



TRABAJO DE GRADO
Seminario
Implementación de Arquitectura en Amazon Web Services (AWS)

Corporación Universitaria Remington
Facultad de Ingeniería
Seminario AWS Cloud
Ingeniería de Sistemas

Cristhian Guillermo Marulanda Herrera
Julian Andres Martinez Sanchez

Juan Pablo Berrío López (docente del seminario).
Seminario De Grado
2024

Contenido

Resumen	3
Palabras clave	4
Marco conceptual	5
Desarrollo e implementación del aprendizaje	6
Caso de estudio: decisión estratégica de infraestructura IT para TechSolutions S.A.	6
Contexto de la empresa	6
Requisitos y consideraciones	6
Tareas para los estudiantes	7
Puntos clave para la discusión.....	8
Evaluación de riesgos y seguridad:	8
Riesgos en la nube:.....	9
Escalabilidad y flexibilidad:.....	10
Impacto en el personal y operaciones:	10
Implementación de instancia EC2.....	10
Ingreso por MobaXterm	12
Instalación de Docker	13
Creación de Contenedores.....	14
Descarga e Instalación de la Aplicación	15
Instalación de Nginx.....	16
Implementación de S3	18
Alojamiento de Sitios Web Estáticos	19
Carga de Imágenes e Index	20
Creación de Base de Datos.....	20
Configuración de la Base de Datos	22
Instalación de heidiSQL	23
Creación de Nuevo Usuario y Grupo para Permisos Pertinentes.....	25
Creación de Balanceadores de Carga en EC2	25
Creación de Copia de Instancia.....	27
EC2 Auto Scaling.....	29
Conclusiones	30
Agradecimientos.....	31
Referencias Bibliográficas	32
Enlace de videos.....	33

Resumen

En la primera entrega del seminario se realizó un estudio de la empresa TechSolutions S.A. la cual está experimentando un rápido crecimiento desarrollando software y brindando soluciones tecnológicas a clientes. La empresa necesita expandir su capacidad de procesamiento y almacenamiento de datos.

Una expansión de un Datacenter local conlleva una inversión adicional, tiempos y recurso; la compra de licencias, seguros de protección, y adecuación para cumplir con las normas.

En cambio, al migrar a un servidor en la nube minimizas gastos operacionales una vez contratado el servicio con el proveedor ya que tendrá un gran beneficio en tiempo real según la demanda teniendo en cuenta la optimización de recursos, al contratar con una compañía como AWS o AZURE puedes establecer un control absoluto de las operaciones delegando mantenimientos y actualización con el proveedor.

En la segunda entrega, se implementa una instancia utilizando EC2 con dos contenedores y un balanceador de carga se realiza directamente en el proveedor de servicios de Amazon Web Services (AWS). Primero se debe crear la instancia usando EC2, para luego dirigirse a MobaXterm para ingresar por medio de SSH e instalar el Docker.

Aquí se definen los contenedores con las configuraciones pertinentes de los puertos para el uso deseado. Se agregan los puertos de conexión en EC2. Se descarga la aplicación y se verifica el estado del contenedor. Se crea el balanceador de carga Nginx y se configura para distribuir la carga entre los dos contenedores eficientemente y se realiza la prueba.

La implementación usando S3 con el sitio estático se crea el bucket para uso general tomando en cuenta los permisos públicos para el acceso de contenido.

La implementación de la arquitectura con AWS aprovecha un balanceador de carga de aplicación que distribuya el tráfico generado entre las dos instancias. Cada una de las instancias

utiliza un balanceador de carga de los contenedores internos que sirve para administrar y a su vez optimizar la distribución de las solicitudes ejecutadas dentro de ellas.

La configuración creada de doble capa de balanceo de carga nos permite garantizar una escalabilidad, alta disponibilidad y rendimiento con las aplicaciones implementadas.

Palabras clave

Arquitectura AWS, Optimización de Recursos, Balanceador de Carga de Aplicaciones, Tolerancia a Fallos, Bucket.

Marco conceptual

Datacenter: Es un centro de datos a gran escala gestionado y operado por un proveedor de servicios el cual se encarga del almacenamiento y la distribución de datos.

Escalabilidad: Es la capacidad de ampliación y adaptación de un sistema con respecto al rendimiento y crecimiento de la empresa.

Infraestructura: Es un conjunto de software, hardware, redes y servicios que están conectados para garantizar el funcionamiento de los sistemas y las comunicaciones dentro de la empresa.

Flexibilidad: Capacidad del producto que le permite adaptarse de forma eficiente a diferentes entornos determinados.

Amazon EC2: Es un servicio web que proporciona unos recursos del servidor y brinda servicios en la nube ya que este te ayuda, a crear y ejecutar cualquier aplicación.

Balancedador de Carga: Es un componente o dispositivo que distribuye equitativamente el tráfico, las solicitudes de clientes, y las cargas de trabajo.

MobaXterm: Es un terminal mejorado para Windows el cual incluye varias herramientas de computación remota y todos los paquetes esenciales.

SSH: es un protocolo de red que permite acceso remoto principalmente con una conexión directamente con la maquina la cual accedemos por medios de líneas de comandos.

Docker: es un sistema operativo para el manejo de contenedores de manera similar a una máquina virtual ya que es un proyecto de código abierto el cual proporciona una capa adicional de abstracción en múltiples sistemas operativos.

Contenedores: Son paquetes ligeros que incluyen el código de aplicaciones, sus bibliotecas y archivos de configuración necesarios para su ejecución.

Puertos: Es un punto de conexión entre una computadora y dispositivos internos o externos los cuales permiten abstracciones de datos a través de virtualizaciones.

Nginx: Es un servidor web que también actúa como balanceador de carga, es una estructura del software lo cual permite el procesamiento de muchas solicitudes al mismo tiempo.

ACL: es una lista de control de acceso de red que permite o deniega el tráfico entrante y saliente del servidor.

Desarrollo e implementación del aprendizaje

Caso de estudio: decisión estratégica de infraestructura IT para TechSolutions S.A.

Contexto de la empresa

TechSolutions S.A. es una empresa mediana dedicada al desarrollo de software a medida y soluciones tecnológicas para clientes de diversas industrias. Actualmente, la empresa cuenta con una infraestructura IT básica, pero está experimentando un rápido crecimiento y necesita ampliar y modernizar su capacidad de procesamiento y almacenamiento de datos.

TechSolutions S.A. se enfrenta a la decisión de montar un nuevo datacenter local o migrar sus servicios a la nube. La empresa tiene alrededor de 150 empleados, de los cuales 50 son desarrolladores de software que requieren un entorno robusto y flexible para sus actividades diarias. Los proyectos de la empresa incluyen desde aplicaciones móviles hasta sistemas complejos de gestión empresarial, lo que implica una variabilidad considerable en las necesidades de procesamiento y almacenamiento.

Requisitos y consideraciones

1. Costo:

- Datacenter Local: Costos iniciales elevados (adquisición de hardware, instalación, acondicionamiento del espacio, seguridad, etc.), pero costos operativos controlados a largo plazo.
- Nube: Costos iniciales bajos (pago por uso), pero con posibles incrementos a medida que aumentan las necesidades de recursos y almacenamiento.

2. Escalabilidad:

- Datacenter Local: Limitada por la capacidad física y el hardware disponible. La expansión requiere inversiones adicionales en infraestructura.
- Nube: Altamente escalable. Se pueden añadir o reducir recursos según las necesidades del momento sin inversión inicial adicional.

3. Seguridad y cumplimiento:

- Datacenter Local: Control total sobre la seguridad física y lógica. Cumplimiento de normativas específicas puede ser gestionado internamente.
- Nube: Depende de las políticas de seguridad del proveedor de nube. A menudo ofrecen altos estándares de seguridad y cumplimiento, pero puede haber preocupaciones sobre la ubicación y acceso a los datos.

4. Mantenimiento y operaciones:

- Datacenter Local: Requiere un equipo interno dedicado al mantenimiento y soporte de la infraestructura.
- Nube: El mantenimiento de la infraestructura subyacente es gestionado por el proveedor de servicios en la nube.

5. Desempeño y conectividad:

- Datacenter Local: Desempeño altamente controlable y optimizable para necesidades específicas de la empresa.
- Nube: Desempeño dependiente de la conectividad a internet y del proveedor de servicios. Latencia puede ser un factor a considerar.

Tareas para los estudiantes

1. Evaluación de riesgos y seguridad:

- Analizar los riesgos asociados a cada opción, incluyendo seguridad, cumplimiento de normativas y posibles vulnerabilidades.
- Proponer medidas para mitigar estos riesgos.

2. Escalabilidad y flexibilidad:

- Evaluar cómo cada opción se adapta a las necesidades de crecimiento y flexibilidad de la empresa.

- Proponer estrategias para manejar picos de demanda y variabilidad en las necesidades de recursos.

3. Impacto en el personal y operaciones:

- Considerar el impacto en el personal IT de la empresa, incluyendo la necesidad de formación y cambios en las responsabilidades.
- Evaluar cómo cada opción afecta las operaciones diarias de la empresa y la productividad de los empleados.

4. Recomendación final:

- Basándose en los análisis anteriores, formular una recomendación bien fundamentada sobre cuál es la mejor opción para TechSolutions S.A.

Puntos clave para la discusión

- Flexibilidad vs. Control: Evaluar la importancia de tener control total sobre la infraestructura frente a la flexibilidad y facilidad de escalado de la nube.
- Innovación y Futuro: Considerar cómo las tendencias tecnológicas futuras podrían influir en la decisión.
- Experiencia del Usuario: Cómo cada opción impacta la experiencia de los desarrolladores y usuarios finales de los servicios.

Evaluación de riesgos y seguridad:

Al crear un data center local, los gastos serán más elevados, puesto que debemos de contratar un software de seguridad con licencia, adecuación de un área designada, sistemas antiincendios, contar con un sistema completo de aire acondicionado y protección ante desastres naturales.

Al tener una completa responsabilidad, es importante tomar en cuenta medidas que cumplan con la norma ISO 22301, para implementar sistemas de detección de incendios y sistemas de alimentación ininterrumpida (SAI), y crear respaldos exteriores en lugares seguros.

Adquirir seguros contra desastres naturales, pérdidas accidentales, daños súbitos e imprevistos. Garantizar la calidad del servicio continuo por medio de un proveedor seguro que realice mantenimientos que cumplan con la norma ISO 20000.

Evitar vulnerabilidades en los sistemas de información invirtiendo en firewalls, antivirus, sistemas de detección de intrusos y auditorías regulares de seguridad para evitar la filtración o pérdida de datos cumpliendo con la Ley 1273 de 2009 y 1581 de 2012 que protejan y aseguren los datos y sistemas de la empresa.

Riesgos en la nube:

El proveedor se encarga de proporcionar y cumplir con las medidas de seguridad y el área designada para el almacenamiento de datos.

La empresa carece de control de la infraestructura y se enfrenta a interrupciones de los servicios y/o cambios en las políticas de manejo por parte del proveedor que escoja a lo largo del uso de esta.

También se expone a vulnerabilidad, puesto que los datos se encuentran en manos de terceros, y deben tener en cuenta la Ley 1581 de 2012 para asegurar la protección de datos personales para poder mitigar estos riesgos.

En el caso de cumplimiento de normas debemos escoger el proveedor con la mayor certificación en seguridad en las normas ISO/IEC 27001, etc. Para tener una seguridad extensa en los datos almacenados.

Es fundamental contar con un proveedor que tenga certificaciones en seguridad a nivel internacional como la ISO 27001 y cumpla con las leyes nacionales en Colombia para la protección de datos personales.

Establecer y detallar bien los acuerdos de nivel de servicio (ANS) para definir los procedimientos de seguridad, soporte técnico, y disponibilidad de amplias conexiones a internet y backups de datos en varias ubicaciones para evitar interrupciones del servicio.

Escalabilidad y flexibilidad:

Datacenter Local: La escalabilidad puede verse afectada por el hardware a mano y la disposición del espacio físico previniendo la expansión a futuro de la empresa. Se requiere una inversión extra para acomodar la infraestructura que cumplan con la demanda, lo cual necesita de tiempo para realizarla.

Se puede implementar un diseño modular para potencializar sus operaciones proporcionando una estructura flexible y eficiente que permita agregar procesos o adquirir una nube híbrida para cargas temporales.

Nube: Se ajusta a la demanda en tiempo real, brindando una flexibilidad y escalabilidad que beneficia a la empresa con sus trabajos variables. Es recomendable usar el servicio de pago por uso de Amazon Web Services (Azure como alternativa), de esta manera optimizar recursos. El monitoreo continuo permitirá establecer un control de consumos en donde se identifiquen las optimizaciones y reducir gastos innecesarios.

Impacto en el personal y operaciones:

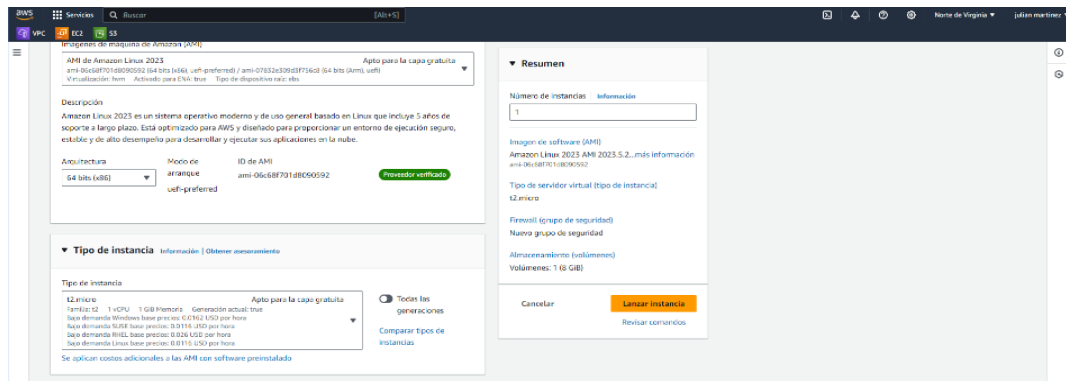
Datacenter Local: La gestión de personal especializado en seguridad, redes, almacenamiento y servidores proveen gastos en contratación. Los mantenimientos y deployments afectan las funciones e interrumpen la productividad. Se debe contar con un personal experimentado que garantice la seguridad y continuidad del Datacenter.

Nube: Minimiza las obligaciones por parte del equipo de IT, ya que el proveedor debe administrar, realizar los mantenimientos y actualizaciones de la infraestructura. De esta forma IT puede dar prioridad a planificar y capacitarse en las nuevas herramientas y tecnologías ofrecidas de la nube aprovechando los beneficios que brinda.

Implementación de instancia EC2

Se realiza la creación de instancia en Amazon Web Services con la interfaz de *Amazon Linux* y arquitectura de 64 bits, modo de arranque *uefi-preferred*, ID de AMI *ami-06c68f701d8090592* y una instancias de *t2.micro*.

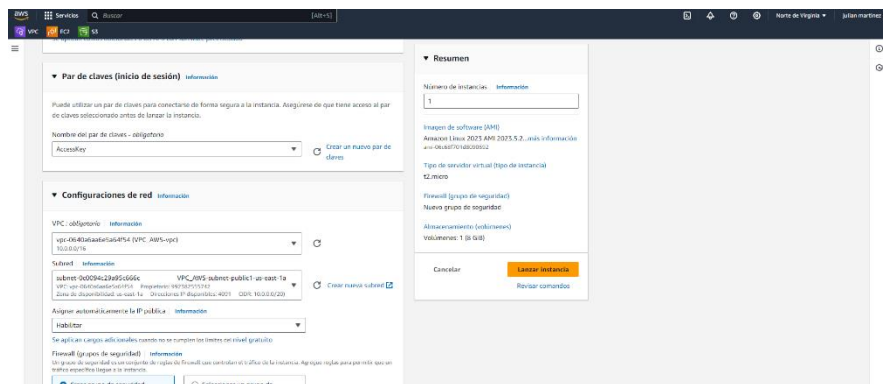
Ilustración 1 Tipo de instancia



Tomado de: Creación propia en Amazon Web Services

El nombre del par de claves AccesKey, configuración de red VPC_AWS-vpc, IP pública, grupo de seguridad SG_ServerCol1, reglas de seguridad de entrada SSH con acceso de cualquier lugar, volumen de 8GiB gp3.

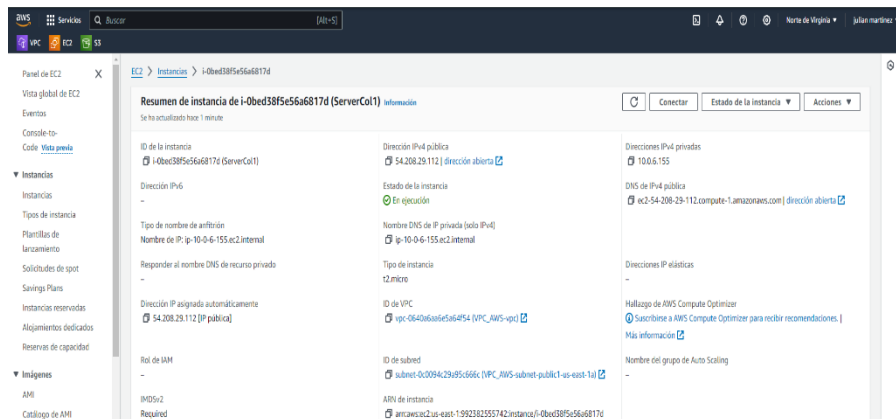
Ilustración 2 Configuración de red



Tomado de: Creación propia en Amazon Web Services

Se verifica que la configuración de la instancia creada sea exitosa, dentro de esta se puede validar el resumen con los datos de ID, dirección IPs DNS entre otras características de la red.

Ilustración 3 Resumen de la instancia

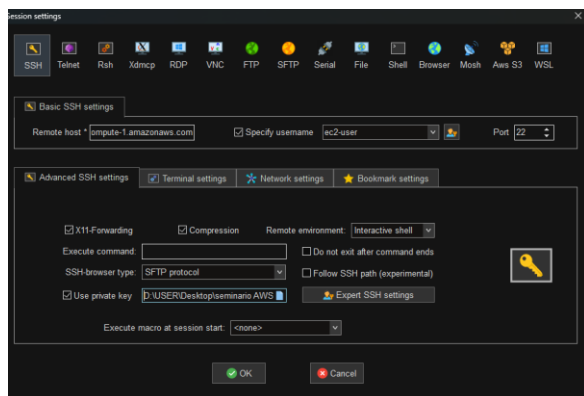


Tomado de: Creación propia en Amazon Web Services

Ingreso por MobaXterm

Al crear la instancia, se dirige a la aplicación previamente descargada de MobaXterm y realizar conexión de sesión por **SSH** en la parte superior izquierda.

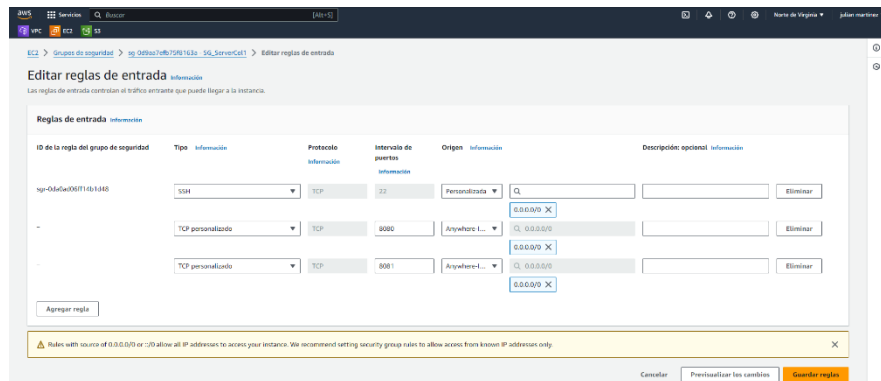
Ilustración 4 Conexión por MobaXterm



Tomado de: Creación propia en MobaXterm

La configuración **Basic SSH Settings**, se encuentra en **Conectar > Cliente SSH > Ejemplo: ssh -i "AccessKey.pem" [ec2-user@ec2-54-208-29-112.compute-1.amazonaws.com](https://ec2-54-208-29-112.compute-1.amazonaws.com)**, especificar el usuario **ec2-user** y la llave privada descargada del **VPC** creado.

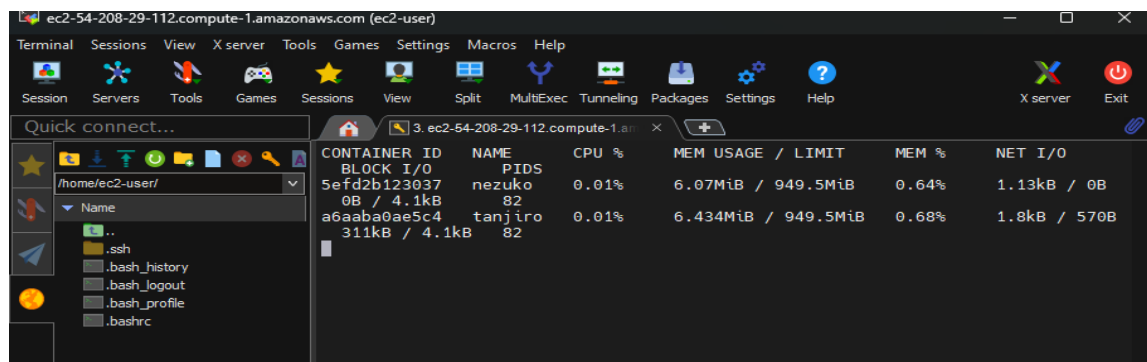
Ilustración 9 Configuración de puertos en EC2



Tomado de: Creación propia en Amazon Web Services

Se usa el comando `docker stats` para saber las estadísticas de los contenedores.

Ilustración 10 Estadísticas de contenedores

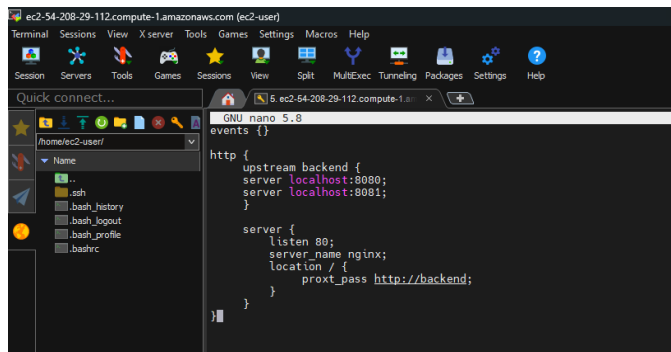


Tomado de: Creación propia en MobaXterm

Descarga e Instalación de la Aplicación

Se realiza la descarga de la página de prueba de la aplicación con la siguiente ruta con el comando `wget https://html5up.net/massively/download`, se desbloquea con `unzip download` y se verifican los estatus del contenedor después de agregarle la aplicación.

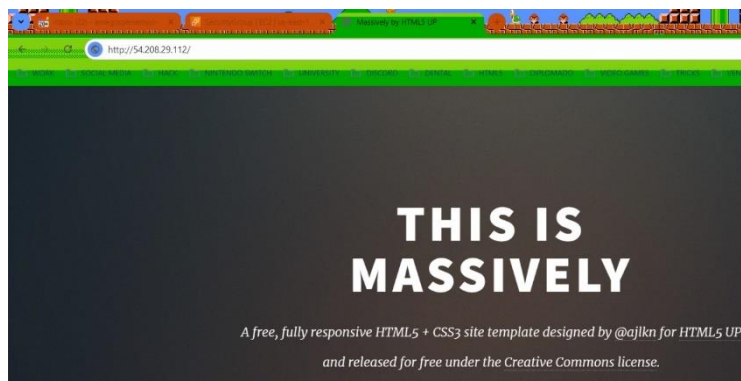
Ilustración 13 Configuración del balanceador de carga



Tomado de: Creación propia en MobaXterm

Se realiza la verificación por medio de la URL pública <http://54.208.29.112>.

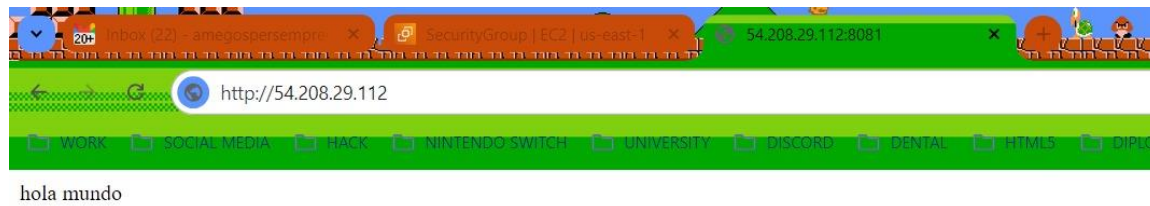
Ilustración 14 Prueba de puerto 8080



Tomado de: Creación propia en Localhost

Se actualiza de nuevo la URL para verificar que el balanceador de carga funcione con el siguiente puerto.

Ilustración 15 Prueba de puerto 8081

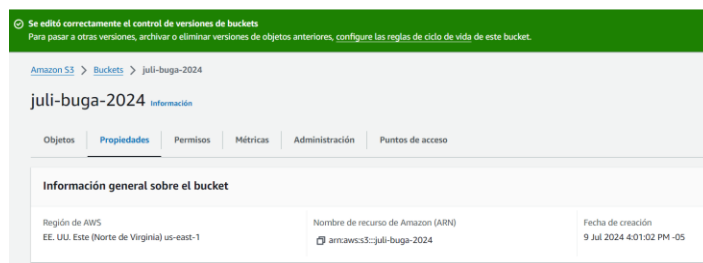


Tomado de: *Creación propia en Localhost*

Implementación de S3

Para la implementación del **S3** con el sitio estático usado previamente en **EC2** se realiza la creación del **Bucket** con el nombre único de **juli-buga-2024** para uso general, se activa el control de versiones del bucket y se realiza la carga de prueba de un archivo cualquiera.

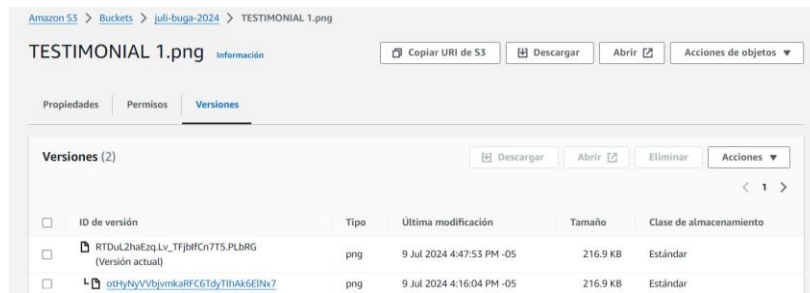
Ilustración 16 Creación de Bucket



Tomado de: *Creación propia en Amazon Web Services*

Una vez el **objeto** es creado, se puede apreciar la URL del objeto **https://juli-buga-2024.s3.amazonaws.com/TESTIMONIAL+1.png** y se verifican las versiones subiendo el mismo archivo.

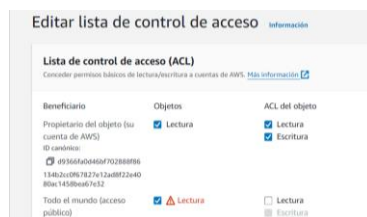
Ilustración 17 Carga y prueba de objeto



Tomado de: *Creación propia en Amazon Web Services*

Se habilita el **ACL** para activar los permisos de solo lectura.

Ilustración 18 Habilitación de permisos ACL



Tomado de: *Creación propia en Amazon Web Services*

Alojamiento de Sitios Web Estáticos

Se realiza la edición de alojamiento de sitios web estáticos para incorporar la aplicación usada en el **EC2** y habilitando como tipo de **Alojar un sitio web estático** para que los usuarios puedan acceder al contenido desde un sitio web.

Ilustración 19 Configuración de sitios web estáticos

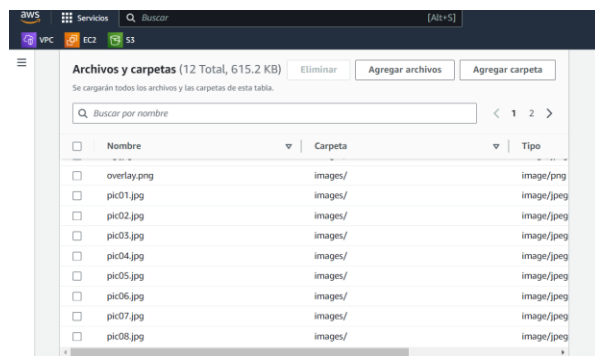


Tomado de: Creación propia en Amazon Web Services

Carga de Imágenes e Index

Es necesario descomprimir el archivo descargado de la aplicación usada con **EC2**. Dentro de la carpeta se encontrarán los archivos **index.html**, las carpetas de **images** y **assets**. Se realiza la carga del archivo **index**, la carga de las carpetas **images** y **assets**. Se habilita el acceso del **index** con permisos en **ACL** de solo lectura y se obtiene la URL del S3 terminado.

Ilustración 20 Carga de imágenes e index



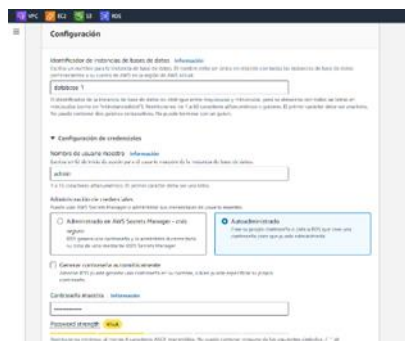
Tomado de: Creación propia en Amazon Web

Creación de Base de Datos

Se realiza la creación de la base de datos en Amazon Relational Databases (**RDS**). Se elige un método de creación de base de datos con **Creación estandar**, y opción de motor en **MySQL**. Se selecciona la versión del motor **MySQL 8.0.35** con una plantilla de **Capa gratuita**. Se crea un

identificador de instancias de bases de datos único **database-1**, nombre de usuario maestro **admin** con administración de credenciales de **Autoadministrado** y contraseña maestra **Coll234567890** (esta se encuentra oculta una vez se ingresa completa), al igual que su confirmación.

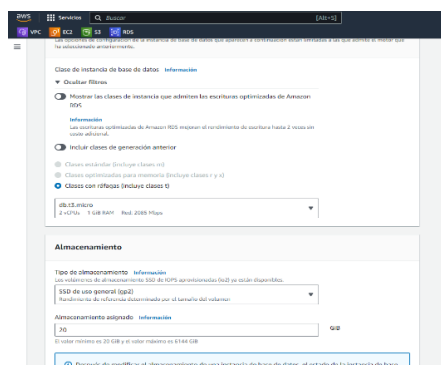
Ilustración 21 Creación de la base de datos



Tomado de: *Creación propia en Amazon Web Services*

La clase de la instancia de base de datos se usa con clases de ráfagas **db.t3.micro** con un tipo de almacenamiento de **SSD de uso general (gp2)** de **20GiB**, con recurso de computación **No se conecte a un recurso informático EC2**, tipo de red **IPv4**. Se indica una nube privada virtual (VPC) de **VPC_AWS-vpc (vpc-0640a6aa6e5a64f54)** y con un grupo de subredes de la base de datos **Crear un nuevo grupo de subredes de base de datos** con acceso público. El grupo de seguridad de VPC (firewall) es el grupo creado previamente **SG_ServerColl**, **Sin preferencia** de zona de disponibilidad y una entidad certificación **rds-ca-rsa2048-g1 (predeterminado)**.

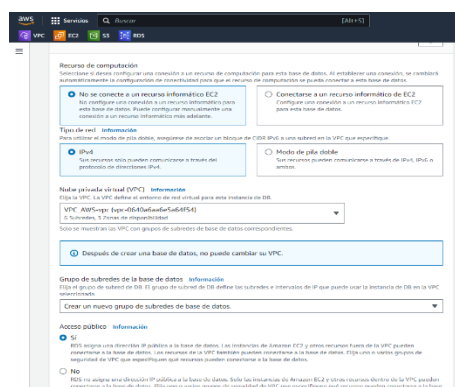
Ilustración 22 Configuración de base de datos



Tomado de: *Creación propia en Amazon Web Services*

La autenticación de la base de datos se usa de **Autenticación con contraseña** y la supervisión **Activar la monitorización mejorada** con grado de detalle de **60 segundos** y rol de supervisión **predeterminado**. La estimación de facturación total es de **14.71 USD** (12.41 USD por la instancia de base de datos y 2.30 USD de almacenamiento).

Ilustración 23 Recursos de computación

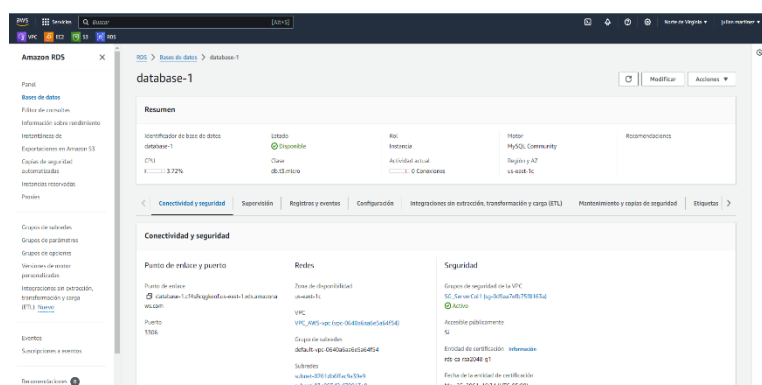


Tomado de: *Creación propia en Amazon Web Services*

Configuración de la Base de Datos

Una vez creada la base de datos, se verifica en el resumen de la página la **Conectividad y seguridad**.

Ilustración 24 Resumen de base de datos creada

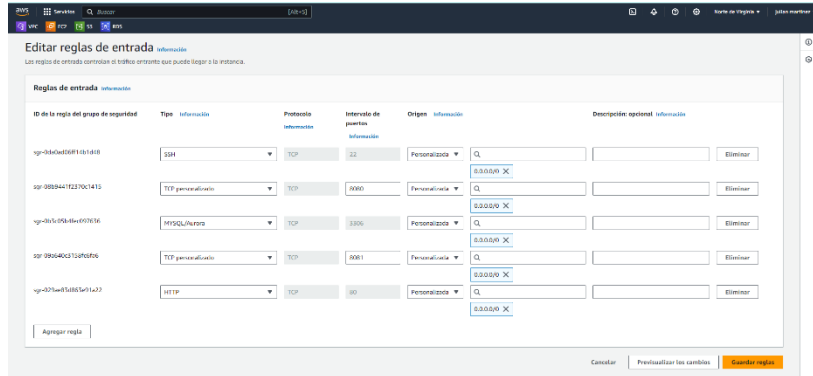


Tomado de: *Creación propia en Amazon Web Services*

El punto de enlace es el **database-1.cf4s8cqqkeof.us-east-1.rds.amazonaws.com** con puerto **3306** y la previa información brindada durante la creación de la base de datos. Se editan las reglas

de entrada de los puertos *SSH*, *TCP personalizado* con puerto *8080*, *MYSQL/Aurora*, *TCP personalizado* con puerto *8081*, y *HTTP*, todos con regla *0.0.0.0/0*.

Ilustración 25 Reglas de entrada de puertos

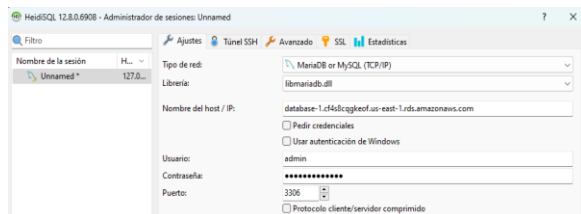


Tomado de: *Creación propia en Amazon Web Services*

Instalación de heidiSQL

Una vez se instala la aplicación de heidiSQL, se selecciona en *Nueva*, se colocan el puerto, enlace de usuario y contraseña.

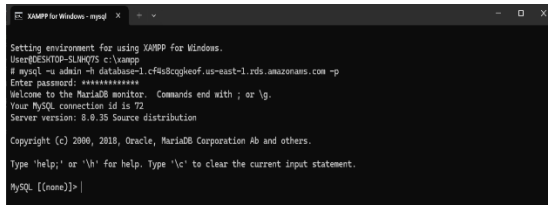
Ilustración 26 Administrador de sesiones



Tomado de: *Creación propia en heidiSQL*

Se ingresa a la base de datos por medio de la consola, en esta caso se utilizó *XAMPP*.

Ilustración 27 Acceso a la base de datos



```

Setting environment for using XAMPP for Windows.
USER@ESITOP-SLIM775 C:\xampp
# mysql -u admin -h database-1.cfk5bcgkeof.us-east-1.rds.amazonaws.com -p
Enter password: *****
Welcome to the MariaDB prompt: Commands end with ; or \g.
Your MySQL connection id is 92
Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

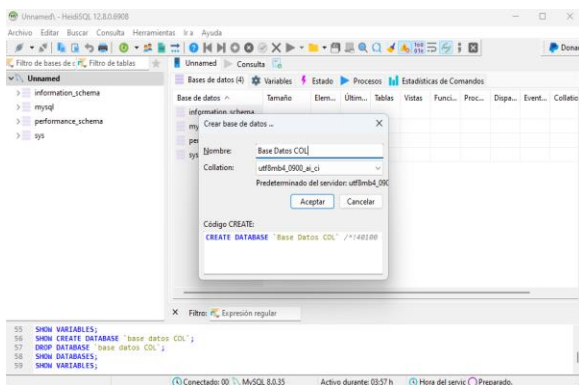
MySQL [(none)]>|

```

Tomado de: Creación propia en XAMPP 1

Se crea una base de datos dentro de heidiSQL con nombre **Base Datos COL** y collation de **utf8mb4_0900_ai_ci**.

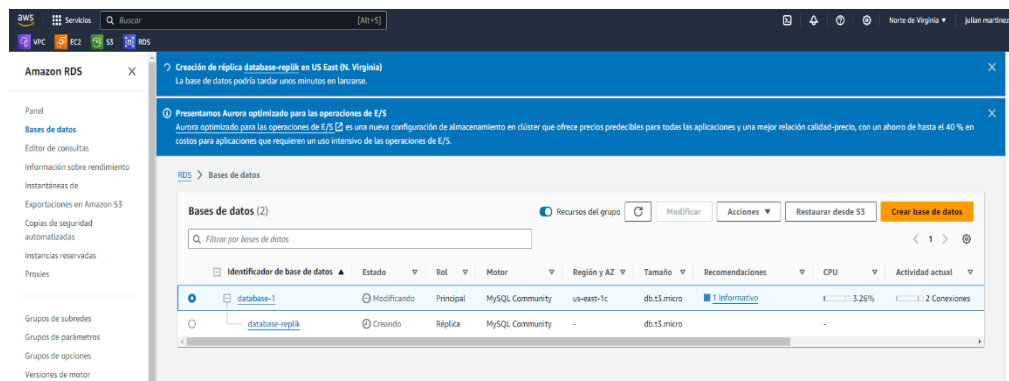
Ilustración 28 Creación de base de datos



Tomado de: Creación propia en heidiSQL

Se muestra por medio de la consola aplicando el comando **show databases** y creamos una replica de la base de datos en la plataforma con clic-derecho bajo el nombre de la base de datos con la misma instancia. Se realiza una copia de seguridad instantánea.

Ilustración 29 Replica de base de datos

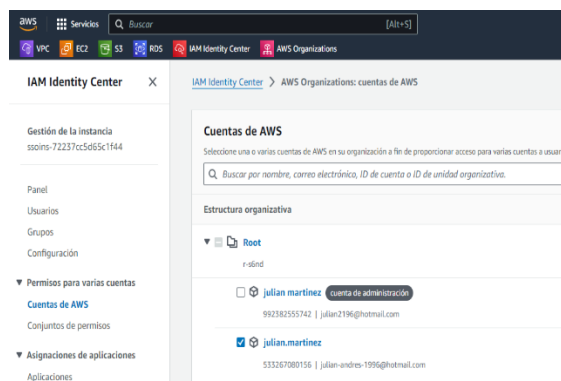


Tomado de: Creación propia en Amazon Web Services

Creación de Nuevo Usuario y Grupo para Permisos Pertinentes

Se crea una nueva cuenta de AWS con conjunto de permisos personalizados. Se seleccionan las políticas **AmazonEC2FullAccess** con **acceso total** de permisos con una duración de la sesión de **8 horas**. Una vez creada, se selecciona la cuenta y se asigna al grupo junto con los permisos de la cuenta con **accesototal**.

Ilustración 30 Creación de usuario



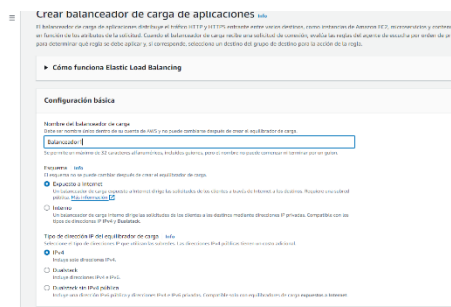
Tomado de: Creación propia en Amazon Web Services

Creación de Balanceadores de Carga en EC2

Se entra en el **EC2** y se dirige al apartado de balanceadores de carga. Se selecciona el **Balanceador de carga de aplicaciones** y se empieza con la configuración y la creación del

balanceador de carga con nombre **Balanceador1**, esquema **Expuesto a Internet, IPv4** para la dirección IP del equilibrador de carga.

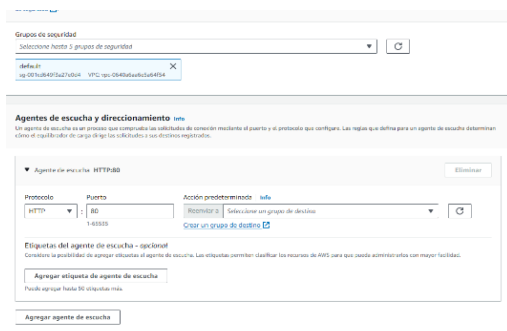
Ilustración 31 Creación de balanceadores de carga



Tomado de: *Creación propia en Amazon Web Services*

Una vez se tienen las instancias creadas, se crea el grupo de destino con grupo de seguridad **default** (predeterminado) y un protocolo de escucha para **HTTP** con puerto **80**.

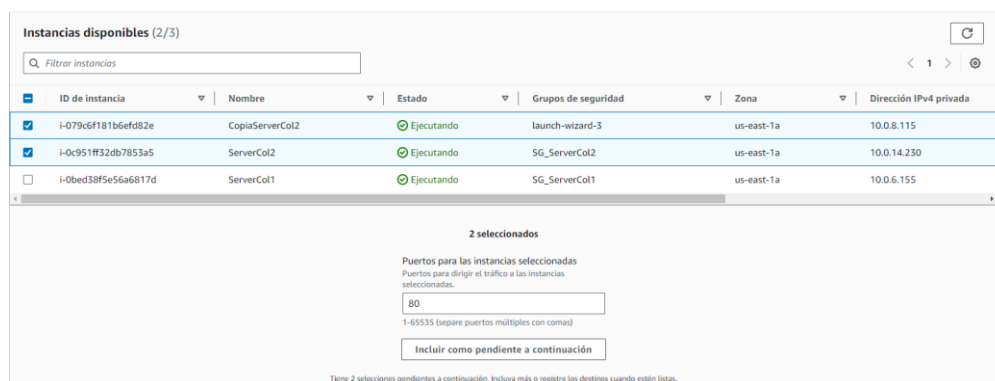
Ilustración 32 Ajuste de protocolos



Tomado de: *Creación propia en Amazon Web Services*

En la creación del grupo se elige el tipo de destino con una configuración de **Instancias**, se comprueba el puerto requerido con versión de protocolo **HTTP1**, Se escoge la instancia la cual se dirigirá el balanceador de carga y se seleccionan las dos.

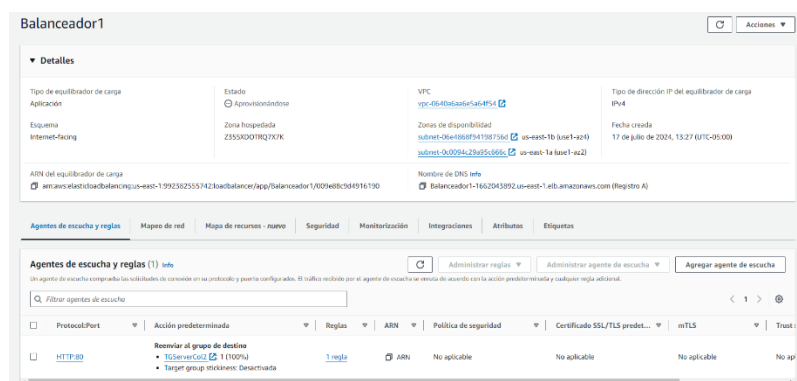
Ilustración 33 Instancias de grupo



Tomado de: *Creación propia en Amazon Web Services*

Se revisa la ruta de destino y los detalles del grupo de instancias para poder finalizar el balanceador de carga.

Ilustración 34 Ruta de destino



Tomado de: *Creación propia en Amazon Web Services*

Creación de Copia de Instancia

Se crea una copia de la instancia desde el mismo lugar inicial de la instancia creada para usar las propias AMI *De mi propiedad* con nombre *CopiaServerCol2*. Tipo de instancia de *t2.micro* que cuente con un nuevo grupo de seguridad denominado *“launch-wizard-1”* con las reglas *Permitir el tráfico de SSH desde Cualquier lugar* y almacenamiento de *8GiB gp3* de volumen de raíz (sin cifrar).

Ilustración 35 Tipo de instancia

▼ Tipo de instancia Información | Obtener asesoramiento

Tipo de instancia

t2.micro Apto para la capa gratuita Todas las generaciones

Familia: t2 1 vCPU 1 GiB Memoria Generación actual: true

Bajo demanda Windows base precios: 0.0162 USD por hora

Bajo demanda SUSE base precios: 0.0116 USD por hora

Bajo demanda RHEL base precios: 0.026 USD por hora

Bajo demanda Linux base precios: 0.0116 USD por hora

Se aplican costos adicionales a las AMI con software preinstalado

Comparar tipos de instancias

▼ Par de claves (inicio de sesión) Información

Puede utilizar un par de claves para conectarse de forma segura a la instancia. Asegúrese de que tiene acceso al par de claves seleccionado antes de lanzar la instancia.

Nombre del par de claves - obligatorio

AccessKey Crear un nuevo par de claves

Para las instancias de Windows, utilice un par de claves para descifrar la contraseña del administrador y, a continuación, utilice la contraseña descifrada para conectarse a la instancia.

Tomado de: Creación propia en Amazon Web Services

La configuración de red de VPC `vpc-0640a6aa6e5a64f54` (VPC_AWS-vpc) con IP pública asignada.

Ilustración 36 Configuración de red

▼ Configuraciones de red Información

VPC: obligatorio Información

vpc-0640a6aa6e5a64f54 (VPC_AWS-vpc) 10.0.0.0/16

Subred Información

subnet-0a0004c29295c666c VPC_AWS-subnet-public-1-us-east-1a Crear nueva subred

VPC: vpc-0640a6aa6e5a64f54 Proprietario: 992782155742

Zona de disponibilidad: us-east-1a Direcciones IP disponibles: 4096 CIDR: 10.0.0.0/20

Asignar automáticamente la IP pública Información

Habilitar

Se aplican cargos adicionales cuando no se cumplen los límites del nivel gratuito

Firewall (grupos de seguridad) Información

Un grupo de seguridad es un conjunto de reglas de firewall que controlan el tráfico de la instancia. Agregue reglas para permitir que un tráfico específico llegue a la instancia.

Crear grupo de seguridad Seleccionar un grupo de seguridad existente

Nombre del grupo de seguridad - obligatorio

launch-wizard-3

Descripción - obligatorio Información

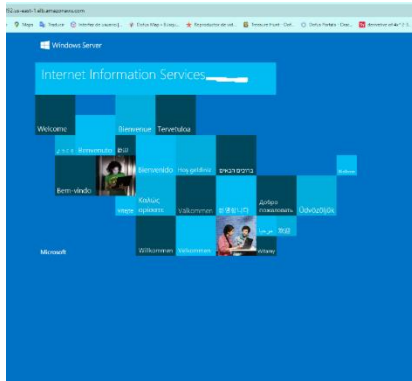
launch-wizard-3 created 2024-07-17T18:11:26.977Z

Reglas de grupos de seguridad de entrada

Tomado de: Creación propia en Amazon Web Services

Una vez terminada la creación del balanceador de carga verificamos el DNS que se otorga para acceder a las páginas de la instancia que muestre dos pantallas.

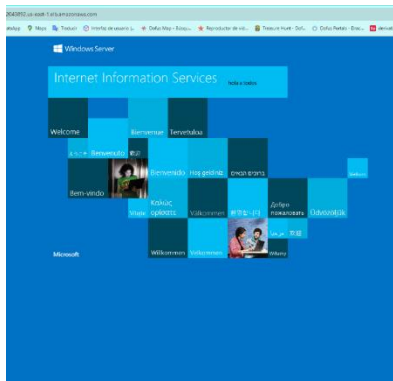
Ilustración 37 DNS de balanceador de carga 1



Tomado de: Creación propia en Amazon Web Services

Luego se verifica el siguiente.

Ilustración 38 DNS de balanceador de carga 2



Tomado de: Creación propia en Amazon Web Services

EC2 Auto Scaling

Se crea un grupo de auto escalado y se elige una pantalla de lanzamiento. Se crea una configuración de lanzamiento.

Desarrollar una arquitectura utilizando una herramienta como Amazon Web Services (AWS) para implementar

TechSolutions S.A. es una empresa que está creciendo y requiere de necesidades flexibles junto a un equipo de desarrollo diligente, por eso recomendamos migrar a la nube. Obtendrán flexibilidad, escalabilidad, y reducir costos a futuro que se acoplan con los objetivos deseados. A pesar de los riesgos que se presenta en cuanto a ciberseguridad y acceso de información de terceros, se puede filtrar y contratar un proveedor que cumpla con todas las certificaciones, medidas de seguridad y acuerdos de nivel de servicio.

Esta recomendación se adapta a la necesidad actual de la empresa, pero esta deberá realizar un seguimiento y reevaluación periódica que garantice que la infraestructura de IT se alinee con las tendencias del mercado.

Al tener una instancia en EC2 se da un poco más de manejo a la creación de contenedores y balanceadores de carga ya que al descargar los archivos, se acomoda directamente a comparación de una instancia de S3, puesto que maneja carpetas y subida de archivos; teniendo por defecto el subir de los archivos en solo lectura omitiendo imágenes y dejando una página con solo texto a comparación de la EC2.

Agradecimientos

En primer lugar, queremos agradecer a Dios, por su guía y fortaleza nos has, acompañado en cada paso de este camino. Gracias a tu inspiración y apoyo, este logro no habría sido posible.

A nuestras familias, nuestro más profundo agradecimiento por su incondicional amor, y apoyo. Gracias por estar siempre a nuestro lado, brindándonos palabras de aliento y motivación. Su paciencia y sacrificio han sido fundamentales para alcanzar esta meta.

Referencias Bibliográficas

Normas ISO. (n.d.). ISO 27001. Disponible en <https://normaiso27001.es/>

Requisitos del sistema de gestión de servicios.

SISTEMA DE ALIMENTACIÓN ININTERRUMPIDA (SAI/UPS):

IGSA. (n.d.). ¿Qué es y para qué sirve un UPS? Recuperado de <https://www.igsa.com.mx/articulos-de-blog/articulos-de-computo-y-telecomunicaciones/que-es-y-para-que-sirve-un-ups/>

NQA. (n.d.). ISO 22301: Continuidad del negocio. Recuperado de <https://www.nqa.com/es-co/certification/standards/iso-22301#:~:text=La%20ISO%2022301%20tiene%20como,de%20posibles%20amenazas%20e%20interrupciones>

Liberty Seguros. (n.d.). Seguros de protección para la empresa. Recuperado de <https://www.libertycolombia.com.co/empresas/seguros-proteccion-para-la-empresa>

Normas ISO. (n.d.). ISO 20000. Recuperado de <https://www.normas-iso.com/iso-20000/>

Departamento Administrativo de la Función Pública. (2009). Ley 1273 de 2009. Recuperado de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=34492>

Departamento Administrativo de la Función Pública. (2012). Ley 1581 de 2012. Recuperado de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=49981>

GeniusX. (n.d.). Arquitecturas modulares en la nube. Recuperado de <https://geniusx.eu/arq-software-modular-en-la-nube/#:~:text=Las%20arquitecturas%20de%20software%20modulares%20en%20la%20nube%20son%20un,de%20interfaces%20claras%20y%20est%3%A1ndar.>

DataCenter Dynamics. (n.d.). El impacto del diseño modular en data centers corporativos: Entrevista con una experta. Recuperado de <https://www.datacenterdynamics.com/es/opinion/el-impacto-del-diseno-modular-en-data-centers-corporativos-entrevista-con-una-experta/#:~:text=El%20dise%C3%B1o%20modular%20potencia%20la,funci%C3%B3n%20de%20sus%20necesidades%20cambiantes.>

Amazon Web Services. (n.d.). AWS pago por uso. Recuperado de <https://aws.amazon.com/es/aws-cost-management/>

HTML5 UP. (n.d.). Massively [Archivo PDF]. Recuperado el 9 de julio de 2024, de <https://html5up.net/massively/download>

Docker. (n.d.). Official Apache HTTP Server Docker Image. Recuperado el 9 de julio de 2024, de https://hub.docker.com/_/httpd

Amazon Web Services. (n.d.). AWS Management Console. Recuperado el 17 de julio de 2024, de <https://aws.amazon.com/es/console/>

Berrio López, J. P. (2024). Seminario sobre Amazon Web Services. Seminario presentado en la Universidad Uniremington, Medellín.

Enlace de videos

Segunda entrega:

<https://drive.google.com/file/d/1Pp8ZAF-QGut2pGWyDp6nk11aeGKyeWZs/view?usp=sharing>

Tercera entrega:

<https://drive.google.com/file/d/1DEBB6yMWt485K6YI94m4vkZ-kn85pxWm/view?usp=sharing>