos mediante la implementación de mejores prácticas en redes, infraestructura, observabilidad y despliegues de flujos de trabajo productivos.

# **OBJETIVOS**

En esta sección se presentarán los objetivos del trabajo de grado que se esperan lograr a través del desarrollo del proyecto.

A continuación, se presentan los objetivos generales y específicos que se desarrollarán en el trabajo de grado para obtener el título de “Máster en Ciberseguridad”.

## **1. Objetivo General**

Implementar un esquema SIEM distribuido para arquitecturas en la nube de “Amazon Web Services”, asegurando una protección avanzada contra ciberataques.

## **2. Objetivos Específicos**

1. Crear un esquema organizacional de Amazon Web Services con múltiples cuentas basadas en áreas críticas de un negocio.
2. Diseñar un esquema de redes e infraestructura de comunicaciones compatible con la nube de Amazon Web Services.
3. Desplegar diversos flujos de trabajo con servidores remotos en AWS simulando el cómputo y procesamiento de una empresa.
4. Implementar y configurar un sistema SIEM avanzado en Amazon Web Services para la protección de una empresa.
5. Simular un ciberataque sobre alguno de los flujos de trabajo y validar que el sistema SIEM detecte el incidente automáticamente.

# MARCO DE REFERENCIA

Este trabajo de máster está enfocado en la creación de un sistema SIEM (*Security Information and Event Management*) (Microsoft, 2024), y tiene como objetivo lograr realizar un proyecto exploratorio sobre las mejores prácticas de ciberseguridad enfocadas en la nube.

Es por esto, que en el marco de referencia se mostrará el estado del arte de 3 temáticas importantes para el proyecto: computación en la nube, seguridad en la nube y tecnologías SIEM.

## 1. Computación en la Nube

Actualmente, la computación en la nube se ha convertido en uno de los pilares más relevantes para cualquier industria moderna debido a la posibilidad de proporcionar flexibilidad, escalabilidad, ahorro de costos y agilidad en el diseño de soluciones tecnológicas (Google Cloud Platform, 2024). En un entorno empresarial cada vez más competitivo, el uso de soluciones en la nube permite a las empresas crear, modernizar y adaptar sus infraestructuras digitales, sin la necesidad de administrar servidores, reduciendo significativamente la carga operativa de operaciones y mantenimiento del hardware. Esto lleva a las organizaciones a implementar soluciones digitales cada vez más rápidas, impulsando la agilidad, competitividad y el tiempo de llevar una idea al mercado.

El mercado de la computación en la nube ha crecido exponencialmente en los últimos 10 años, y se proyecta que para el año 2032, tenga una capitalización de mercado de aproximadamente 2.300 billones de USD (Fortune BI, 2024).

North America Cloud Computing Market Size, 2019-2032 (USD Billion)

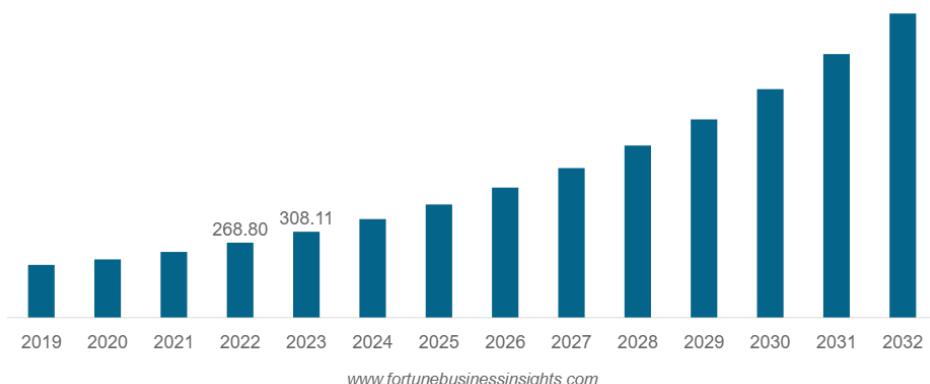


Ilustración 1. Mercado de Computación en la Nube en Estados Unidos. Tomado de (Fortune BI, 2024).

Este crecimiento plantea un desafío importante: cómo implementar las mejores prácticas de ciberseguridad en un entorno de adopción tan acelerada, sin renunciar a los beneficios que ofrece la nube. Lograr esto es complejo, pues la adopción de la nube requiere que las organizaciones tengan entrenamientos, estudios y un plan de trabajo para poder obtener todas las bondades (Rainbow Secure, 2024).

Adicionalmente, hoy en día existen 3 proveedores principales de computación en la nube:

- Amazon Web Services
- Azure
- Google Cloud Platform

Cada una de estas plataformas presenta sus propios desafíos, productos, servicios y enfoques de seguridad, lo que añade complejidad al panorama tecnológico para la adopción de estas. Esto dificulta que una sola persona o equipo pueda mantenerse al día con las últimas innovaciones y asegurar que todas las implementaciones sigan las mejores prácticas de ciberseguridad.

Figure 1: Magic Quadrant for Strategic Cloud Platform Services



Ilustración 2. Cuadrante Mágico de Proveedores de Nube 2023. Tomado de (Gartner, 2024) .

Según los estudios realizados por Gartner, Amazon Web Services es la nube líder en el mercado, seguida por Microsoft (Azure) y Google (Google Cloud Platform) (Gartner, 2024).

Teniendo en cuenta esta métrica, este trabajo de máster será enfocada en Amazon Web Services, pues permitirá replicar las tendencias del mercado y la industria.

## 2. Seguridad en la Nube

Si bien la computación en la nube ofrece numerosos beneficios, como cualquier otra tecnología, debe ser gestionada adecuadamente. El uso de la nube está siempre regido por una premisa clave: el 'Modelo de Responsabilidad Compartida', que define claramente las obligaciones tanto del proveedor como del cliente en términos de seguridad y manejo de los datos. A continuación, se presenta una gráfica ilustrativa de los puntos clave de seguridad compartida enfocados en AWS (Amazon Web Services):

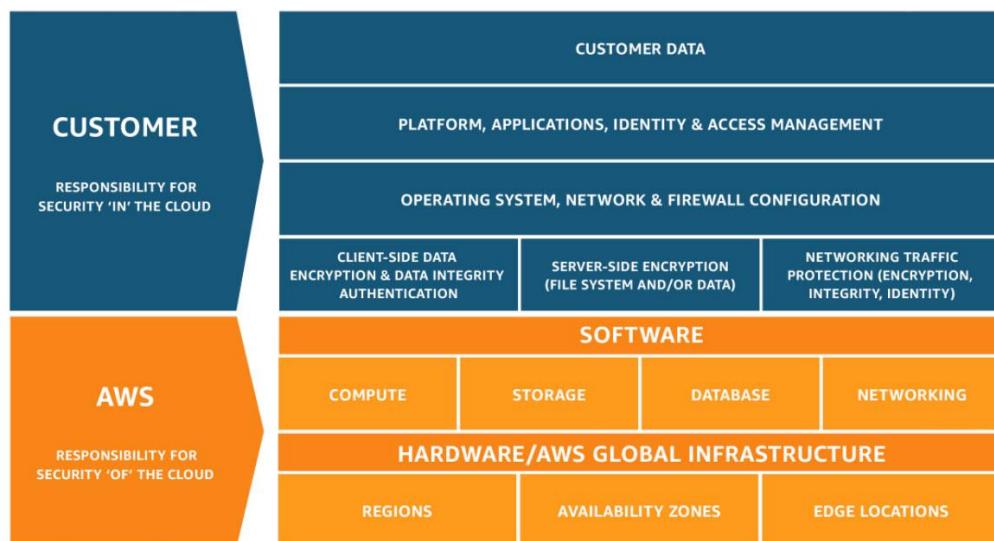


Ilustración 3. Modelo de Responsabilidad Compartida para la Nube de AWS. Tomado de (Amazon Web Services, 2024).

Así como se puede observar en la Ilustración 3, la seguridad en la nube siempre tendrá un esquema compartido, en donde el proveedor (AWS), se encargará de la capa principal de hardware, software e infraestructura en donde corren sus servicios, mientras que las empresas se encargarán de la seguridad de los servidores, roles, permisos, datos y demás (Amazon Web Services, 2024).

Por lo tanto, es crucial que los ingenieros especializados en ciberseguridad estén capacitados para comprender con claridad qué aspectos de los flujos de trabajo son responsabilidad del proveedor de nube y cuáles recaen en la gobernanza de la empresa. Este entendimiento es fundamental para asegurar una protección integral, ya que solo mediante una correcta diferenciación de responsabilidades se puede garantizar que tanto la

infraestructura en la nube como los datos y aplicaciones gestionados por la empresa cumplan con los estándares más altos de seguridad.

Así mismo, a medida que las empresas crecen en tamaño y complejidad de sus servicios, se vuelve muy importante llevar esfuerzos de modernización para estructurar sus estándares de seguridad siguiendo las mejores prácticas. Un ejemplo claro de esta necesidad son las organizaciones que gestionan múltiples cuentas de Amazon Web Services, cada una dedicada a distintos flujos de trabajo, o aquellas que adoptan un enfoque 'Multi-Cloud', utilizando diversas plataformas en la nube para sus procesos. Estas estructuras requieren un enfoque estratégico y bien definido para garantizar la seguridad en todos los niveles. A continuación, hay un ejemplo de cómo se pueden estructurar dichas empresas a nivel tecnológico en AWS:

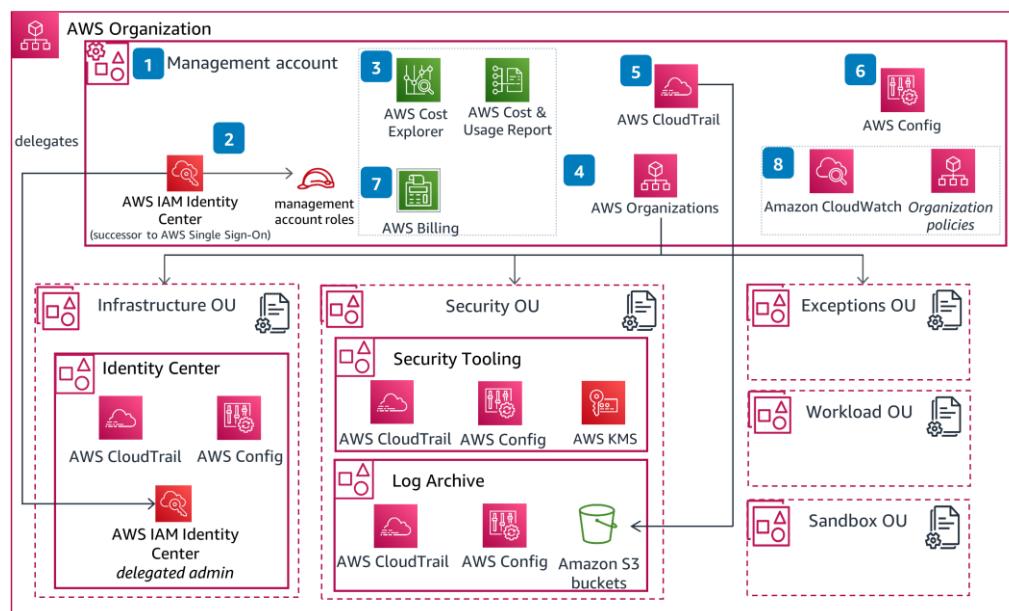


Ilustración 4. Ejemplo del uso de AWS Organizations para control de múltiples cuentas de AWS. Tomado de (Amazon Web Services, 2024).

Estos desafíos traen consigo la necesidad de implementar nuevas herramientas, controles y políticas para gestionar múltiples cuentas y conexiones con otras nubes. Un caso común es el uso de servicios como 'AWS Organizations' o 'Control Tower', que facilitan la aplicación de controles avanzados necesarios para grandes empresas. Sin embargo, estos servicios también introducen nuevos retos en términos de seguridad, gobernanza y administración de múltiples cuentas en AWS (Amazon Web Services, 2024). A lo largo del trabajo de máster, se profundizará en estas funcionalidades de seguridad y su importancia en entornos empresariales complejos.

### 3. Tecnologías SIEM

Las tecnologías SIEM (*Security Information and Event Management*), son soluciones avanzadas de seguridad enfocadas en detectar, analizar y responder a brechas de seguridad antes de que afecten significativamente al negocio. El objetivo principal de las soluciones SIEM es elevar la cobertura, protección y respuesta a las diversas amenazas que existen hoy en día a nivel de ciberseguridad (Martínez, 2017).



Ilustración 5. Ciberataques promedio por empresa entre 2021 y 2024. Tomado de (Check Point, 2024).

En la actualidad, las soluciones SIEM modernas requieren no solo una integración con diversos proveedores de nube, sino también la capacidad de apalancarse en herramientas de inteligencia artificial para lograr aumentar los estándares de protección empresarial de ataques cibernéticos (IBM, 2024). Algunas de las características más importantes de los sistemas SIEM son:

- Recolección y consolidación de logs.
- Detección de anomalías de red.
- Análisis de comportamiento de usuarios y agentes.
- Configuración de dispositivos y endpoints.
- Priorización de vulnerabilidades.
- Correlación de Eventos.
- Gestión de incidentes y alertas de seguridad.
- Generación de informes.

Teniendo en cuenta la definición y características, es indispensable que todas las empresas tengan un sistema SIEM central para garantizar la protección y mejores estándares de seguridad ante un posible incidente, pues, así como muestra la Ilustración 5,

ha habido un incremento de más del 30% en ciberataques en 2024-Q2, llegando a los 1636 ataques por semana por organización (Check Point, 2024).

Existen múltiples soluciones SIEM modernas, algunas de ellas son:

- Splunk.
- IBM QRadar and X-Force.
- LogRhythm.
- Wazuh.
- Elastic.

La elección de la herramienta adecuada para administrar soluciones SIEM y XDR dependerá en gran medida de las necesidades específicas y el estado actual de la empresa. Todas estas herramientas ofrecen grandes capacidades, con similitudes importantes y fortalezas únicas. La decisión final suele basarse en factores como el costo, la madurez tecnológica de la solución y su facilidad de integración con los componentes de software existentes en la organización. No hay una única opción correcta, sino que cada empresa debe seleccionar la herramienta que mejor se ajuste a su contexto y requerimientos actuales.

# METODOLOGÍA

Existen varias metodologías para el diseño científico destinadas a llevar a cabo un proceso de divulgación de investigaciones. Para este trabajo de máster, se decidió implementar una metodología aplicada a procesos de ingeniería, donde se siguen una serie de diseños iterativos que buscan lograr un producto final detallado (Dieter, Schmidt, & Linda, 2012).

Es importante señalar que el desarrollo de esta investigación tendrá un enfoque exploratorio hacia la solución tecnológica y práctica del problema a resolver, por lo que no se desarrollarán las etapas de diseño centradas en el estudio de mercado ni en los componentes financieros para generar un plan de negocio a nivel comercial.

## 1. Identificación de Requisitos

En primer lugar, se analizaron las características más relevantes para el proyecto de grado. Esto se logró teniendo en cuenta la literatura existente sobre computación en la nube, seguridad en componentes distribuidos, tecnologías SIEM y ciberseguridad. En esta selección de los requisitos del usuario, se identificaron las siguientes necesidades:

Requisito	Relevancia	Medida
Sistema debe ser compatible con múltiples cuentas del proveedor de nube elegido	4	Objetiva
Sistema debe ser compatible con diversos proveedores de nube	5	Objetiva
Sistema debe poder escalar y crecer según tráfico	4	Objetiva
Sistema debe garantizar esquema de red encriptado	5	Objetiva
Sistema debe proteger los flujos de trabajo existentes	5	Objetiva
Sistema debe tener panel de acceso amigable para los operadores	4	Subjetiva
Sistema debe detectar vulnerabilidades o ataques	5	Objetiva
Sistema debe ser de licencia abierta	3	Objetiva

Tabla 1. Requisitos del Sistema SIEM en la Nube. Elaboración propia.

## 2. Casa de la Calidad

La “Casa de la Calidad” (*House of Quality, HoQ*) es una herramienta de diseño que permite un enfoque simple y poderoso para priorizar los puntos más relevantes en un diseño de ingeniería. Esta herramienta se originó en el astillero de Mitsubishi en Kobe y ha ayudado en el desarrollo de múltiples soluciones industriales, como electrodomésticos, infraestructuras, electrónica, ropa, circuitos integrados y más (Hauser, 1988).

La metodología de la Casa de la Calidad mapea los requisitos correspondientes con condiciones de ingeniería importantes y medibles. El resultado de estas características genera dos partes fundamentales: la matriz principal y la matriz de correlación del techo. Estas matrices son clave para comprender las relaciones subyacentes y su importancia de una manera medible.

Este enfoque puede adaptarse perfectamente a los procesos de seguridad en la nube, facilitando la identificación y priorización de los requisitos de seguridad en función de las condiciones técnicas específicas y medibles del entorno de computación en la nube.

Requisitos técnicos:																	
Requisitos del cliente:	Comunidad Proveedor Nube	Escalabilidad Proveedor Nube	Costos Proveedor Nube	Integraciones Proveedor Nube	Seguridad Proveedor Nube	Licenciamiento Proveedor Nube	Comunidad SIEM	Escalabilidad SIEM	Costos SIEM	Seguridad SIEM	Licenciamiento SIEM	Interfaz Gráfica SIEM	Tiempos respuesta SIEM	Despliegue SIEM	Número de Fila	Peso Fila	Importancia
Múltiples cuentas		●		●											1	2	4
Múltiples proveedores	●	●		●			●	●						●	2	6	5
Escalar según tráfico		●	●				●	●					●	●	3	6	4
Red encriptada				●						●					4	2	5
Protección flujos existentes		●		●			●		●					●	5	5	5
Panel de acceso					●		●		●			●			6	2	4
Licencia abierta	●			●		●					●				7	4	3
Número Columna	1	2	3	4	5	6	7	8	9	10	11	12	13	14			
Puntaje Columna	8	18	4	12	10	3	9	14	4	14	3	4	4	14			
Prioridad Final (Top)		1		5				2		4					3		

Ilustración 6. Casa de la Calidad para el sistema SIEM. Elaboración propia.

La elaboración de la casa de la calidad permite concluir que, para el desarrollo del trabajo de máster, los requisitos finales más importantes para tener en cuenta son:

- Escalabilidad proveedor nube.
- Escalabilidad SIEM.
- Despliegue SIEM.
- Seguridad SIEM.
- Integraciones proveedor nube.

Se tomarán estas métricas/índices como los elementos prioritarios en la elección de las herramientas finales del SIEM.

### 3. Matriz Morfológica

La matriz morfológica es una herramienta poderosa que permite generar soluciones basadas en las posibles variaciones de las características de un problema (Derek, 2010).

Las características primordiales para analizar serán:

- Proveedor de Nube.
- Herramienta SIEM.
- Herramienta de Infraestructura como Código.

#### Proveedor de Nube:

A continuación, se evaluarán y compararán las 3 alternativas principales para el proveedor de computación en la nube: AWS, Azure y GCP.

Alternativa	Ventajas	Desventajas
Amazon Web Services  Ilustración 7. AWS Logo. Tomado de (AWS, 2024).	AWS es la nube líder según el cuadrante de Gartner 2023. Mayor comunidad y ecosistema. Tiene la infraestructura más robusta a nivel mundial.	Costo potencialmente alto. Curva de aprendizaje alta para nuevos usuarios debido a todos sus servicios.

Alternativa	Ventajas	Desventajas
Azure  Ilustración 8. Azure Logo. Tomado de (Microsoft, 2024).	Integración con servicios y productos de Microsoft. Estrategia híbrida robusta para integraciones con otros proveedores. Mejor nomenclatura de servicios.	Interfaz de usuario menos intuitiva para nuevos usuarios. Rendimiento inferior en algunas regiones en comparación con AWS o GCP.

Alternativa	Ventajas	Desventajas
Google Cloud Platform	Mayor innovación en Machine Learning y Big Data.	Menor madurez tecnológica en comparación con AWS y Azure.

 <p><i>Ilustración 9. GCP Logo.</i> <i>Tomado de (Google, 2024).</i></p>	<p>Precios más competitivos debido a su proceso de atracción de clientes.</p> <p>Excelente infraestructura global.</p>	<p>Ecosistema con menor comunidad y soporte.</p> <p>Menor oferta de servicios y administración avanzada de recursos.</p>
---	--	--

Tabla 2. Alternativas de proveedor de nube. Elaboración propia.

### Herramienta SIEM:

A continuación, se evaluarán y compararán las 3 alternativas principales para la herramienta SIEM: Splunk, IBM QRadar, Wazuh.

Alternativa	Ventajas	Desventajas
 <p><i>Ilustración 10. Splunk Logo. Tomado de (Splunk, 2024).</i></p>	<p>Líder en el mercado según Gartner 2023.</p> <p>Interfaz de usuario intuitiva con integraciones directas.</p> <p>Altamente escalable para grandes volúmenes de datos y eventos.</p>	<p>Alto costo, tanto en almacenamiento como licencias.</p> <p>Infraestructura robusta y considerable para su ejecución y mantenimiento.</p>

Alternativa	Ventajas	Desventajas
 <p><i>Ilustración 11. IBM QRadar Logo. Tomado de (IBM, s.f.).</i></p>	<p>Detección y prevención de amenazas a través de inteligencia artificial.</p> <p>Integración con productos de IBM como WAS, MQ, DataPower y IHS.</p> <p>Arquitectura modular según las necesidades de la empresa.</p>	<p>Costo elevado en licencias y mantenimiento.</p> <p>Curva de aprendizaje significativa, especialmente por funcionalidades avanzadas.</p> <p>Difícil personalización.</p>

Alternativa	Ventajas	Desventajas

 <i>Ilustración 12. Wazuh Logo. Tomado de (Wazuh, 2024).</i>	<p>Código 100% open-source, con una comunidad creciente en los últimos años.</p> <p>Fácil despliegue y escalabilidad según las necesidades de la empresa.</p> <p>Integración directa con servicios de nube como AWS y Azure.</p>	<p>Menos características avanzadas que sí ofrecen Splunk y QRadar.</p> <p>Soporte limitado y menores tiempos de respuesta debido a ser de código abierto.</p> <p>Curva de aprendizaje mayor.</p>
--	--	--

Tabla 3. Alternativas de herramienta SIEM. Elaboración propia.

### Herramienta de Infraestructura como Código:

A continuación, se evaluarán y compararán las 3 alternativas principales para la herramienta de Infraestructura como Código: Terraform, CDK y CloudFormation.

Alternativa	Ventajas	Desventajas
 <i>Ilustración 13. Terraform Logo. Tomado de (Hashicorp, 2024).</i>	<p>Compatibilidad con múltiples proveedores (<i>providers</i>), tanto de nube como de servicios adicionales.</p> <p>Altamente veloz.</p> <p>Gran comunidad activa con alto porcentaje de código abierto.</p> <p>Agnóstico a la nube que se use.</p>	<p>Menor compatibilidad algunos servicios de AWS.</p> <p>Gestión del estado (<i>state</i>) compleja.</p> <p>No se garantiza soporte a nivel productivo en caso de errores.</p>

Alternativa	Ventajas	Desventajas
 <i>Ilustración 14. CDK Logo. Tomado de (AWS, 2024).</i>	<p>Infraestructura como Código en lenguajes familiares (Python, TypeScript, Java, etc).</p> <p>Abstracciones muy sencillas de utilizar para servicios de AWS.</p> <p>Soporte nativo de AWS con mejores prácticas de seguridad incluidas.</p>	<p>Altamente vinculado a AWS.</p> <p>Curva de aprendizaje mayor por su relación con CloudFormation.</p> <p>Más lento que Terraform, ya que sintetiza a CloudFormation.</p>

Alternativa	Ventajas	Desventajas
<p>CloudFormation</p>  <p>Ilustración 15. CloudFormation Logo. Tomado de (AWS, 2024).</p>	<p>Herramienta nativa de AWS con soporte directo hacia todos los servicios.</p> <p>Excelente para empezar, debido a su lenguaje declarativo en YAML o JSON (sin necesidad de requerir programar).</p>	<p>Altamente vinculado a AWS.</p> <p>Su definición en YAML o JSON lo hace poco dinámico para proyectos robustos.</p> <p>Manejo de errores limitado.</p>

Tabla 4. Alternativas de Infraestructura como Código. Elaboración propia.

#### Resultados de la matriz morfológica:

Reto	Alternativa 1	Alternativa 2	Alternativa 3
Despliegue en la nube escalable	 <span>AWS</span> <div style="display: flex; justify-content: space-around;"> <span>A</span> <span>C</span> </div>	 <span>Azure</span> <div style="display: flex; justify-content: space-around;"> <span>B</span> </div>	 <span>GCP</span>
Herramienta SIEM avanzada	 <span>splunk</span> <div style="display: flex; justify-content: space-around;"> <span>A</span> </div>	 <span>IBM Radar</span> <div style="display: flex; justify-content: space-around;"> <span>B</span> </div>	 <span>wazuh</span> <div style="display: flex; justify-content: space-around;"> <span>C</span> </div>
Herramienta de Infraestructura como Código	 <span>Terraform</span> <div style="display: flex; justify-content: space-around;"> <span>A</span> </div>	 <span>CDK</span> <div style="display: flex; justify-content: space-around;"> <span>C</span> </div>	 <span>CF</span> <div style="display: flex; justify-content: space-around;"> <span>B</span> </div>
<b>Solución A</b>			
<b>Solución B</b>			
<b>Solución C</b>			

Tabla 5. Matriz morfológica para solución SIEM con opciones A, B, C. Elaboración propia.

Una vez teniendo listas las diversas soluciones (A, B, C), se procede a llevar a cabo una elección basada en la matriz PUGH. Esta permite a un ingeniero discernir desde una matriz morfológica y llegar a la mejor alternativa basada en componentes cualitativos. Esta

compara los requisitos explorados en la sección “**Error! Reference source not found.**”, con las 3 soluciones propuestas. Se evalúan de forma objetiva y se llega a una decisión final.

Solución	A	B	C
Sistema debe ser compatible con múltiples cuentas del proveedor	=	-	=
Sistema debe ser compatible con diversos proveedores de nube	=	-	=
Sistema debe poder escalar y crecer según tráfico	=	=	-
Sistema debe garantizar esquema de red encriptado	=	=	=
Sistema debe proteger los flujos de trabajo existentes	=	+	=
Sistema debe tener panel de acceso amigable para los operadores	=	=	+
Sistema debe detectar vulnerabilidades o ataques	=	=	=
Sistema debe ser de licencia abierta	=	-	+
Puntos positivos (total)	0	1	2
Puntos negativos (total)	0	3	1
<b>Resultado final</b>	<b>0</b>	<b>-2</b>	<b>1</b>

Tabla 6. Matriz de decisión PUGH. Elaboración propia.

Como se puede observar en la Tabla 6, la solución más alineada con los criterios de decisión final fue la “**Solución C**”. Esto quiere decir que la solución óptima para el trabajo de máster tendrá los siguientes componentes principales:

- Proveedor nube: Amazon Web Services.
- Herramienta SIEM: Wazuh.
- Herramienta Infraestructura como Código: Cloud Development Kit.

A lo largo de las siguientes secciones, se procederá a llevar a cabo la elaboración exhaustiva de los componentes, arquitecturas, codificaciones e implementaciones para llevar esta idea a una realidad que pueda ser aplicada en entornos empresariales modernos.

## 4. Arquitectura General del Sistema

Teniendo en cuenta las secciones anteriores, se procede a diseñar una arquitectura general del sistema SIEM enfocado en la nube de AWS:

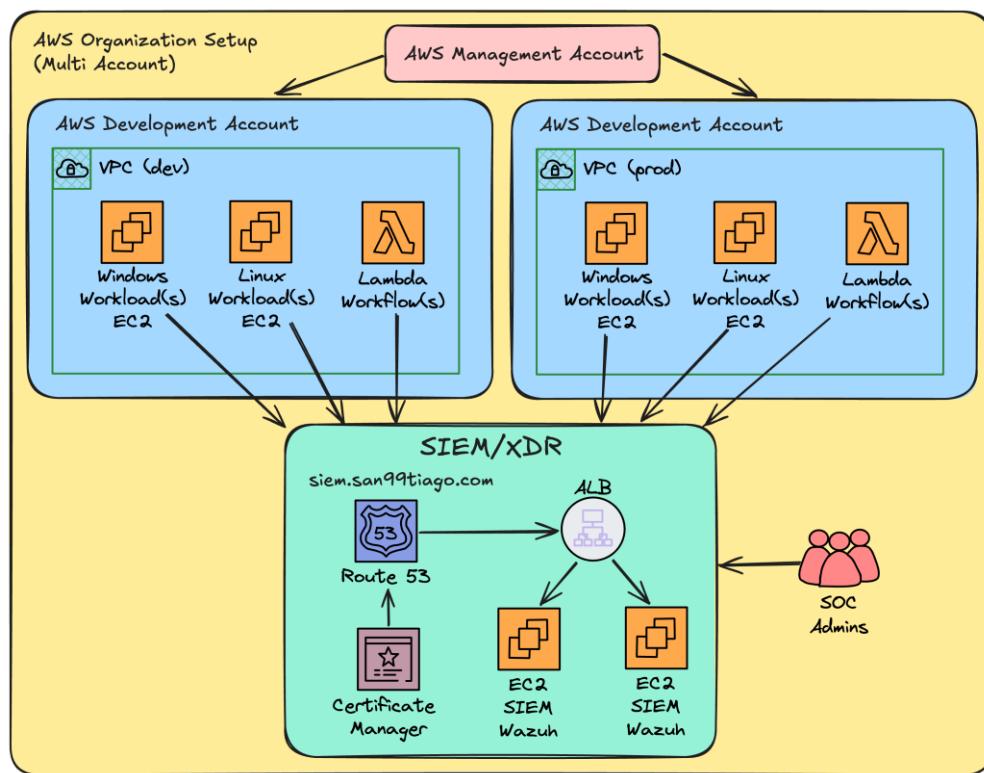


Ilustración 16. Arquitectura general del sistema SIEM en AWS. Elaboración propia.

En la Ilustración 16, se pueden evidenciar los componentes generales de la solución propuesta, en donde se tienen los componentes:

- Esquema multi cuenta de AWS manejado por AWS Organizations.
- Esquema de red granular en donde cada cuenta tiene su propia red privada encriptada.
- Sistema SIEM y XDR implementado con DNS centralizado (en este caso con el dominio de mi propia empresa “`siem.san99tiago.com`”).
- Posibilidad de acceso a los especialistas de seguridad SOC (admins), con acceso directo a los servidores SIEM.

En las siguientes secciones se procede a explicar el paso a paso de cómo construir este esquema de ciberseguridad SIEM enfocado en la nube de Amazon Web Services.

## DESARROLLO DEL PROYECTO

En esta sección del trabajo de máster, se mostrará de forma resumida el proceso de planeación, diseño, codificación y despliegue de cada uno de los componentes del proyecto. Cada subsección estará enfocada en uno de los objetivos específicos planteados en la definición de (Objetivos Específicos).

De igual forma, el proyecto fue realizado con ayuda de Git, el software más importante de “Control de Versiones”, que permite llevar a cabo proyectos de programación de forma iterativa y con las mejores prácticas de la industria. Este es el enlace del repositorio Open-Source creado para compartir el proyecto y que futuros ingenieros de ciberseguridad puedan aprender de él:



Ilustración 17. Imagen de mi proyecto Open-Source para la elaboración del trabajo de grado. Elaboración propia.

### Repository de GitHub (elaboración propia).

Enlace: <https://github.com/san99tiago/aws-cybersecurity-siem>

Tabla 7. Repository de GitHub del proyecto (aws-cybersecurity-siem). Elaboración propia.

## 1. Esquema Organizacional en AWS

Cuando se desarrollan proyectos avanzados en la nube de Amazon Web Services (AWS), es fundamental planificar y aplicar las mejores prácticas para la estructura organizacional, tanto a nivel de estandarización, como se seguridad. Muchas empresas suelen iniciar con una sola cuenta en AWS, pero conforme sus proyectos crecen, se vuelve esencial modernizar la infraestructura y gobernanza para escalar de manera eficiente a múltiples cuentas independientes.

Estas cuentas, administradas desde una cuenta principal, permiten gestionar centralmente costos, etiquetas, proyectos, controles de seguridad y estándares de la

organización, garantizando así aplicar las mejores prácticas para cada una de sus cuentas de una forma estandarizada. Para facilitar esta gestión, AWS ofrece el servicio de "AWS Organizations", diseñado específicamente para manejar estas necesidades de manera escalable y eficiente (Amazon Web Services, 2024).

A continuación, se muestra un ejemplo de una empresa que a lo largo del tiempo ha creado cuentas de AWS sin ninguna jerarquía, organización o estándar. Esto es altamente peligroso, porque aumenta la superficie de ataque, y es muy complejo responder a incidentes de ciberseguridad, pues no se tiene un lugar centralizado de análisis y control de todas las cuentas:

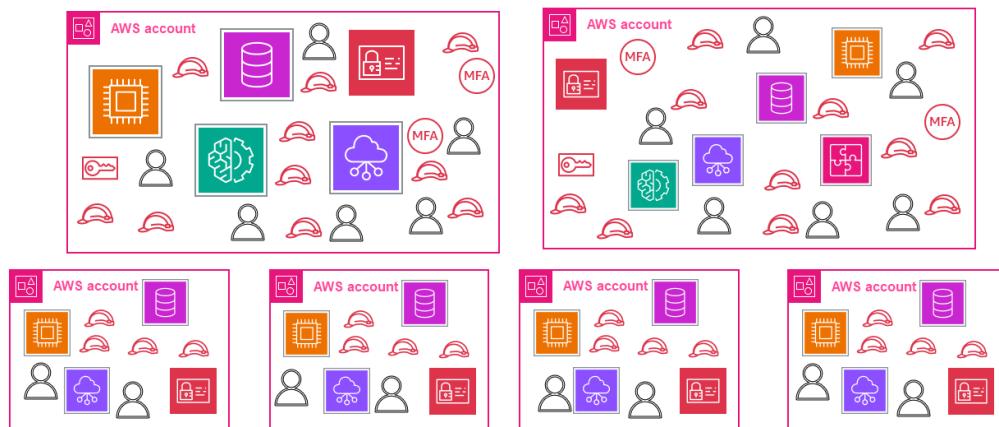


Ilustración 18. Ejemplo de múltiples cuentas de AWS sin ningún tipo de jerarquía ni esquema organizacional. Elaboración propia.

Como se puede observar en la Ilustración 18, se vuelve extremadamente complejo responder a preguntas clave sobre esta empresa. Por ejemplo:

- ¿Qué recursos son valiosos para la empresa y cuáles ya no hacen parte de los flujos de trabajo?
- ¿Qué cuentas y servicios corresponden a qué línea de negocio?
- ¿Cuáles son los permisos y límites de accesos a los recursos en cada cuenta?
- ¿Qué equipos/áreas son las dueñas de ciertos recursos?
- ¿Cómo están distribuidos los costos operacionales según centros de costos o áreas internas de la empresa?
- ¿Qué controles de seguridad existen y cómo limitar los permisos de acceso en caso de un incidente?

Todas estas preguntas son indispensables para cualquier empresa a nivel tecnológico y con esquemas de seguridad antiguos, es muy complejo llevar la trazabilidad de cada punto estratégico (Lankford, Andrew, & Rice, 2022).

Es por esto por lo que en este proyecto mostraré el uso de AWS Organizations poder para dar solución a cada una de estas preguntas y establecer una jerarquía organizacional en las cuentas de AWS, que permita las mejores prácticas de estandarización y control en Amazon Web Services.

### AWS Organizations:

Es un servicio de AWS que permite la creación, administración y gobernanza de entornos de AWS a medida que una empresa escala sus recursos (Amazon Web Services, 2024). Ofrece las siguientes ventajas competitivas en una organización:

- Crear cuentas de AWS y alojar recursos en ellas.
- Agrupar cuentas según jerarquías y flujos de trabajo.
- Aplicar políticas de control y gobernanza según la jerarquía necesaria.
- Simplificar el pago y distribución de recursos en múltiples cuentas.
- Habilitar servicios especializados para organización y protección de cuentas.

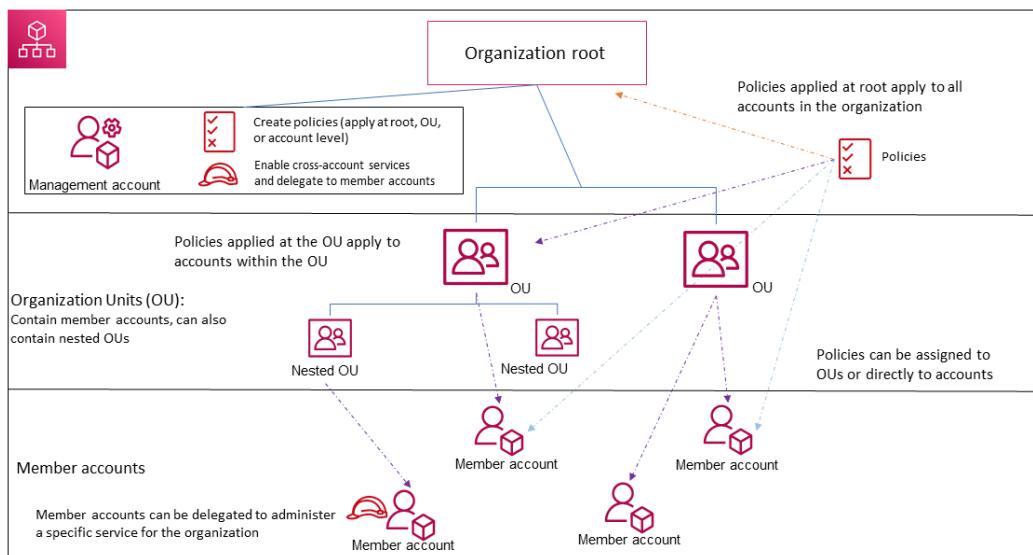


Ilustración 19. Ejemplo de AWS Organizations con los componentes principales. Tomado de (AWS, 2024).

### Uso de AWS Organizations para el proyecto de máster:

Para este trabajo de máster, el manejo de AWS Organizations tiene alta relevancia, pues se estará trabajando con un esquema SIEM moderno, en donde tendremos diversas cuentas de Amazon Web Services, todas con diversos requisitos de controles de seguridad y políticas de servicios restringidos.

A continuación, se muestra la arquitectura propuesta para el esquema organizacional:

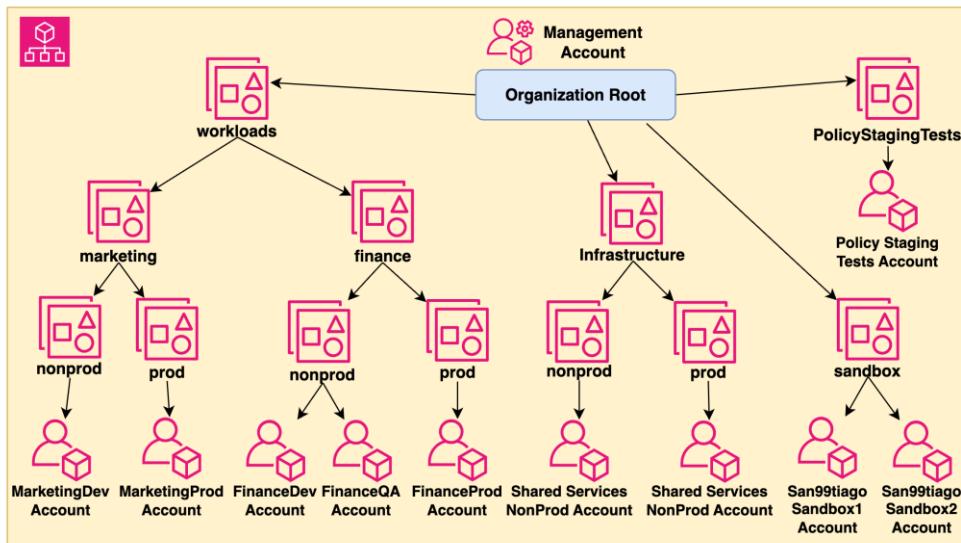


Ilustración 20. Esquema organizacional en AWS para mi proyecto de máster. Elaboración propia.

Como se puede visualizar en la Ilustración 20, la estructura elegida tiene las siguientes características relevantes:

- Cuenta principal de administración (cuenta personal del autor de la tesis “san99tiago” [Santiago García Arango]).
- Múltiples cuentas de “Workloads” (flujos de trabajo), en donde se tendrán los componentes del proyecto, tales como el SIEM y los servidores de prueba para validar cargas de trabajo simulando flujos productivos en AWS.
- Creación de “Service Control Policies” (políticas de control de servicios), para aplicar restricciones específicas a cada cuenta, y poder garantizar una gobernanza estandarizada en el esquema de seguridad.
- Cuentas de “Sandbox” (pruebas), en donde se manejarán los nuevos servicios, experimentos o validaciones de concepto que estén desacopladas de las cuentas organizacionales principales. Esto garantizará aislar componentes de experimentos, de los flujos de trabajo reales.

A continuación, se muestra mi repositorio de GitHub Open-Source en donde codifiqué mi esquema organizacional en AWS enfocado a las mejores prácticas de IaC (Infraestructura como Código) y múltiples cuentas de AWS productivas:



Ilustración 21. Repositorio propio para el esquema organizacional del proyecto de máster. Elaboración propia.

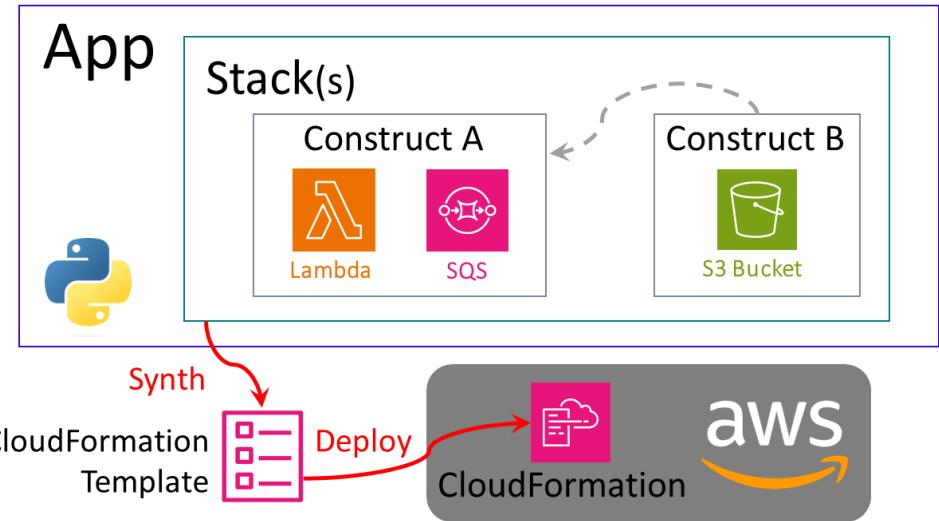
### Repositorio de GitHub (elaboración propia).

Enlace: <https://github.com/san99tiago/aws-cdk-organizations-demo>

Tabla 8. Repositorio de GitHub del proyecto (aws-cdk-organizations-demo). Elaboración propia.

Llevar a cabo esta estructura fue un gran reto, debido a que no solamente se creó el diseño organizacional con las mejores prácticas de ciberseguridad, sino que también se codificó todo con “Infraestructura como Código”, es decir, con cada uno de los componentes organizacionales administrados desde un repositorio en GitHub OpenSource, que permitiera desplegar cada cuenta, control y política de forma automatizada.

Para automatizar y desplegar el esquema organizacional de la Ilustración 20, se procedió a codificar cada uno de los componentes de dicha organización. Para esto, se empleó la herramienta de Infraestructura CDK “Cloud Development Kit” (AWS, 2024). Esta herramienta permite crear componentes de infraestructura de forma programática a través de los lenguajes de programación más relevantes en la industria, como Python, TypeScript, Java, entre otros (AWS, 2024).



Si bien el repositorio tiene más de mil líneas de código fuente y es complejo explicarlo de forma detallada en este documento, procederé a indicar los puntos de acceso esenciales para entender los componentes de software:

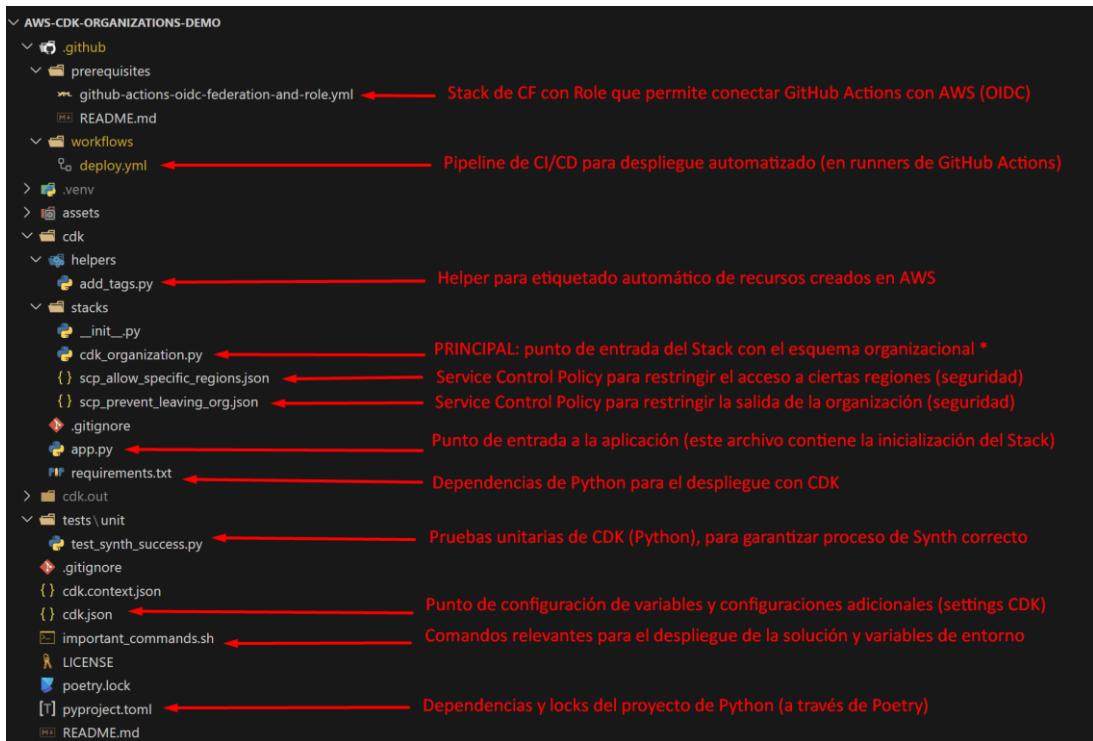


Ilustración 23. Explicación detallada del repositorio propio para el despliegue organizacional. Elaboración propia.

Una vez definida la organización, se procede a crear un pipeline de CI/CD (Integración Continua y Despliegue Continuo), en donde se utiliza la herramienta de GitHub Actions (GitHub, 2024) para poder desplegar la solución en AWS de forma transparente y con IaC:

## [CI/CD] san99tiago/aws-cdk-organizations-demo

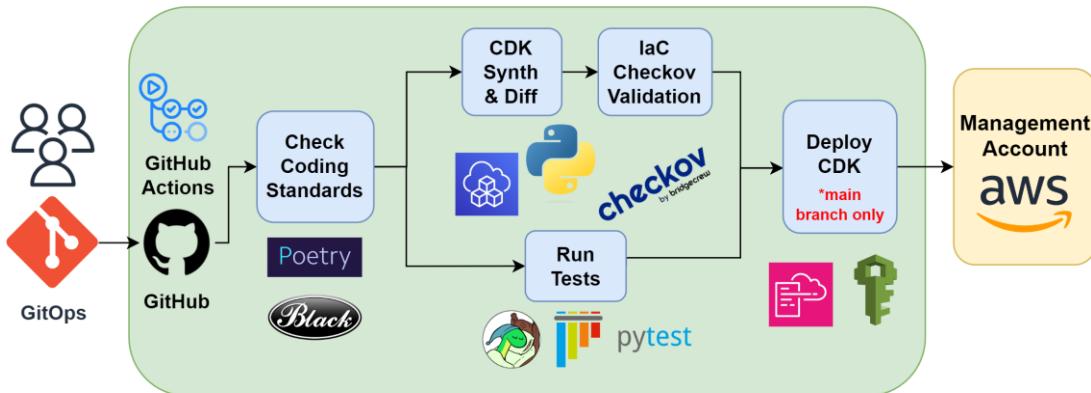


Ilustración 24. Pipeline de Despliegue Automático para el esquema organizacional. Elaboración propia.

Así como se puede observar en la Ilustración 24, se logra la automatización completa de esta solución organizacional. Esto permitirá a los ingenieros en ciberseguridad la aplicación de políticas de control y gobernanza de forma automatizada y transparente en todas las cuentas de AWS que se tengan en la empresa.

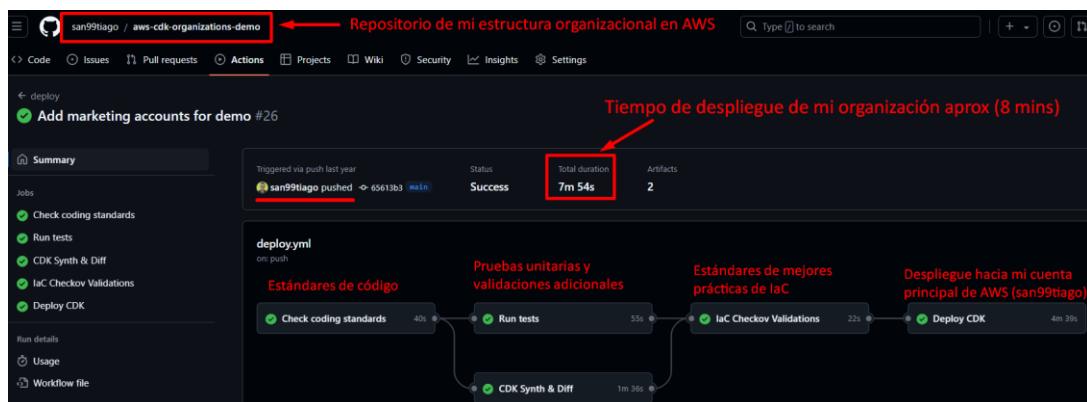


Ilustración 25. Despliegue automatizado de estructura organizacional propia. Elaboración propia.

Una vez realizado el despliegue, se procede a validar que el Single Sign On (también conocido como “Identity Center” en AWS), haya quedado bien configurado:

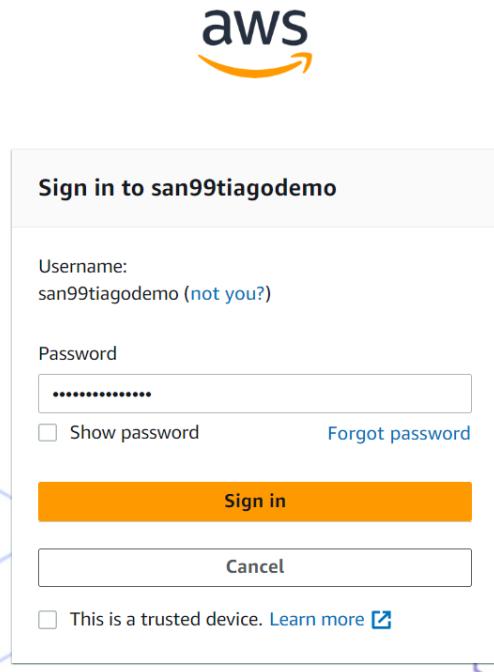


Ilustración 26. Resultado del Single Sign On de mi organización. Elaboración propia.

Una vez realizado el login, se visualiza cómo se tiene acceso a cada una de las cuentas, facilitando el control, validación e implementación de controles de seguridad para cada una de las cuentas de la empresa:

Account Name	Email Address
finance-dev	754647513616   san99tiagodemo+finance-dev@gmail.com
finance-prod	659491924207   san99tiagodemo+finance-prod@gmail.com
finance-qa	600079372754   san99tiagodemo+finance-qa@gmail.com
marketing-dev	571619567540   san99tiagodemo+marketing-dev@gmail.com
marketing-prod	350405871851   san99tiagodemo+marketing-prod@gmail.com
policy-staging-tests	218227490026   san99tiagodemo+policy-staging-tests@gmail.com
san99tiago-sandbox-1	756402921065   san99tiagodemo+san99tiago-sandbox-1@gmail.com
<b>san99tiagodemo</b>	800782014132   san99tiagodemo@gmail.com
shared-services-non-prod	672205386739   san99tiagodemo+shared-services-non-prod@gmail.com
shared-services-prod	250755223596   san99tiagodemo+shared-services-prod@gmail.com

Ilustración 27. Estructura organizacional para usuario admin con acceso a todas las cuentas. Elaboración propia.

Ahora, se procede a realizar una prueba de seguridad, en donde se tratarán de realizar 2 acciones que NO están aceptadas por las políticas de control de servicios de la empresa. Estas son:

- Prueba 1: Crear servidores no aceptados por los controles internos de la empresa.
- Prueba 2: Desplegar servicios de AWS en regiones fuera de la zona geográfica en donde está la empresa (en este caso Estados Unidos).

### Prueba 1: Crear servidores no aceptados por los controles internos de la empresa

A continuación, se procede a simular la creación de un servidor que no cumple con las políticas organizacionales. Se debe validar que no se tengan permisos para lograrlo, pues esto podría simular el acto de un intruso en la organización realizando un ataque (ejemplo: desplegar un servidor con grandes capacidades de CPU/RAM para minar criptomonedas).

En este caso, se tratará de desplegar una instancia de EC2 de tipo "t2.micro" (**permitida**), y debe ser exitoso a nivel de despliegue:

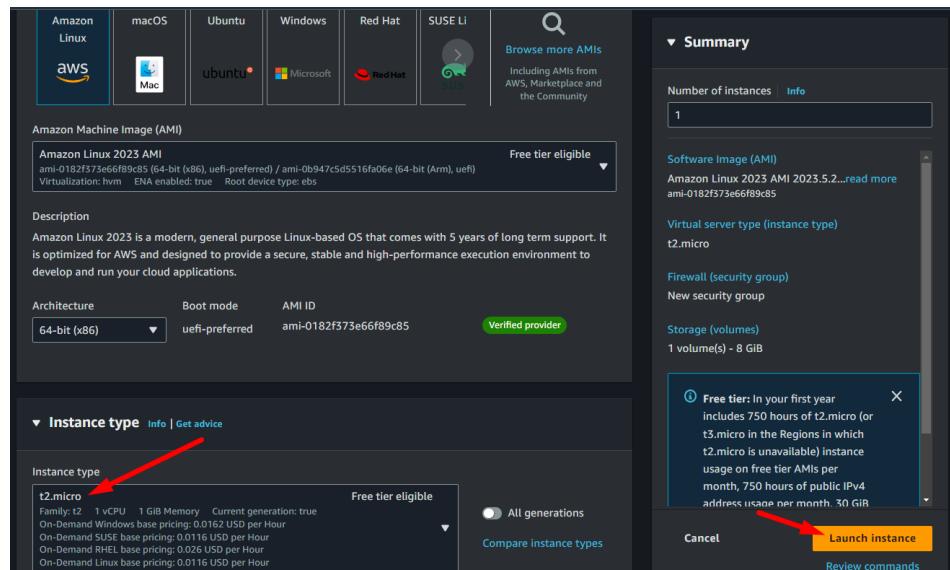


Ilustración 28. Proceso de despliegue de instancia permitida "t2.micro". Elaboración propia.

Esta acción fue aceptada de forma exitosa:

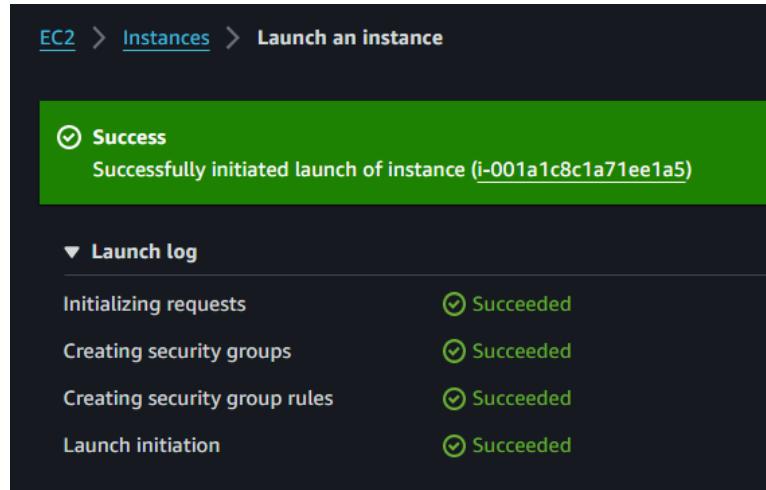


Ilustración 29. Resultado despliegue de instancia permitida. Elaboración propia.

Sin embargo, si procedemos a tratar de desplegar una instancia con muchos recursos, que no está en el plan de gobernanza empresarial (**no permitida**), se debería tener un control preventivo que bloquee dicha acción. Ahora se procede a llevar a cabo este proceso:

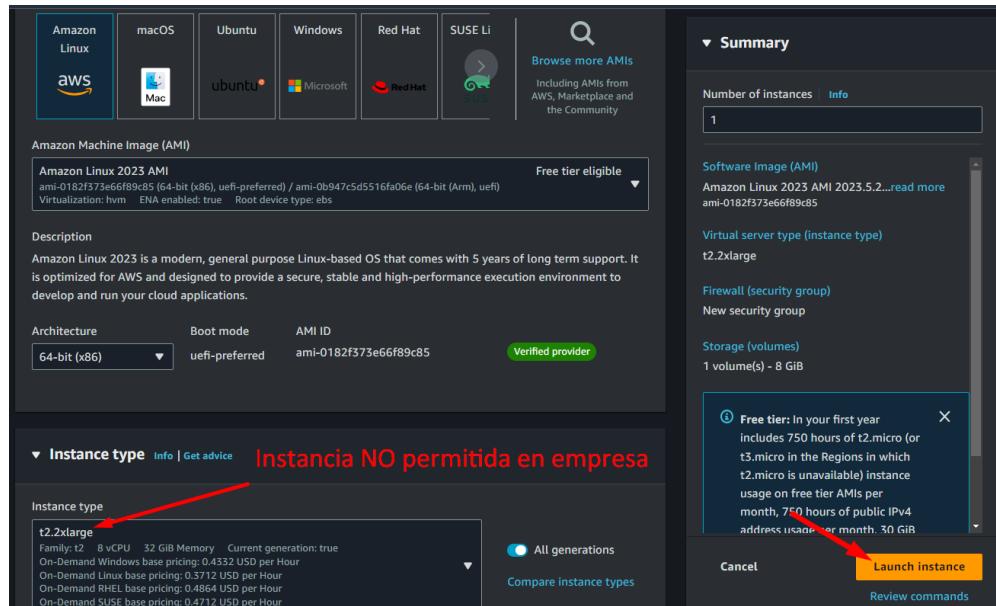


Ilustración 30. Proceso de despliegue de instancia permitida "t2.2xlarge". Elaboración propia

Una vez realizada esta acción, se logra ver que hay un bloqueo preventivo (así como fue planeado con los “Service Control Policies” de la organización):



Ilustración 31. Resultado de despliegue fallido para instancias no permitidas en la estructura organizacional. Elaboración propia.

Con la ilustración Ilustración 31, se puede observar de forma exitosa el esquema de ciberseguridad preventivo desplegado a nivel de recursos de AWS para cada una de las cuentas de la empresa.

## Prueba 2: Desplegar servicios de AWS en regiones fuera de la zona geográfica en donde está la empresa

A continuación, se procede a mostrar una acción de ciberseguridad no permitida por la empresa, en donde se intenta desplegar una base de datos en una región fuera de las ubicaciones permitidas de la empresa. Esto con el fin de prevenir la creación de servicios en sitios que no están avalados por el SOC de la empresa.

En este caso, se procede a intentar desplegar una base de datos RDS ("Relational Database Service") desde la región de "Asia Pacific / Mumbai" no autorizada:



Ilustración 32. Despliegue fallido de RDS debido a control preventivo en regiones no permitidas. Elaboración propia.

Con estas pruebas, procedemos a validar que la estructura organizacional cumple los estándares de seguridad esperados preventivos, garantizando así la gobernanza y seguridad de la empresa desde el punto de vista de excelencia organizacional en la nube.

The screenshot shows the AWS IAM Identity Center console with the following details:

- Left sidebar:** Shows navigation options like Dashboard, Users, Groups, Settings, Multi-account permissions (AWS accounts, Permission sets), Application assignments (Applications), and Related consoles (IAM).
- Top header:** Shows the AWS logo, Services, Search bar, and N. Virginia region.
- Main content area:** Title: "AWS accounts" with a sub-section "AWS Organizations AWS accounts". A red box highlights the "AWS accounts" section. Sub-section title: "AWS accounts". A search bar and two buttons: "Hierarchy" and "List".
- Table:** Lists AWS accounts with their names and associated email addresses:
  - san9tiago (management account)
  - san9tiago-sandbox-1 (san9tiago-devsecops+san9tiago-sandbox-1@gmail.com)
  - san9tiago-sandbox-mon (san9tiago-devsecops+san9tiago-sandbox-mon@gmail.com)
  - san9tiago-tutorials-dev (san9tiago-devsecops+san9tiago-tutorials-dev@gmail.com)
  - san9tiago-tutorials-prod (san9tiago-devsecops+san9tiago-tutorials-prod@gmail.com)
  - san9tiago-workloads-prod (san9tiago-devsecops+san9tiago-workloads-prod@gmail.com)
  - san9tiago-workloads-dev (san9tiago-devsecops+san9tiago-workloads-dev@gmail.com)
- Right sidebar:** "Permission sets" column lists various access levels for each account, such as "ReadOnlyAccess", "AdministratorAccess", and "AdministratorAccesses".

Ilustración 33. Esquema organizacional exitoso desde el panel de control de AWS Identity Center.

Elaboración propia.

## 2. Creación de Redes e Infraestructura de Comunicaciones

La infraestructura de redes es crucial para la seguridad y el correcto funcionamiento de cualquier entorno empresarial, ya que permite la comunicación eficiente y segura entre sistemas, usuarios y servicios.

En este contexto, el manejo adecuado de acceso y la encriptación de los datos en tránsito es esencial para garantizar que la información intercambiada no sea interceptada o modificada por actores malignos. Implementar protocolos de seguridad como TLS/SSL en las comunicaciones, tanto internas como externas, protege la confidencialidad e integridad de los datos, reduciendo riesgos de ataques como la interceptación de paquetes o (MAM) "Man-In-the-Middle" (CloudFlare, 2024). Sin esta capa de protección, los datos pueden ser vulnerables durante su transmisión, lo que compromete la seguridad de toda la red (CISCO, 2024).

En AWS, el servicio de Amazon VPC ("Virtual Private Cloud") es fundamental para crear arquitecturas de red seguras y personalizadas en la nube. Con VPC, las empresas pueden definir sus propios entornos de red virtuales, configurando subredes, tablas de enrutamiento y gateways para controlar el tráfico de manera granular. Amazon VPC actúa como el pilar de la seguridad en la nube de AWS, permitiendo el aislamiento de recursos, el control del tráfico entrante y saliente mediante listas de control de acceso (ACL) y grupos de seguridad ("Security Groups"), y la conexión segura entre las redes internas y las externas (AWS, 2024).

En el contexto de AWS y redes, los componentes más importantes a considerar para la creación de recursos son (tomado de (AWS, 2024)):

- **Virtual Private Cloud (VPC):** Es una red virtual privada que permite aislar los recursos en AWS, ofreciendo control total sobre el entorno de red, como rangos de IP, subredes, y el enrutamiento del tráfico.
- **Subnets:** Son divisiones de una VPC que permiten segmentar y organizar tus recursos dentro de la red. Pueden ser públicas (con acceso a internet) o privadas (aisladas del acceso externo).
- **Internet Gateway:** Es el componente que permite que las subredes públicas dentro de una VPC tengan acceso a internet y reciban tráfico desde afuera.
- **NAT Gateway:** Es un servicio que permite que las subredes privadas dentro de una VPC puedan acceder a internet para descargar actualizaciones o acceder a servicios, sin exponerse directamente al tráfico de internet.

- **Route Tables:** Son tablas que definen las reglas de enrutamiento para controlar el flujo de tráfico entre las subredes, la internet, y otros servicios dentro o fuera de la VPC.

A continuación, se muestra un ejemplo genérico de un esquema de red en AWS:

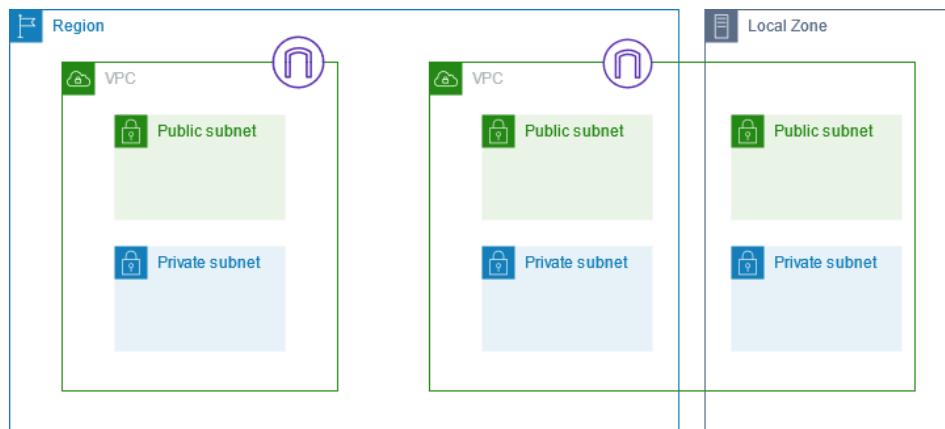


Ilustración 34. Ejemplo básico de estructura de red con VPC, Subnets y Local Zone. Tomado de (AWS, 2024).

### Creación de redes para el proyecto de máster:

Una vez entendidos los componentes más relevantes en AWS para la creación de redes en la nube, se procede a crear el siguiente esquema de redes que permitan el despliegue de recursos en la nube de forma aislada, encriptada y con las mejores prácticas de seguridad. Este proceso se realizó pensando en un esquema multi cuenta, en donde cada entorno tenga CIDRs asociados diferentes, simplificando la identificación y aislamiento de recursos en caso de un ataque cibernético.

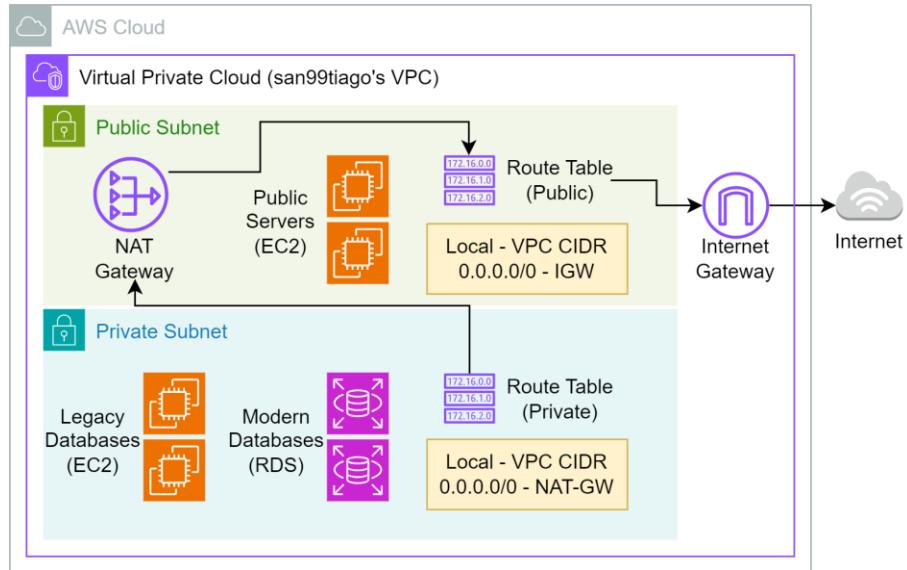


Ilustración 35. Esquema de red en AWS con VPC para proyecto de máster. Elaboración propia.

Teniendo en cuenta esta estructura, se puede observar que cada cuenta tendrá un esquema de red con los siguientes componentes:

- Virtual Private Cloud (san99tiago's VPC): red principal.
- Subnet Pública: subred pública enfocada servidores de acceso abierto.
- Subnet Privada: subred privada enfocada a bases de datos.
- Internet Gateway: permitirá la conexión con internet desde la red pública.
- Nat Gateway: permitirá conectividad hacia internet desde la red privada (one-way).

Teniendo listo el esquema de red de la Ilustración 35, la idea es poder automatizar el despliegue de forma sencilla. Se procedió a codificar cada uno de los componentes (VPC, Subnets, Log Groups, Endpoints, etc). Para esto, se empleó la herramienta de Infraestructura CDK “Cloud Development Kit” (AWS, 2024). Esta herramienta permite crear componentes de infraestructura de forma programática a través de los lenguajes de programación más relevantes en la industria, como Python, TypeScript, Java, entre otros (AWS, 2024).

Si bien el repositorio tiene muchas líneas de código fuente y es complejo explicarlo de forma detallada en este documento, procederé a indicar los puntos de acceso esenciales para entender los componentes de software y de red:

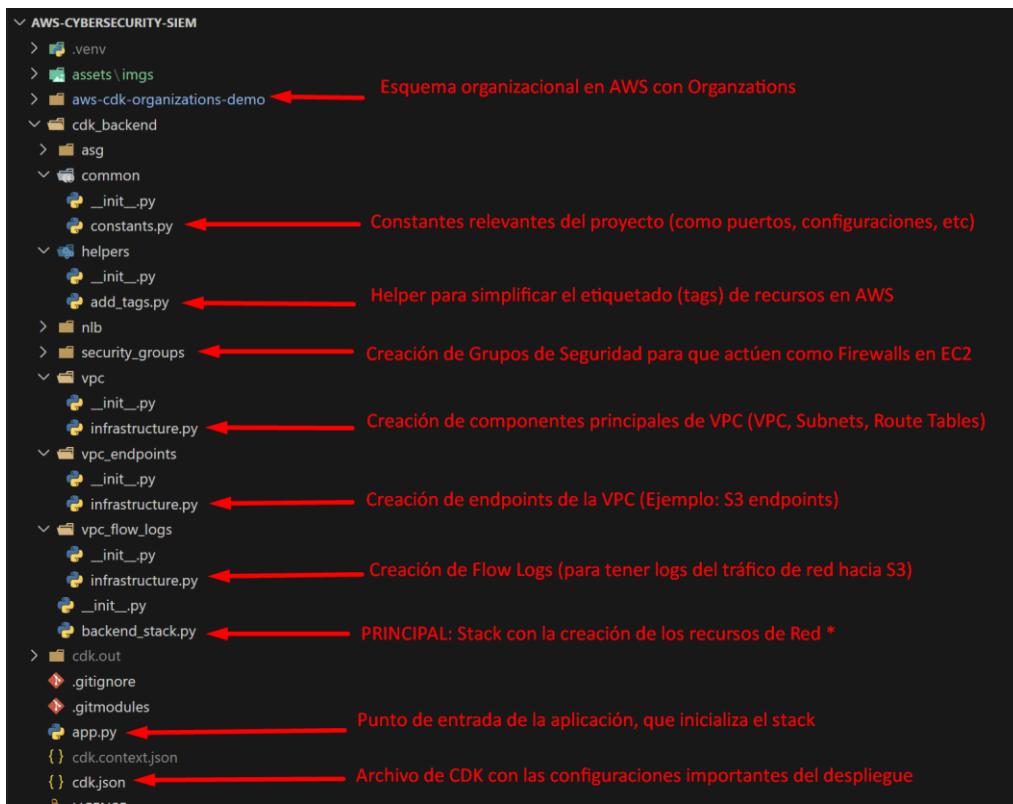


Ilustración 36. Explicación detallada del repositorio propio para el despliegue de Redes. Elaboración propia.

Una vez mostrada la configuración del código, se procede a mostrar el archivo de configuración general (CDK.json), el cual permite personalizar cada red que se planea desplegar, según la cuenta de AWS objetivo. En este caso, se desplegarán esquemas de red de tipo “DESARROLLO” y “PRODUCCIÓN”, garantizando las mejores prácticas de aislamiento y seguridad multi entorno:

```

1  {
2    "dev": {
3      "networking": {
4        "vpc_name": "main-vpc-dev",
5        "vpc_cidr": "10.0.0.0/20",
6        "public_subnet_mask": 24,
7        "private_subnet_mask": 24,
8        "enable_nat_gateway": false,
9        "enable_vpc_flow_logs": true,
10       "enable_vpc_endpoints": true
11     }
12   },
13   "prod": {
14     "networking": {
15       "vpc_name": "main-vpc-prod",
16       "vpc_cidr": "10.0.16.0/20",
17       "public_subnet_mask": 24,
18       "private_subnet_mask": 24,
19       "enable_nat_gateway": false,
20       "enable_vpc_flow_logs": true,
21       "enable_vpc_endpoints": true
22     }
23   }
24 }
```

Ilustración 37. Configuraciones para esquema de Red con CDK. Elaboración propia.

Con esta granularidad y configuración multi entorno, se procede a realizar los despliegues y a continuación se muestra el resultado del despliegue, con su respectivo mapa de red y conectividad:

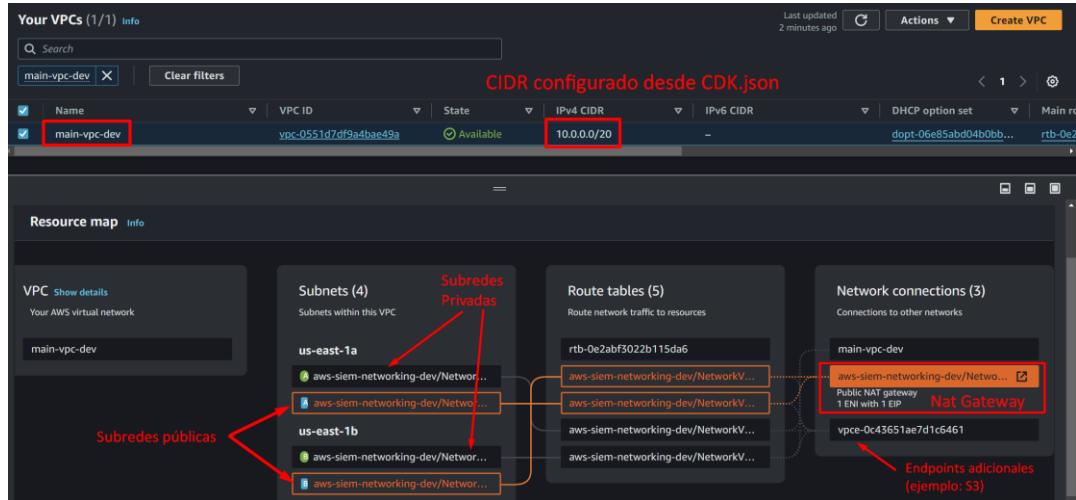


Ilustración 38. Resultados de la red desplegada en AWS con IaC. Elaboración propia.

Una vez desplegada la red, se procede a comprobar la correcta conectividad de esta, a través del despliegue de un servidor (EC2) en la red pública:



Ilustración 39. Despliegue de servidor en red pública. Elaboración propia.

Finalmente, se conecta a dicho servidor vía AWS Systems Manager (servicio para conectarse a servidores de EC2 a través de agentes sin necesidad de abrir puertos):

Session ID: san99tiago-  
srlz7utdkxeuzwolc5pn6pkcm

Instance ID: i-0346d17f0a622df56

**Terminate**

```
[root@ip-10-0-0-241 bin]# wget --server-response --spider https://www.google.com 2>&1
Spider mode enabled. Check if remote file exists.
--2024-09-09 04:11:13-- https://www.google.com/
Resolving www.google.com (www.google.com)... 142.251.16.105, 142.251.16.99, 142.251.16.147, ...
Connecting to www.google.com (www.google.com)|142.251.16.105|:443... connected.
HTTP request sent, awaiting response...
HTTP/1.1 200 OK ←
Content-Type: text/html; charset=ISO-8859-1
Content-Security-Policy-Report-Only: object-src 'none';base-uri 'self';script-src 'nonce-nyqNickDsMEYmsGyIRXzow' 'strict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https: http;;report-uri https://csp.withgoogle.com/csp/gws/other-hp
Accept-CH: Sec-CH-Prefers-Color-Scheme
P3P: CP="This is not a P3P policy! See g.co/p3phelp for more info."
Date: Mon, 09 Sep 2024 04:11:13 GMT
Server: gws
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Expires: Mon, 09 Sep 2024 04:11:13 GMT
Cache-Control: private
Set-Cookie: AEC=AVYB7crlt4-J7StvpTZNLAlyK3Dh9WkEMBqSIlwgklR5AeZ0Sj_si58DmQ; expires=Sat, 08-Mar-2025 04:11:13 GMT; path=/; domain=.google.com; Secure; HttpOnly; SameSite=lax
Set-Cookie: NID=517=09YrFmlvrL3C_P2tlgCNd16miPe8TqcZzSGe10kbqdE0nDdWerzvm4TOhJsc0eMtyl_hGdGZ65Ki0xCycGKmjKAtCnVTU10Sedy0FtccC4M0apmfFI7cwZfB1f33gcm5ZyVx0SRCxvWM7m0HV24b3UnyHsvLLixuktVpKcoPUbI153baFzRqWw; expires=Thu, 11-Mar-2025 04:11:13 GMT; path=/; domain=.google.com; HttpOnly
Alt-Svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000
Length: unspecified [text/html]
Remote file exists and could contain further links, but recursion is disabled -- not retrieving.
```

Respuesta OK desde servidores de google.com  
Conectividad validada!

[root@ip-10-0-0-241 bin]# ]]

Ilustración 40. Respuesta correcta de conectividad a internet (google.com) desde red pública. Elaboración propia.

Con estos experimentos se muestra la validez de las redes y su correcto funcionamiento a través del servicio de VPC en AWS, validado desde servidores de EC2 en las respectivas subredes.

Con el esquema de red, se puede proceder a la siguiente sección, en donde se hablarán de los flujos de trabajo (demo) de servidores en la nube, simulando procesamiento o bases de datos empresariales desplegadas en la nube.

### 3. Despliegue de Flujos de Trabajo (Servidores)

En esta sección, se desplegarán múltiples instancias de Amazon EC2 que simularán servidores dentro de los entornos de desarrollo (dev) y producción (prod) de la empresa. Estos servidores permitirán recrear flujos de trabajo reales que reflejan las operaciones diarias, con el objetivo de probar la integración y efectividad del sistema de gestión de eventos e información de seguridad (SIEM).

A través de estas simulaciones, se evaluarán las capacidades de detección de brechas de seguridad, asegurando que el SIEM sea capaz de monitorear y responder a incidentes en ambos entornos de red de manera efectiva.

#### Creación de servidores (EC2) para el proyecto de máster:

Así como se ha trabajado en las secciones anteriores, para la creación de servidores también se utilizará la herramienta de *Cloud Development Kit* (CDK) (AWS, 2024).

El objetivo será el despliegue de aproximadamente múltiples servidores, simulando flujos de trabajo en desarrollo y producción, a través de 3 entornos:

- Cuenta de producción: Tendrá 5 servidores Linux (Amazon Linux 2023 y Ubuntu) y 1 servidor Windows (Windows Server 2022).
- Cuenta de desarrollo: Tendrá 2 servidores Linux (Amazon Linux 2023).
- On Premises: Tendrá 1 servidor Windows (Windows 11 Home version 10).

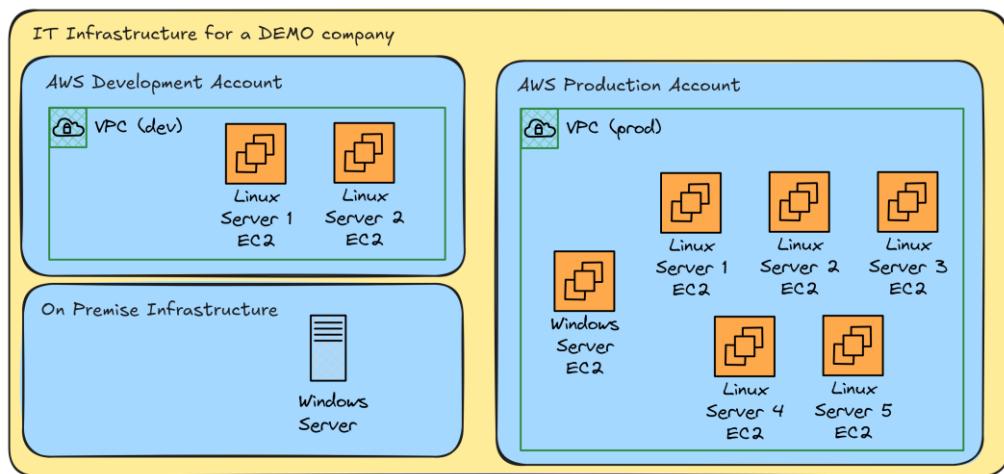


Ilustración 41. Esquema de servidores simulando flujo de trabajo en esquema híbrido con múltiples cuentas de AWS y On-Premise. Elaboración propia.

Si bien el repositorio tiene muchas líneas de código fuente y es complejo explicarlo de forma detallada en este documento, procederé a indicar los puntos de acceso esenciales para entender los componentes de los servidores demo para la simulación de flujos de trabajo:

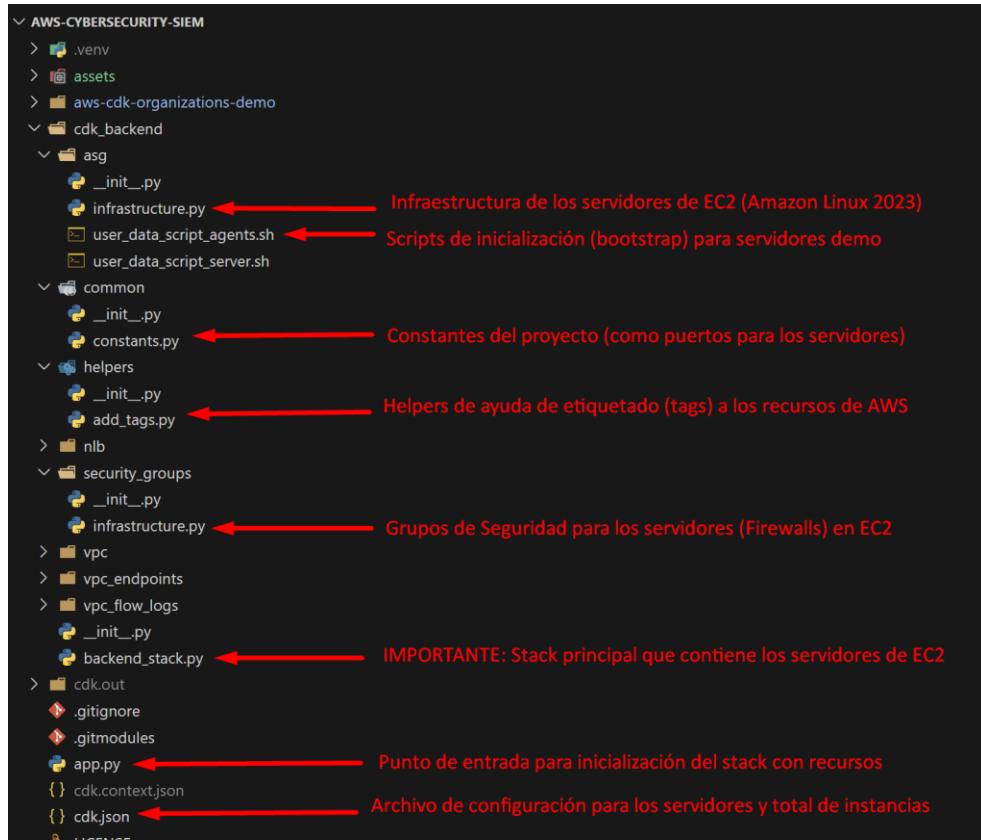


Ilustración 42. Explicación detallada del repositorio propio para el despliegue de servidores demo.

Elaboración propia.

Una vez mostrada la configuración del código, se procede a mostrar el archivo de configuración general (CDK.json), el cual permite personalizar cada instancia o servidor que se planea desplegar, según la cuenta de AWS objetivo. En este caso, se desplegarán esquemas servidores tanto en “DESARROLLO” como “PRODUCCIÓN”, garantizando las mejores prácticas de aislamiento y seguridad multi entorno:

```

1  {
2    "app_config": {
3      "dev": {
4        "demo_servers": {
5          "short_name": "demo-server",
6          "sg_cidrs_list": ["0.0.0.0/0"],
7          "instance_type": "t2.micro",
8          "ami_name": "al2023-ami-2023.5.20240903.0-kernel-6.1-x86_64",
9          "min_capacity": 2,
10         "max_capacity": 2,
11         "desired_capacity": 2
12       }
13     },
14     "prod": {
15       "demo_servers": {
16         "short_name": "demo-server",
17         "sg_cidrs_list": ["0.0.0.0/0"],
18         "instance_type": "t2.micro",
19         "ami_name": "al2023-ami-2023.5.20240903.0-kernel-6.1-x86_64",
20         "min_capacity": 5,
21         "max_capacity": 5,
22         "desired_capacity": 5
23       }
24     }
25   }
26 }

```

Ilustración 43. Configuraciones para servidores demo con CDK. Elaboración propia.

Una vez creada la Infraestructura como Código, se procede a desplegar los servidores y validar su correcto funcionamiento en la respectiva red. A continuación se muestran los servidores productivos en el entorno de AWS:

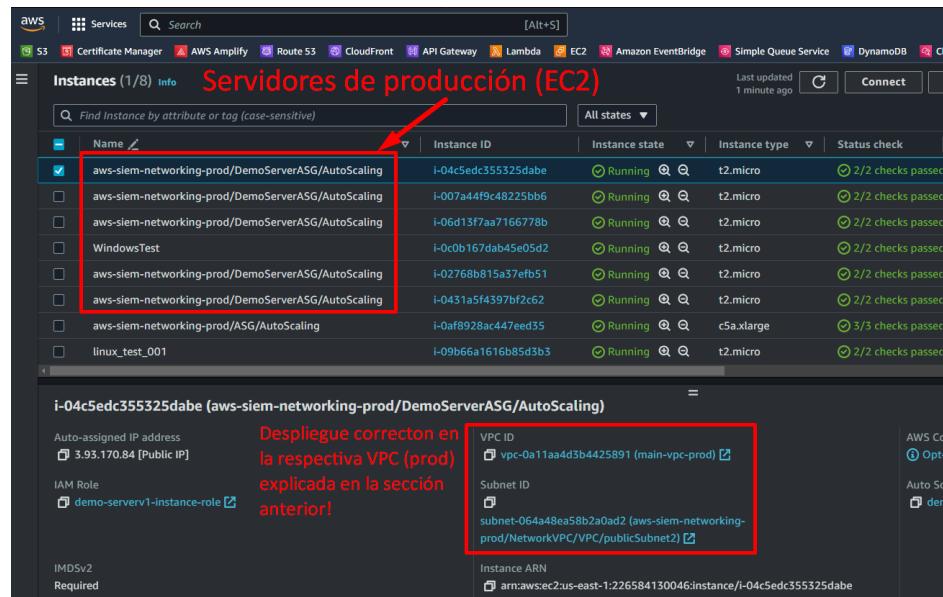


Ilustración 44. Servidores demo para el trabajo de máster simulando flujos productivos en AWS.

Elaboración propia.

Con estos servidores, se podrá simular las cargas de trabajo que una empresa real debe manejar para el correcto funcionamiento tecnológico. Se puede validar que todas se encuentran en la red empresarial explicada en la sección anterior mediante la VPC.

## 4. Implementación y Configuración del SIEM

Los sistemas de gestión de eventos e información de seguridad (SIEM) son herramientas fundamentales para la protección de las infraestructuras tecnológicas en cualquier organización (Microsoft, 2024). Permiten recopilar, analizar y correlacionar eventos y logs de múltiples fuentes dentro de la red, proporcionando una visión unificada del estado de la seguridad. Los SIEM no solo facilitan la detección y respuesta ante incidentes de seguridad en tiempo real, sino que también ofrecen capacidades avanzadas de monitoreo, auditoría y cumplimiento normativo, lo que los convierte en una pieza clave para la gestión de la seguridad en entornos complejos como los de AWS.

Para este proyecto de máster, se ha decidido implementar **Wazuh** como el SIEM de elección. Wazuh ha ganado gran popularidad en los últimos años gracias a su naturaleza 100% open-source, lo que lo hace accesible y flexible para empresas de todos los tamaños. Además, su activa y creciente comunidad global contribuye continuamente al desarrollo y mejora de la plataforma, garantizando que se mantenga a la vanguardia en términos de seguridad y funcionalidad (Wazuh, 2024). Este enfoque colaborativo y su capacidad de integración con AWS lo convierten en la opción ideal para este proyecto, asegurando tanto robustez como eficiencia en la gestión de eventos de seguridad.

La herramienta Wazuh tiene una serie de componentes importantes para su funcionamiento. En la siguiente imagen, se muestra la arquitectura oficial de este sistema:

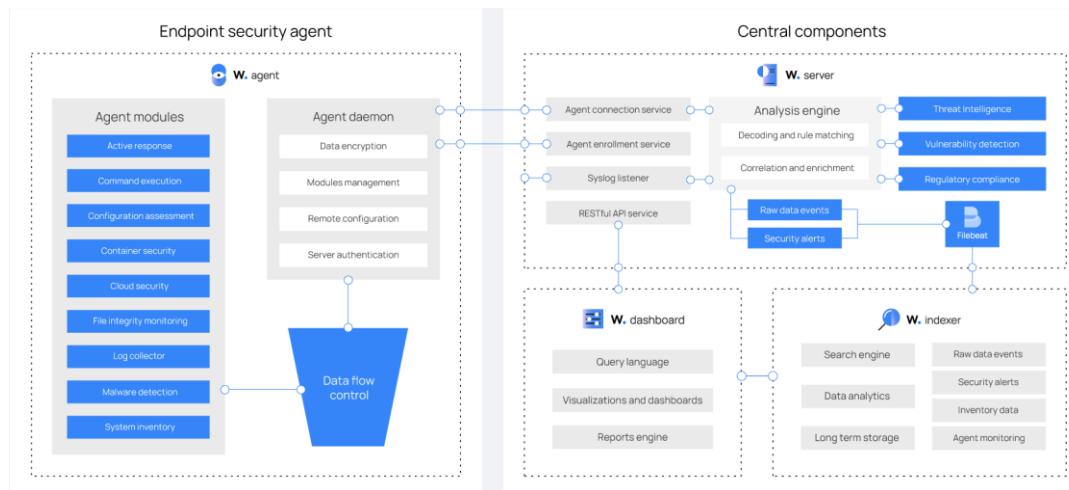


Ilustración 45. Componentes del SIEM Wazuh. Tomado de (Wazuh, 2024).

Teniendo en cuenta los componentes, es indispensable la creación y uso de los siguientes elementos en el trabajo de grado (Wazuh, 2024):

- **Wazuh Indexer:** Motor de búsqueda y análisis altamente escalable que indexa y almacena las alertas generadas por el servidor Wazuh.

- **Wazuh Server:** Analiza los datos de los agentes mediante decodificadores y reglas. Utiliza inteligencia de amenazas para detectar indicadores de compromiso (IOC). Puede gestionar cientos o miles de agentes y escalar horizontalmente como un clúster.
- **Wazuh Dashboard:** Interfaz web para la visualización y análisis de datos. Ofrece dashboards prediseñados para cumplimiento normativo (PCI DSS, GDPR, CIS, HIPAA), monitoreo de integridad de archivos, vulnerabilidades detectadas y más.
- **Wazuh Agents:** Instalados en endpoints (laptops, servidores, instancias en la nube). Proveen capacidades de prevención, detección y respuesta ante amenazas. Compatibles con varios sistemas operativos (Linux, Windows, macOS, etc.).

#### **Arquitectura SIEM para el proyecto de máster:**

Es importante recordar que el proyecto fue realizado y versionado con Git, el software más importante de “Control de Versiones”, que permite llevar a cabo proyectos de programación de forma iterativa y con las mejores prácticas de la industria. Este es el enlace del repositorio Open-Source creado para compartir el proyecto y que futuros ingenieros de ciberseguridad puedan aprender de él:



Ilustración 46. Imagen de mi proyecto SIEM Open-Source para la elaboración del trabajo de grado.

Elaboración propia.

#### **Repository de GitHub (elaboración propia).**

**Enlace:** <https://github.com/san99tiago/aws-cybersecurity-siem>

Tabla 9. Repository de GitHub del proyecto SIEM (aws-cybersecurity-siem). Elaboración propia.

En primer lugar, es importante entender que existen diversos modelos de despliegue de Wazuh en entornos empresariales. Los más utilizados son:

- **Amazon Machine Images:** Permiten desplegar las instancias de EC2 desde una AMI propietaria de Wazuh en el Marketplace.
- **Docker:** Existen imágenes de Docker para desplegar los contenedores de Wazuh. Se suelen proporcionar con Docker-Compose.
- **Kubernetes:** Existen manifestos de Kubernetes apalancados en las imágenes de Docker, para garantizar el despliegue contenerizado distribuido.
- **Instalación desde las fuentes:** Se usan los scripts del código fuente, para generar los binarios de Wazuh en diversos sistemas operativos.
- **Máquinas Virtuales con OVA:** imágenes pre-construidas con compatibilidad hacia los proveedores principales de virtualización, como por ejemplo "Virtual Box".

Para el trabajo de máster, se implementará la solución recomendada para AWS, la cual es a través de las "*Amazon Machine Images*" (AMIs) (Wazuh, 2024). Esta solución ofrece compatibilidad directa con instancias de EC2 y una fácil personalización a través del UserData, en caso de requerir componentes en modo "Clúster".

Si bien los componentes de Wazuh son importantes, también es indispensable obtener un DNS propio, que permita a la empresa (en este caso a mi estructura organizacional simulada), lograr comunicarse con los servidores SIEM de forma segura, encriptada y garantizando los certificados TLS.

Para esto, compré un dominio DNS llamado "[san99tiago.com](http://san99tiago.com)", a través de Route 53, y lo configuré con un esquema multi-account. Este proceso se puede entender en la siguiente imagen:

## MULTI-ACCOUNT DNS SETUP FOR SAN99TIAGO.COM

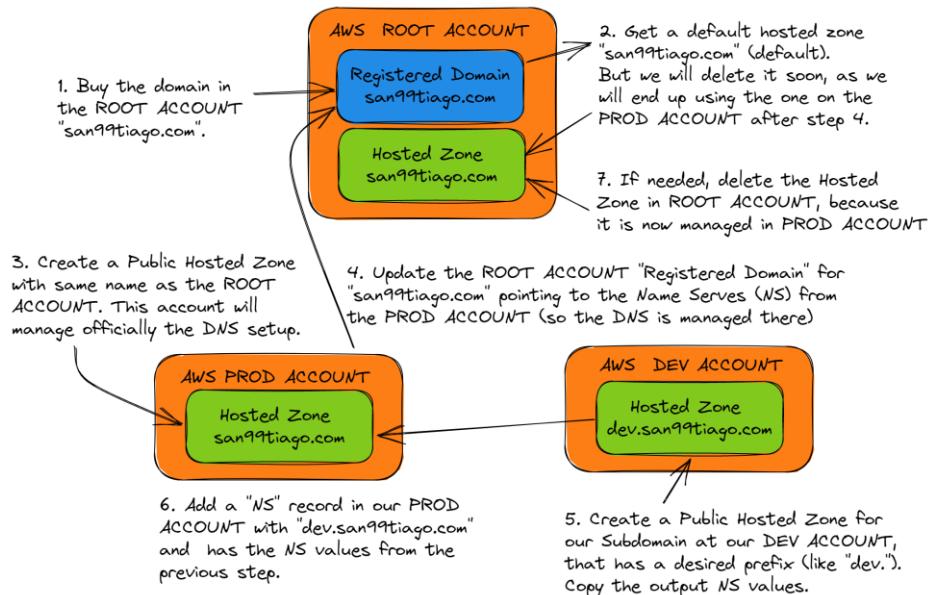


Ilustración 47. Configuración de DNS multi cuenta para AWS con mi dominio san99tiago.com .

Elaboración propia.

Una vez adquirido el dominio , se procedió a diseñar la arquitectura de despliegue del componente SIEM, utilizando un DNS personalizado adaptado a mi caso de uso: **"siem.san99tiago.com"**. Este enfoque tiene la ventaja de permitir el despliegue del SIEM con balanceadores de carga, lo que asegura que, en caso de que las direcciones IP de los servidores Wazuh cambien, el rendimiento y la disponibilidad no se vean afectados. Esto es especialmente importante, ya que un despliegue sin un DNS propio podría experimentar fallos en estas situaciones.

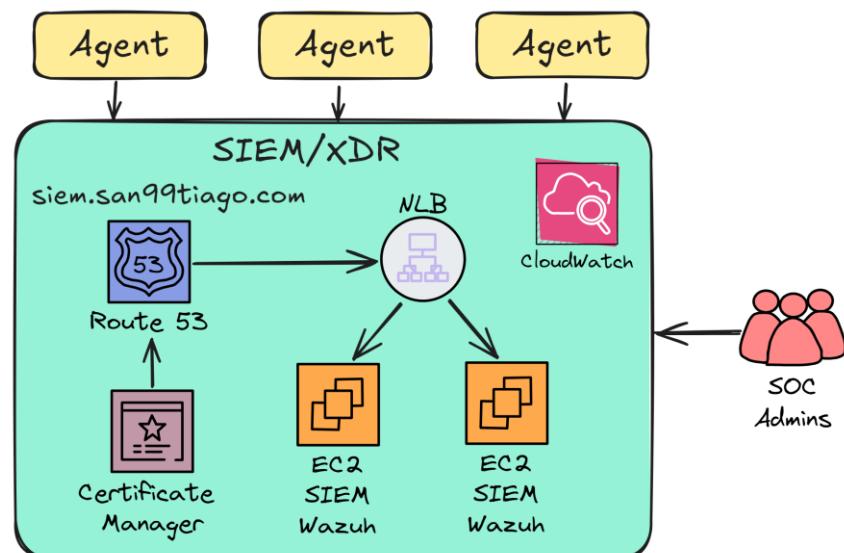


Ilustración 48. Arquitectura en AWS del sistema SIEM basado en Wazuh con DNS personalizado.

Elaboración propia.

Si bien el repositorio tiene muchas líneas de código fuente y es complejo explicarlo de forma detallada en este documento, procederé a indicar los puntos de acceso esenciales para entender los componentes del SIEM en AWS:

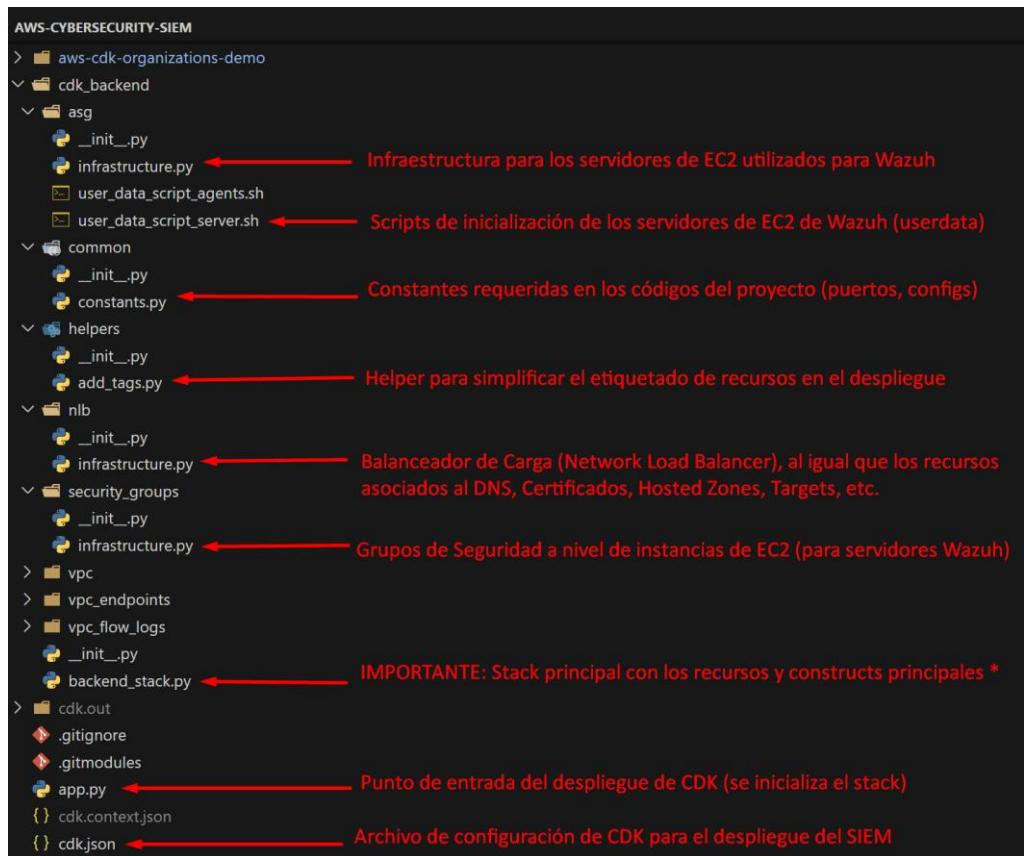


Ilustración 49. Explicación detallada del repositorio propio para el despliegue del SIEM en AWS.

Elaboración propia.

Una vez mostrada la configuración del código, se procede a mostrar el archivo de configuración general (CDK.json), el cual permite personalizar el despliegue del SIEM (Wazuh), según la cuenta de AWS objetivo. En este caso, se desplegará únicamente en la cuenta de “PRODUCCIÓN”, pues el objetivo de este es ser un centralizador de eventos y logs para obtener las mejores prácticas de seguridad en la empresa:

```

1  {
2      "prod": {
3          "siem": {
4              "short_name": "wazuh-siem",
5              "sg_cidrs_list": ["0.0.0.0/0"],
6              "instance_type": "c5a.xlarge",
7              "ami_name": "Wazuh_v4.9.0-1-79ced6c9-1e2d-4f22-ada6-dc528473b3f8",
8              "min_capacity": 1,
9              "max_capacity": 1,
10             "desired_capacity": 1,
11             "hosted_zone_name": "san99tiago.com"
12         }
13     }
14 }

```

Ilustración 50. Configuraciones para despliegue del SIEM con CDK. Elaboración propia.

Una vez se despliega la solución propuesta, se pueden ver los recursos en AWS desde el Stack de CloudFormation resultante, el cual muestra cada uno de los componentes explicados en el transcurso de las secciones previas:

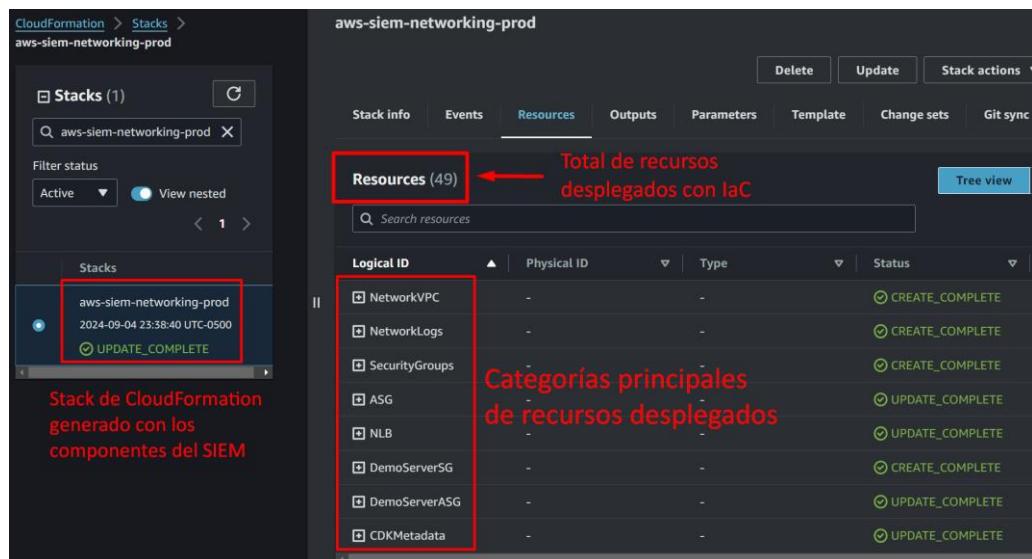


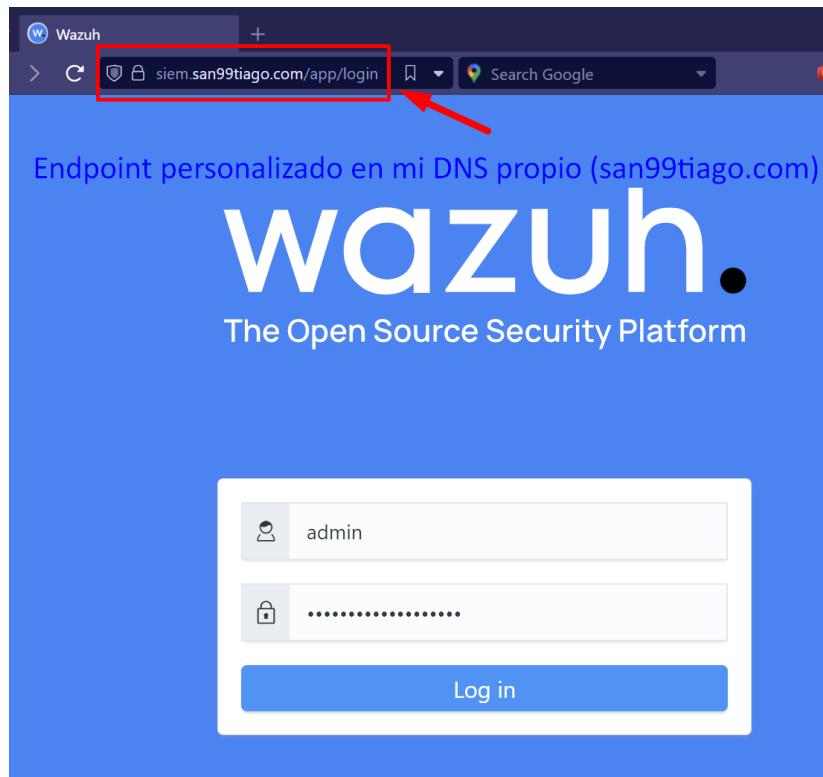
Ilustración 51. Resultado del despliegue del Stack de CloudFormation del SIEM en AWS. Elaboración propia.

Teniendo en cuenta lo anterior, se procede a validar el funcionamiento del SIEM a través del DNS personalizado “siem.san99tiago.com”, con el objetivo de asegurar que cada uno de los componentes internos de la arquitectura de Wazuh esté operando correctamente. Adicionalmente, se verifica que los principales puertos de acceso estén abiertos para garantizar la comunicación adecuada entre los módulos:

- **Dashboard Port:** 443 (puerto del Wazuh Dashboard)
- **Indexer Port:** 9200 (puerto del Wazuh Indexer)
- **Manager Port (Remoted Module):** 1514 (para agentes)
- **Manager Port (Authd Module):** 1515 (para autenticación de agentes)

- **Manager Port (RESTful API Module):** 55000 (para la API REST)

Se valida entonces el acceso al SIEM de la siguiente forma:



*Ilustración 52. Resultado del Dashboard de Wazuh (SIEM) en mi endpoint personalizado siem.san99tiago.com . Elaboración propia.*

Como se puede observar en la Ilustración 56, se logra validar que el SIEM se haya configurado de forma exitosa y con los certificados TLS correctos para mi sitio web “[siem.san99tiago.com](http://siem.san99tiago.com)”.

Una vez desplegado el sistema SIEM, se proceden a agregar los agentes en los servidores desplegados. Esto se puede realizar de dos formas:

- **Manual:** A través de copiar/pegar los scripts de inicialización de agentes de Wazuh.
- **Automatizado:** A través de agregar los scripts en los comandos de Bootstrap de las instancias de EC2.

En este caso, se realizará el proceso automatizado para las instancias Linux, y el proceso manual para las instancias Windows (de esta forma se logra ver ambas opciones en un entorno de trabajo productivo):

Run the following commands to download and install the agent:

**Comando para instalación y configuración de agentes de Windows:**

```
Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.8.2-1.msi -OutFile ${env.tmp}\wazuh-agent; msieexec.exe /i ${env.tmp}\wazuh-agent /q WAZUH_MANAGER='siem.san99tiago.com' WAZUH_AGENT_NAME='WindowsProd_Test_0c0b167dab45e05d2'
```

① Requirements

- You will need administrator privileges to perform this installation.
- PowerShell 3.0 or greater is required.

Keep in mind you need to run this command in a Windows PowerShell terminal.

DNS del sistema  
SIEM de mi proyecto  
de máster!

5 Start the agent:

Comando inicialización agente de Windows!

```
NET START WazuhSvc
```

Ilustración 53. Comandos de instalación de agentes para servidores Windows. Elaboración propia.

A continuación, se procede a ejecutar los comandos en los servidores Windows:

```
PS C:\Users\Administrator> Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.8.2-1.msi -OutFile ${env.tmp}\wazuh-agent; msieexec.exe /i ${env.tmp}\wazuh-agent /q WAZUH_MANAGER='siem.san99tiago.com' WAZUH_AGENT_NAME='WindowsProd_Test_0c0b167dab45e05d2'
PS C:\Users\Administrator> NET START WazuhSvc
The Wazuh service is starting.
The Wazuh service was started successfully.
```

Comandos de instalación de agente e inicio de servicio para Wazuh agent!

Ilustración 54. Ejemplo de comandos de instalación de agentes Windows en servidor productivo.

Elaboración propia.

Para los servidores Linux, se proceden a configurar cada uno de los servidores de EC2 con el "UserData" adecuado para instalar los agentes apuntando al DNS personalizado del SIEM. Esto se logra con ayuda de los siguientes scripts:

```

# AGENT INSTALLATION LINUX (WAZUH)
echo "----- Preparing to install Wazuh Agent -----"

# Set the URL for the token and metadata services
TOKEN_URL="http://169.254.169.254/latest/api/token"
METADATA_URL="http://169.254.169.254/latest/meta-data"

# Get a token for the metadata service (valid for 21600 seconds, or 6 hours)
TOKEN=$(curl -s -X PUT "$TOKEN_URL" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600")

# Get the instance ID to have unique names for the agents in the SIEM dashboard
EC2_INSTANCE_ID=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" "$METADATA_URL/instance-id")

WAZUH_AGENT_NAME="workflow-server.${EC2_INSTANCE_ID}"
WAZUH_SIEM_ENDPOINT="siem.san99tiago.com"

echo "----- Installing Wazuh Agent -----"

curl -o wazuh-agent-4.8.2-1.x86_64.rpm https://packages.wazuh.com/4.x/yum/wazuh-agent-4.8.2-1.x86_64.rpm && sudo WAZUH_MANAGER="${WAZUH_SIEM_ENDPOINT}" WAZUH_AGENT_NAME="$WAZUH_AGENT_NAME" rpm -ihv wazuh-agent-4.8.2-1.x86_64.rpm

sudo systemctl daemon-reload
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent

```

Ilustración 55. Scripts de instalación de agentes de Wazuh en servidores Linux de EC2. Elaboración propia.

Una vez configurados todos los agentes de la empresa, se procede a analizar el sistema SIEM y empezar las validaciones de seguridad.

En este punto del proyecto, se puede validar que hay una observabilidad general a nivel de múltiples cuentas de AWS, diversas redes encriptadas, y también la posibilidad de monitorear agentes “en tierra”, es decir, desde servidores ubicados “On-Premise”.

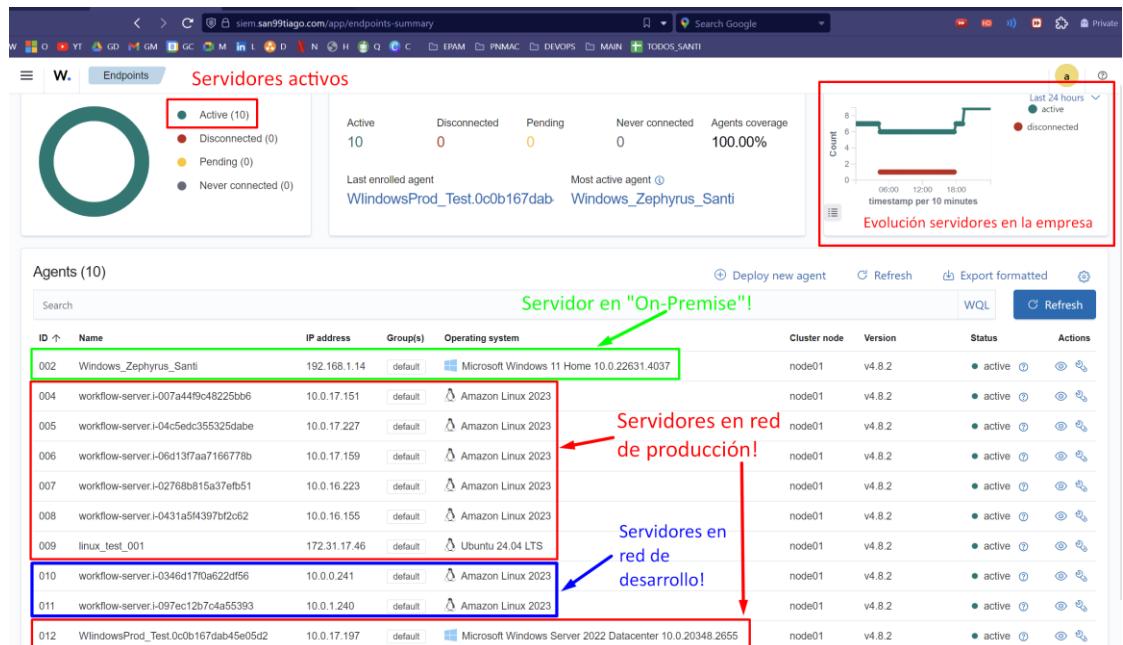


Ilustración 56. Resultado de sistema SIEM con servidores detectados y protegidos. Elaboración propia.

Como se puede observar en la Ilustración 56, se logran detectar los 10 servidores simulados de la empresa. Todos con diversos sistemas operativos, redes y configuraciones de paquetes/scripts diversas.

Esto muestra que el SIEM ha sido correctamente configurado, y la empresa puede crear sus propias reglas de correlación e inteligencia para la detección de anomalías o ataques. Es común que las empresas adopten el SIEM de forma iterativa, pues existe una curva de aprendizaje en el uso y administración de los componentes de software mostrados en este proyecto.

### Validación de Vulnerabilidades a través del SIEM:

Una de las funcionalidades más valiosas, es la detección de vulnerabilidad en los servidores de la empresa. Esto se puede lograr a través del Wazuh, con ayuda de la funcionalidad de “Vulnerability Detection”, la cual ofrece un análisis y detección en tiempo real de las librerías y paquetes instalados para cada uno de los agentes conectados al SIEM (Wazuh, 2024).

Para esta prueba, se procede a validar las vulnerabilidades de los 10 servidores de prueba que se agregaron en las secciones anteriores, mostrando cada una de las posibles CVE que podrían estar siendo vectores de ataque a los endpoints productivos:



Ilustración 57. Detección de vulnerabilidades de forma activa en el Dashboard de Wazuh. Elaboración propia.

Con la información obtenida en la Ilustración 57, los expertos en seguridad de la empresa pueden aplicar las actualizaciones y parches necesarios para cada uno de estos CVEs mostrados gracias al SIEM.

### Simulación de ataque a servidor conectado con el SIEM:

En esta etapa del proyecto, se procede a simular un ataque a uno de los servidores monitoreados por el SIEM. En este caso, a un servidor Windows desplegado en la red productiva, en donde un atacante tuvo acceso.

Se procede a realizar el ataque (simulado) de la siguiente forma:

- Atacante obtiene acceso a la red de producción.
- Atacante tiene conocimiento del endpoint público de la instancia Windows productiva.
- Atacante implementa un ataque de fuerza bruta hacia el servidor Windows con ayuda de la herramienta “*Hydra*”.
- Se espera que una vez se comience el ataque, el servidor central SIEM con Wazuh, logre detectar esta actividad como un “*Threat Hunting*”, generando una alerta basada en las reglas del SIEM.
- Se deben aplicar las medidas de contención y protección necesarias para el ataque.
- Se debe validar que el ataque haya sido correctamente parado, mediante la herramienta SIEM de Wazuh.

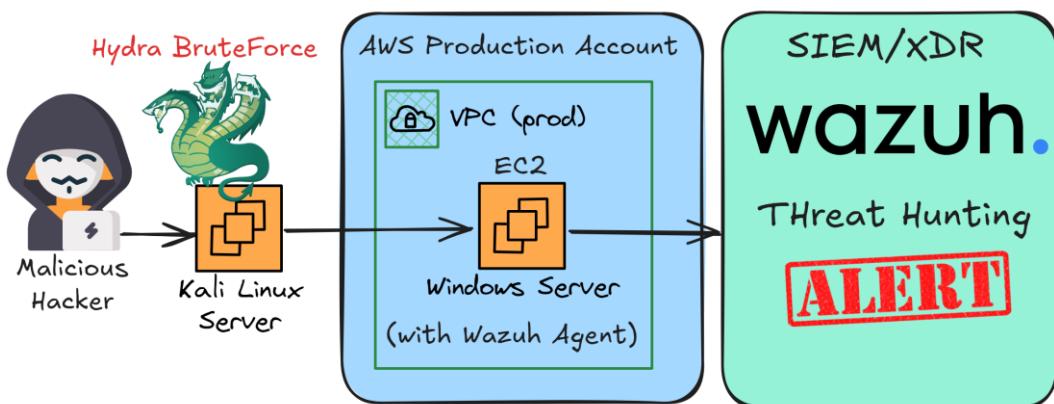


Ilustración 58. Simulación de ataque a servidor Windows productivo en AWS protegido con SIEM Wazuh.  
Elaboración propia.

En primer lugar, se analiza cómo se ve el dashboard de Wazuh, antes de efectuar el ataque:

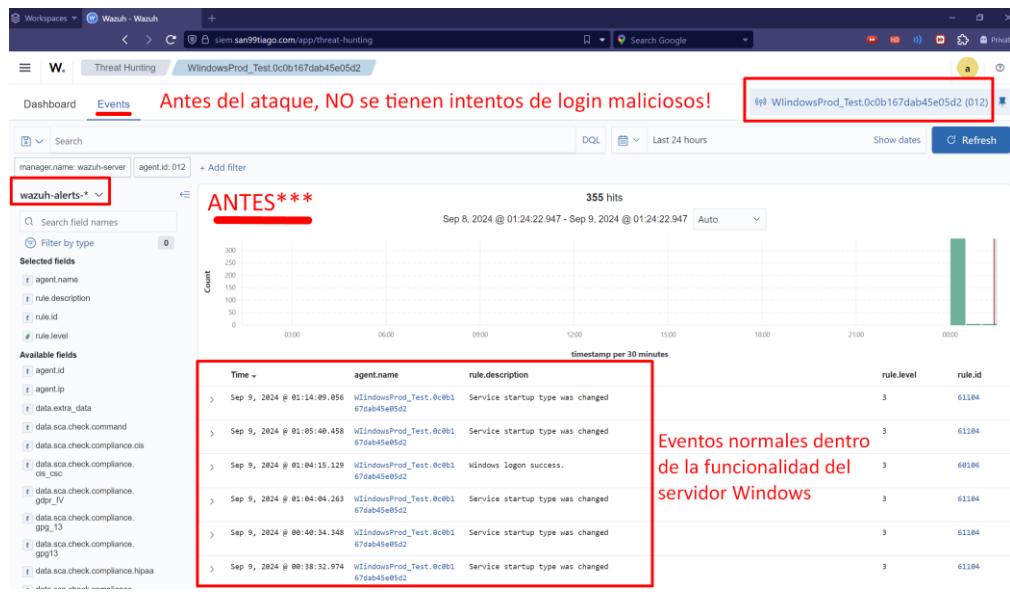


Ilustración 59. Visualización del Dashboard de Wazuh ANTES del ataque de fuerza bruta hacia el servidores Windows. Elaboración propia.

Ahora, se procede a realizar el ataque desde el servidor atacante malicioso:

```
/mnt/c/Users/santi/Documents/PROGRAMMING/deleteMe > sudo hydra -l badguy -P password.txt rdp://3.85.126.65
Hydra v9.2 (c) 2021 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or
thinks anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-09 01:36:00
[WARNING] rdp servers often don't like many connections, use -t 1 or -t 4 to reduce the number of parallel connections and
[INFO] Reduced number of tasks to 4 (rdp does not like many parallel connections)
[WARNING] the rdp module is experimental. Please test, report - and if possible, fix.
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to
[DATA] max 4 tasks per 1 server, overall 4 tasks, 1000 login tries (1:l:p:1000), ~250 tries per task
[DATA] attacking rdp://3.85.126.65:3389/
```

Ataque de fuerza bruta al servidor Windows a través de Hydra

Ilustración 60. Ejecución del ataque (simulado) con herramienta Hydra al servidor Windows productivo. Elaboración propia.

Una vez en ejecución, se procede a ir al Dashboard de Wazuh, en donde se analiza cómo se puede detectar el ataque del servidor Windows:

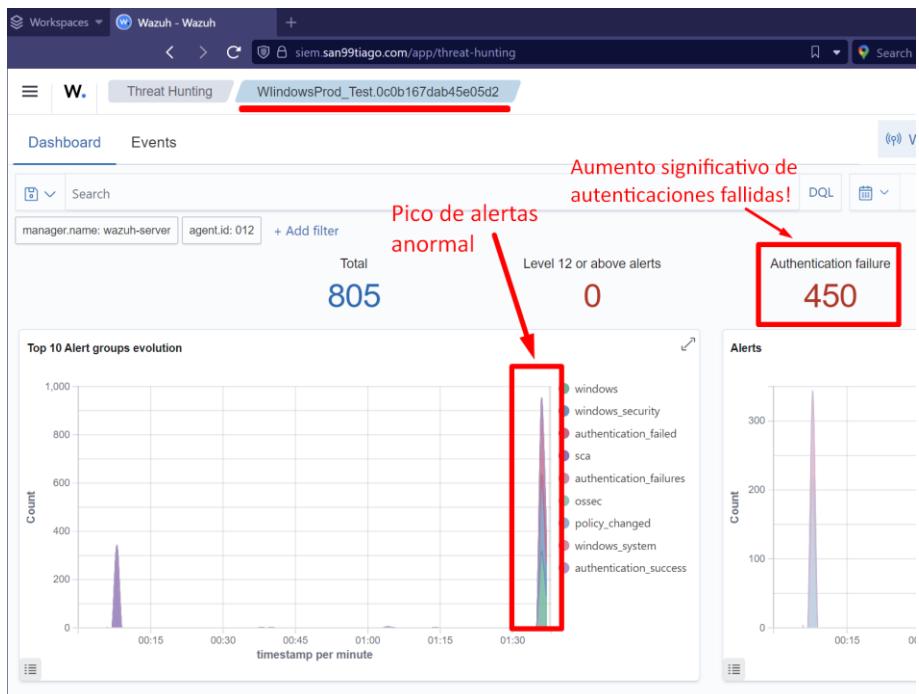


Ilustración 61. Dashboard de Wazuh durante el ataque al servidor Windows. Elaboración propia.

De igual forma, al analizar los eventos, se puede ver claramente que son asociados al ataque de fuerza bruta:

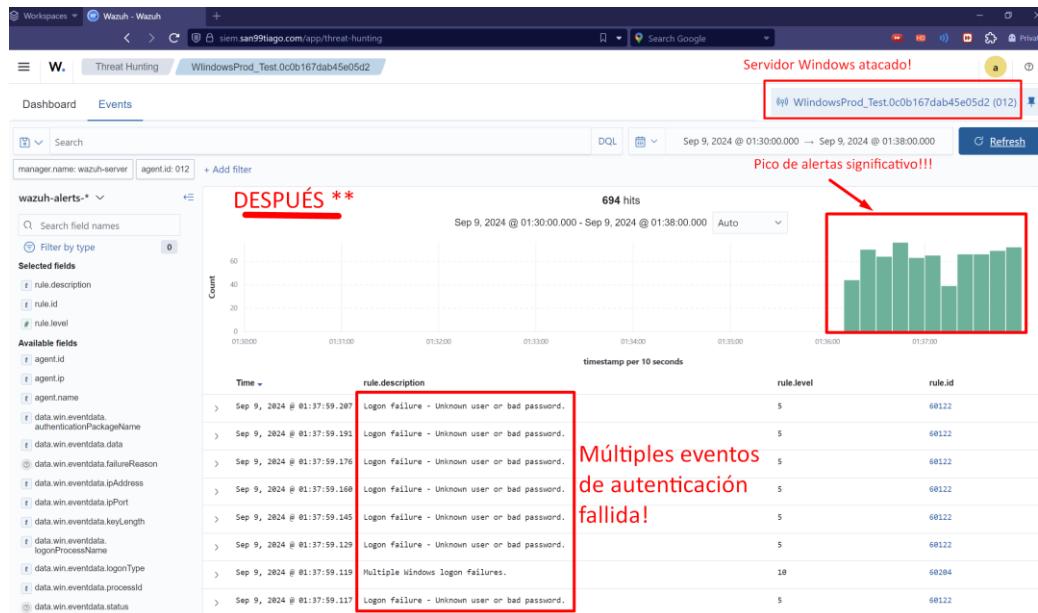


Ilustración 62. Eventos asociados al ataque de fuerza bruta hacia el servidor Windows. Elaboración propia.

Teniendo en cuenta estas alertas, el equipo de ciberseguridad puede proceder a aplicar una acción reactiva, como por ejemplo actualizar los firewalls y así, proteger al equipo Windows. Esto se puede hacer en AWS a través de la actualización de los Security Groups, o

incluso a través de un WAF en caso de ser un ataque más robusto. A continuación se muestra la solución al ataque simulado en AWS:

**Edit inbound rules** Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

**Inbound rules** Info

**Inbound rule 1**

Security group rule ID	Type	Protocol
sgr-0b7f80fb735575b4c	RDP	TCP

**Port range** Info      **Source type** Info      **Source** Info

3389      Custom      10.0.16.0/20 X

**Delete**

Ilustración 63. Medida reactiva al ataque al servidor Windows a través de actualización de Security Groups. Elaboración propia.

Finalmente, se puede validar que el ataque haya sido correctamente contenido, mediante el Dashboard de observabilidad de eventos de Wazuh. A continuación, se muestra cómo se ve el sistema al lograr la protección correcta frente al ataque:

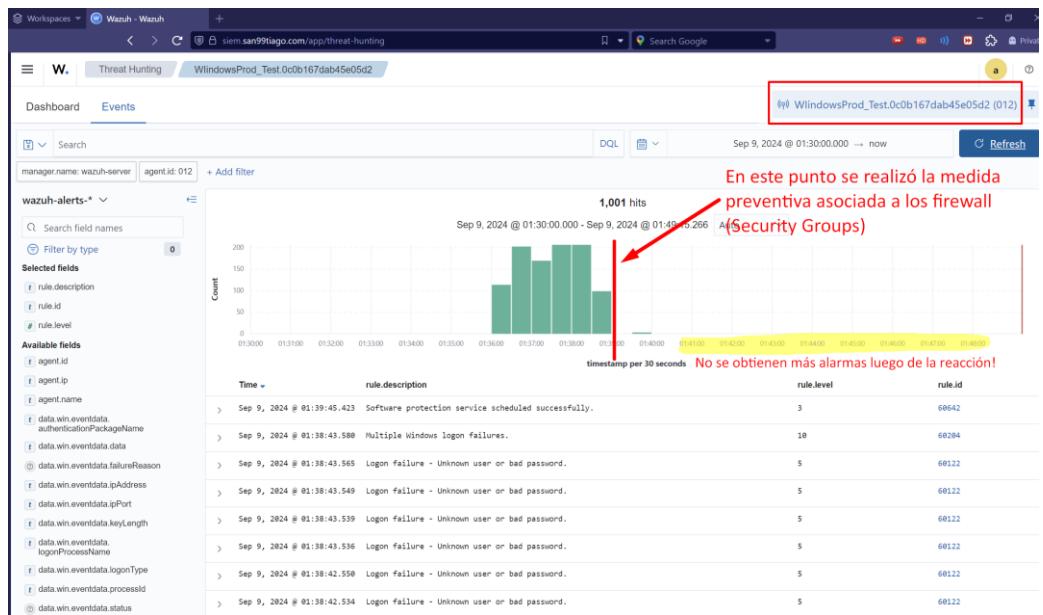


Ilustración 64. Dashboard de Wazuh luego de la contención del ataque exitosa. Elaboración propia.

Se logra ver cómo Wazuh es altamente efectivo para generar alarmas de posibles ataques y eventos maliciosos. La herramienta ofrece reglas mucho más avanzadas, que se podrán configurar según las necesidades de los sistemas y la experticia del equipo de seguridad que sea responsable del manejo del SIEM.

## CONCLUSIONES

En un entorno donde las amenazas ciberneticas crecen exponencialmente, mantenerse a la vanguardia en el uso de sistemas SIEM es esencial para garantizar la protección de las infraestructuras en la nube. La seguridad en la nube ha adquirido una importancia crítica para las organizaciones, dado que la rápida evolución de los componentes de software y las arquitecturas cloud-native generan continuamente nuevos desafíos para los ingenieros de seguridad. Es por esto, que adoptar las mejores prácticas de ciberseguridad y desplegar soluciones robustas, como un SIEM distribuido, es fundamental para enfrentar estos nuevos paradigmas tecnológicos y proteger a las empresas de ataques ciberneticos avanzados.

Este trabajo de máster representa un punto de partida hacia la implementación de un sistema SIEM productivo y eficaz; sin embargo, aún quedan muchas áreas por explorar. La integración de inteligencia artificial para generar reglas adaptativas y la ampliación de integraciones hacia otros componentes y servicios que no fueron abarcados en este proyecto ofrecen un gran potencial para continuar evolucionando el esquema de seguridad, pues la seguridad abarca muchísimos más aspectos a considerar en entornos modernos. Esto subraya la importancia de seguir innovando y adaptando las soluciones SIEM a las crecientes exigencias del ecosistema de ciberseguridad.

Se logró cumplir con el objetivo principal de diseñar y desplegar un sistema SIEM avanzado, distribuido y alineado con las mejores prácticas de ciberseguridad para proteger cargas de trabajo productivas y no productivas en la nube de Amazon Web Services. Utilizando Wazuh como herramienta principal, se validó su funcionalidad en un entorno con múltiples servidores desplegados en la nube, obteniendo resultados satisfactorios en la centralización de logs, detección de vulnerabilidades y respuesta proactiva a ataques simulados. Este trabajo establece una base sólida para futuros proyectos que busquen reforzar la seguridad en la nube.

Para concluir, este proyecto reafirma la importancia de construir sistemas resilientes y seguros en la nube, comprendiendo que la prevención y respuesta ante incidentes es una tarea continua. Como bien dice Werner Vogels, CTO de Amazon: "Todo falla, todo el tiempo". Por ello, la clave está en prepararse para esas fallas y garantizar que, incluso en los momentos más críticos, las empresas puedan mantener la continuidad y seguridad de sus operaciones.

## REFERENCIAS

- Amazon Web Services. (2024). *Shared Responsibility Model*. Obtenido de <https://aws.amazon.com/es/compliance/shared-responsibility-model/>
- Amazon Web Services. (2024). *What is AWS Organizations?* Obtenido de [https://docs.aws.amazon.com/organizations/latest/userguide/orgs\\_introduction.html](https://docs.aws.amazon.com/organizations/latest/userguide/orgs_introduction.html)
- AWS. (2024). Obtenido de Cloud Computing Services - Amazon Web Services (AWS): <https://aws.amazon.com>
- AWS. (2024). *AWS Cloud Development Kit*. Obtenido de <https://aws.amazon.com/cdk/>
- AWS. (2024). *Subnets for your VPC*. Obtenido de <https://docs.aws.amazon.com/vpc/latest/userguide/configure-subnets.html>
- AWS. (08 de 2024). *Terminology and concepts for AWS Organizations*. Obtenido de [https://docs.aws.amazon.com/organizations/latest/userguide/orgs\\_getting-started\\_concepts.html](https://docs.aws.amazon.com/organizations/latest/userguide/orgs_getting-started_concepts.html)
- AWS. (2024). *What is Amazon VPC?* Obtenido de <https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html>
- AWS. (2024). *What is the AWS CDK?* Obtenido de <https://docs.aws.amazon.com/cdk/v2/guide/home.html>
- Check Point. (2024). *Check Point Research Reports Highest Increase of Global Cyber Attacks seen in last two years – a 30% Increase in Q2 2024 Global Cyber Attacks*. Obtenido de <https://blog.checkpoint.com/research/check-point-research-reports-highest-increase-of-global-cyber-attacks-seen-in-last-two-years-a-30-increase-in-q2-2024-global-cyber-attacks>
- CISCO. (August de 2024). *What Is Network Security?* Obtenido de <https://www.cisco.com/c/en/us/products/security/what-is-network-security.html>
- CloudFlare. (2024). *What is TLS (Transport Layer Security)?* Obtenido de <https://www.cloudflare.com/learning/ssl/transport-layer-security-tls/>
- Derek, C. (10 de March de 2010). *How to Use a Morphological Matrix to Generate Ideas*. Obtenido de <https://innovationmanagement.se/2010/03/10/how-to-use-a-morphological-matrix-to-generate-ideas/>
- Dieter, Schmidt, & Linda. (2012). *Engineering Design by Dieter, George, Schmidt, Linda. (McGraw-Hill Science/Engineering/Math,2012) [Hardcover] 5th Edition*. McGraw Hill. Obtenido de <https://www.amazon.com/Engineering-Schmidt-McGraw-Hill-Science-Hardcover/dp/B00E2RP60Q>

- Fortune BI. (19 de August de 2024). *Cloud Computing Market Size, Share & Industry Analysis, By Type*. Obtenido de <https://www.fortunebusinessinsights.com/cloud-computing-market-102697>
- Gartner. (04 de Diciembre de 2023). *Magic Quadrant for Strategic Cloud Platform Services*. Obtenido de <https://aws.amazon.com/resources/analyst-reports/gartner/global-mq-ardm-23-magic-quadrant-for-strategic-cloud-platform-services>
- Gartner. (2024). *Magic Quadrant for Strategic Cloud Platform Services 2023*. Obtenido de <https://aws.amazon.com/blogs/aws/read-the-2023-gartner-magic-quadrant-for-strategic-cloud-platform-services/>
- GitHub. (2024). *GitHub Actions documentation*. Obtenido de <https://docs.github.com/en/actions>
- Google. (2024). *Google Cloud: Cloud Computing Services*. Obtenido de <https://cloud.google.com>
- Google Cloud Platform. (2024). *Advantages of Cloud Computing*. Obtenido de <https://cloud.google.com/learn/advantages-of-cloud-computing>
- Hashicorp. (2024). *Automate infrastructure on any cloud with Terraform*. Obtenido de <https://www.terraform.io>
- Hauser, J. (1988). *The House of Quality*. Obtenido de <https://hbr.org/1988/05/the-house-of-quality>
- IBM. (2024). *What is security information and event management (SIEM)?* . Obtenido de <https://www.ibm.com/topics/siem>
- IBM. (s.f.). *IBM QRadar Suite*. Obtenido de <https://www.ibm.com/qradar>
- Lankford, B., Andrew, B., & Rice, S. (6 de December de 2022). *AWS re:Invent 2022 - Best practices for organizing and operating on AWS (COP305)*. Obtenido de <https://youtu.be/Eeyd6BDpucw>
- Martínez, A. (2017). *NP MASTERCLASS: Ciberseguridad SIEM*. Obtenido de <https://youtu.be/uhxhJJJSQXM>
- McKinsey Technology Council. (16 de Julio de 2024). *McKinsey Technology Trends Outlook 2024*. Obtenido de <https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-top-trends-in-tech>
- Microsoft. (2024). *Microsoft Azure: Cloud Computing Services*. Obtenido de <https://azure.microsoft.com/en-us>
- Microsoft. (2024). *What is SIEM?* Obtenido de <https://www.microsoft.com/en-us/security/business/security-101/what-is-siem>

Rainbow Secure. (8 de January de 2024). *Navigating the Cloud: Challenges, Threats, and Best Practices in Cloud Security*. Obtenido de <https://www.linkedin.com/pulse/navigating-cloud-challenges-threats-best-practices-security-oijye>

Splunk. (2024). *Splunk / The Key to Enterprise Resilience*. Obtenido de <https://www.splunk.com>

Wazuh. (2024). *Getting started with Wazuh Components*. Obtenido de <https://documentation.wazuh.com/current/getting-started/components/index.html>

Wazuh. (2024). *Installation alternatives: Amazon Machine Images (AMI)*. Obtenido de <https://documentation.wazuh.com/current/deployment-options/amazon-machine-images/amazon-machine-images.html>

Wazuh. (2024). *The Open Source*. Obtenido de <https://wazuh.com>

Wazuh. (2024). *Wazuh Documentation*. Obtenido de <https://documentation.wazuh.com/current/index.html>

Wazuh. (2024). *Wazuh Vulnerability detection*. Obtenido de <https://documentation.wazuh.com/current/proof-of-concept-guide/poc-vulnerability-detection.html>

World Economic Forum. (03 de Abril de 2024). *Key strategies for building cyber resilience in 2024*. Obtenido de <https://www.weforum.org/agenda/2024/04/cybersecurity-key-strategies-cyber-resilience-2024/>