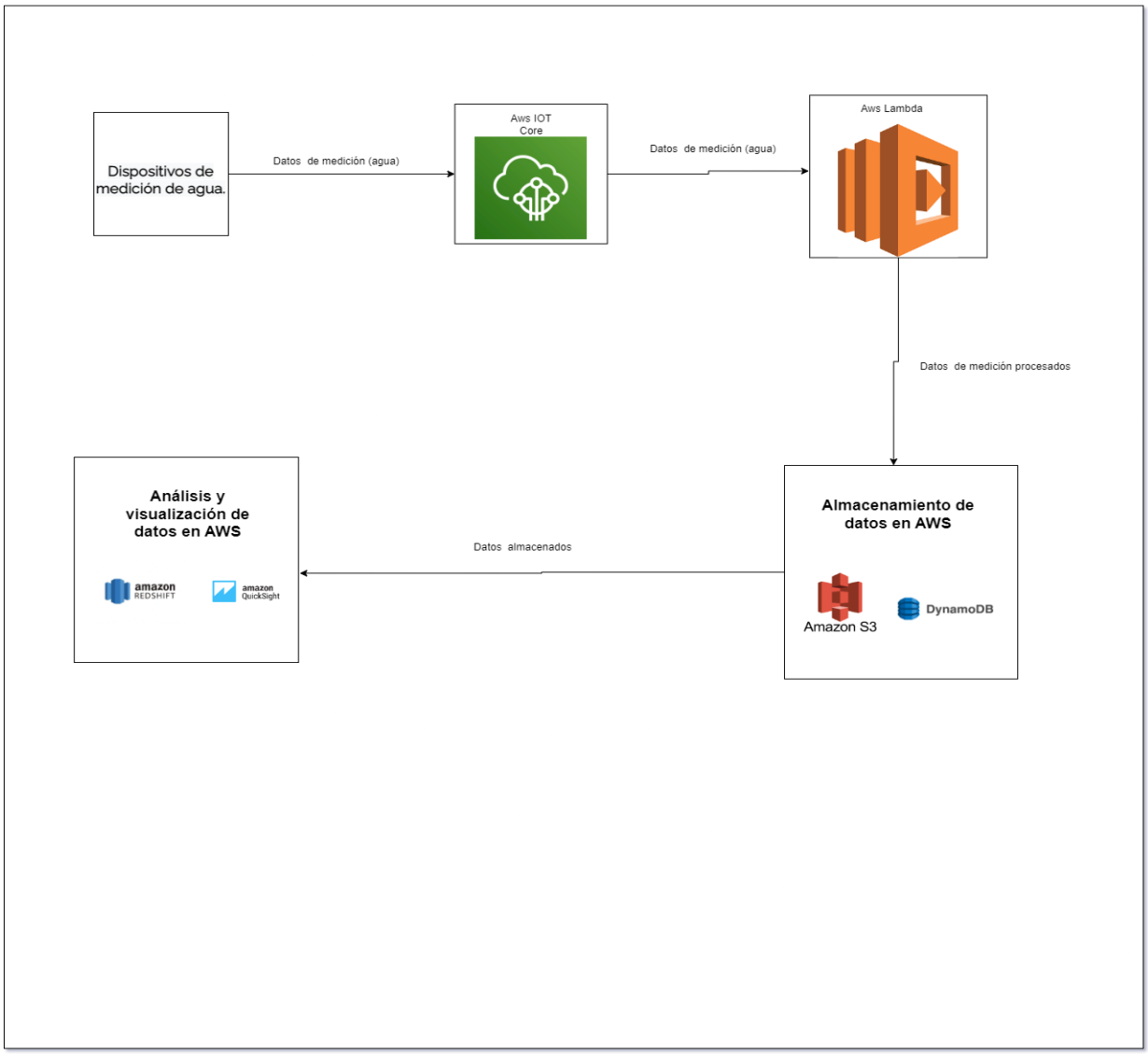
SOLUCIÓN PRUEBA TÉCNICA - APRENDIZ ARQUITECTO DE SOLUCIONES PARTE I

1R/.

Arquitectura propuesta para la solución



Descripción.

En este diagrama, los dispositivos de medición (por ejemplo, medidores de agua) están conectados a AWS IoT Core, que es un servicio de administración de dispositivos de IoT en AWS. AWS IoT Core recibe los datos de medición de los dispositivos y los transmite a través de la nube de AWS.

Luego, los datos de medición se procesan utilizando AWS Lambda, que permite realizar operaciones de procesamiento adicional, como validación de datos o cálculos específicos.

Después del procesamiento, los datos se almacenan en servicios de almacenamiento de AWS, como Amazon S3 (Simple Storage Service) o Amazon DynamoDB, donde se pueden escalar y mantener de manera duradera.

Por último, los datos almacenados se pueden analizar y visualizar utilizando servicios de análisis de datos de AWS, como Amazon Redshift, Amazon QuickSight u otros servicios similares. Esto permite realizar análisis en profundidad y visualizar los resultados para obtener información valiosa.

Cabe destacar que este es solo un ejemplo de arquitectura y se puede adaptar según las necesidades y requisitos específicos de la empresa de servicios públicos.

El flujo de datos en la arquitectura propuesta es el siguiente:

1. Origen de los datos: Los datos de medición de consumo de agua y electricidad se generan en los dispositivos de medición instalados en las ubicaciones geográficas de los clientes de la empresa de servicios públicos.

2. Dispositivos de medición: Los dispositivos de medición capturan los datos de consumo de agua y electricidad y los envían a través de la red de comunicación establecida hacia AWS IoT Core.

3. AWS IoT Core: Este servicio de administración de dispositivos de IoT en AWS actúa como el punto de entrada para los datos. Recibe los datos de medición provenientes de los dispositivos y los procesa para su posterior distribución.

4. AWS Lambda: Los datos de medición entrantes en AWS IoT Core pueden ser procesados utilizando funciones de AWS Lambda. Estas funciones pueden realizar diversas tareas, como validar los datos, enriquecerlos con información adicional o realizar cálculos adicionales.

5. Almacenamiento de datos: Después de ser procesados, los datos de medición son

almacenados en servicios de almacenamiento de AWS, como Amazon S3 o Amazon DynamoDB. Estos servicios ofrecen una alta escalabilidad y durabilidad para almacenar grandes volúmenes de datos.

6. Análisis y visualización de datos: Los datos almacenados pueden ser accedidos y analizados utilizando servicios de análisis de datos de AWS, como Amazon Redshift, Amazon Athena o Amazon QuickSight. Estas herramientas permiten realizar consultas, aplicar algoritmos analíticos y generar visualizaciones interactivas para obtener información valiosa.

En resumen, el flujo de datos comienza en los dispositivos de medición, pasa por AWS IoT Core y AWS Lambda para su procesamiento, luego se almacena en servicios de almacenamiento de AWS, y finalmente se utiliza en servicios de análisis y visualización para obtener información y conocimientos relevantes.

Para garantizar la seguridad y privacidad de los datos en la solución propuesta, se deben implementar las siguientes medidas:

1. Autenticación y autorización: Establecer un sistema de autenticación sólido para garantizar que solo los dispositivos y usuarios autorizados puedan acceder y enviar datos a la solución. Se pueden utilizar mecanismos como claves de acceso, certificados digitales o tokens de seguridad. Además, es importante definir los roles y permisos adecuados para garantizar que solo las personas autorizadas tengan acceso a los datos.

2. Cifrado de datos: Aplicar el cifrado de extremo a extremo para proteger los datos mientras se transmiten desde los dispositivos a través de la red de comunicación hasta la nube de AWS. Esto se puede lograr utilizando protocolos seguros de comunicación, como HTTPS, y cifrado de datos utilizando algoritmos robustos.

3. Protección de datos en reposo: Almacenar los datos de manera segura en la nube de AWS utilizando servicios de almacenamiento como Amazon S3 o Amazon DynamoDB, los cuales ofrecen opciones de cifrado de datos en reposo. Se debe aplicar el cifrado de datos para evitar el acceso no autorizado a la información almacenada.

4. Seguimiento y auditoría: Establecer un sistema de seguimiento y auditoría para registrar y monitorear las actividades relacionadas con los datos. Esto permite identificar y rastrear cualquier intento de acceso no autorizado o uso indebido de los datos. Los servicios de AWS, como Amazon CloudWatch y AWS CloudTrail, pueden ser utilizados para supervisar y registrar eventos relevantes.
5. Formación y concienciación: Capacitar a los empleados y usuarios involucrados en la solución sobre las mejores prácticas de seguridad y privacidad de datos. Esto incluye la importancia de mantener contraseñas seguras, evitar el intercambio de información confidencial y comprender las políticas y procedimientos de seguridad establecidos.

6. Evaluación de riesgos y mitigación: Realizar evaluaciones periódicas de riesgos de seguridad y privacidad para identificar posibles vulnerabilidades y amenazas. Implementar medidas de mitigación apropiadas, como parches de seguridad, actualizaciones de software y controles adicionales, para reducir los riesgos a un nivel aceptable.

Es importante destacar que la implementación de estas medidas debe ser adaptada a las necesidades y requisitos específicos de la empresa de servicios públicos, y se recomienda contar con la asesoría de expertos en seguridad de la información para garantizar una protección adecuada de los datos.

Algunas recomendaciones para mejorar la solución propuesta de ingesta de datos para una empresa de servicios públicos:

1. Redundancia y alta disponibilidad: Implementar mecanismos de redundancia y alta disponibilidad en la arquitectura para garantizar que la solución sea resistente a fallas. Esto puede incluir la replicación de datos en múltiples regiones de AWS y la configuración de autoscaling para garantizar la capacidad suficiente para manejar picos de carga.

2. Segregación de datos: Considerar la segregación de datos entre clientes y aplicar políticas de acceso y control de datos granulares. Esto ayudará a mantener la privacidad de los datos de cada cliente y reducir los riesgos de acceso no autorizado o intercambio de información.

3. Integración de análisis predictivo: Explorar la posibilidad de incorporar análisis predictivo a la solución. Esto permitiría identificar tendencias, patrones de consumo anómalos o pronosticar la demanda futura de agua y electricidad. Los servicios de AWS, como Amazon Machine Learning o Amazon SageMaker, pueden ser utilizados para implementar modelos de aprendizaje automático.

4. Personalización de la visualización de datos: Adaptar las herramientas de visualización de datos, como Amazon QuickSight, para que se ajusten a las necesidades y preferencias específicas de la empresa de servicios públicos. Esto puede incluir la creación de paneles personalizados, informes automatizados y la incorporación de métricas clave relevantes para la empresa.

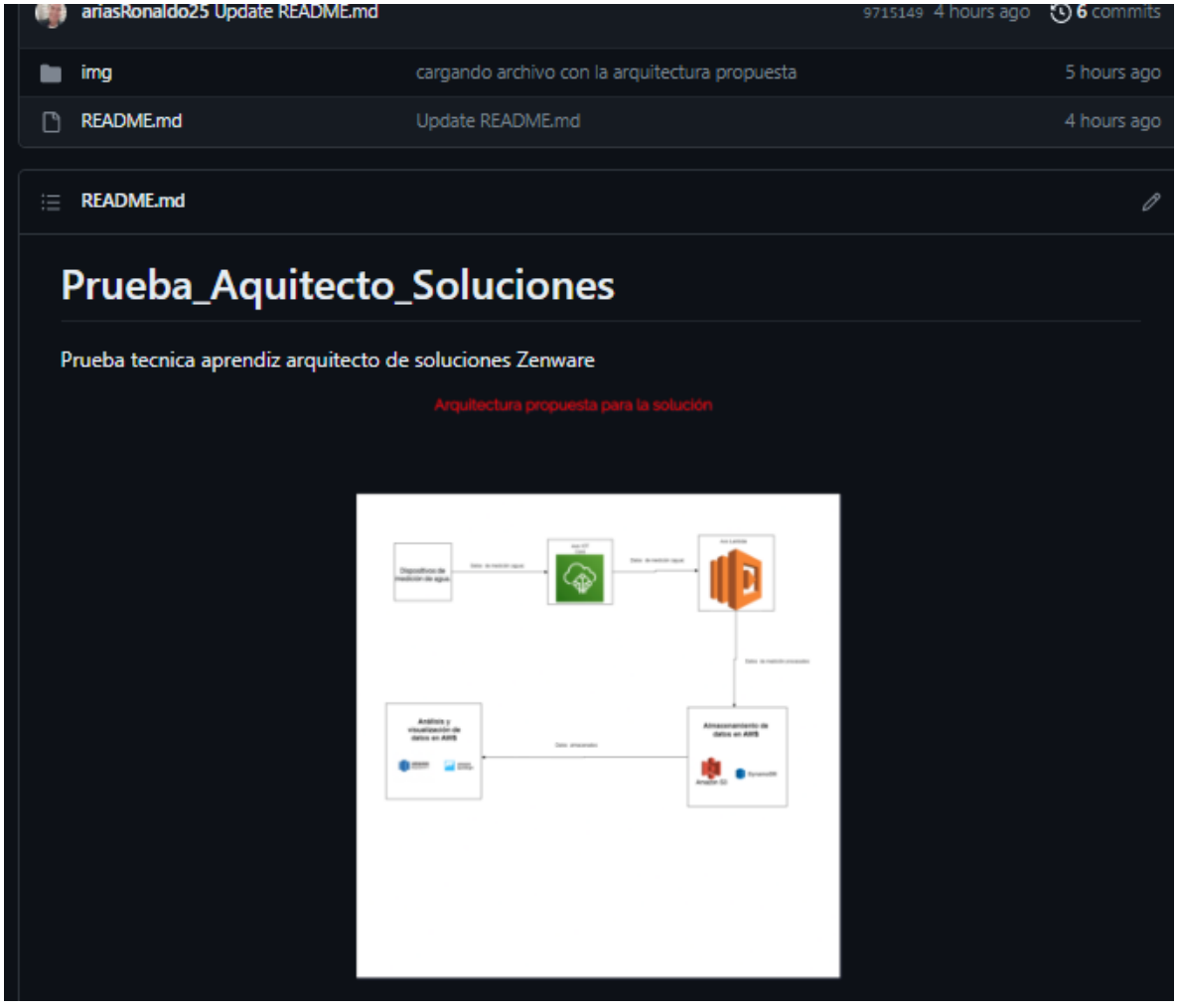
5. Integración con sistemas de facturación: Explorar la integración de la solución con los sistemas de facturación existentes de la empresa de servicios públicos. Esto permitiría automatizar la generación de facturas en función de los datos de consumo recopilados, lo que mejora la eficiencia operativa y la precisión en la facturación.

6. Análisis de datos en tiempo real: Evaluar la posibilidad de realizar análisis de datos en

tiempo real para detectar eventos anómalos o problemas en la infraestructura de suministro de agua y electricidad. Esto ayudaría a mejorar la capacidad de respuesta y permitiría tomar medidas correctivas de manera más rápida.

SOLUCIÓN PRUEBA TÉCNICA - APRENDIZ ARQUITECTO DE SOLUCIONES PARTE II

2 R/. **REPOSITORIO.**



LINK: https://github.com/ariasRonaldo25/Prueba_Aquitecto_Soluciones

Para garantizar la seguridad y privacidad de los datos en el script, se pueden implementar las siguientes medidas:

1. Acceso seguro a la base de datos on-premise: Asegúrate de establecer medidas de seguridad adecuadas en la base de datos on-premise, como autenticación fuerte, roles y permisos adecuados, y encriptación de datos en reposo. Esto ayudará a proteger los datos en su origen y evitar accesos no autorizados.

2. Almacenamiento seguro de credenciales: Evita almacenar credenciales (como contraseñas) directamente en el script. En su lugar, considera utilizar herramientas como variables de entorno, archivos de configuración externos o servicios de administración de secretos, como AWS Secrets Manager, para almacenar y recuperar las credenciales de forma segura.

3. Conexión segura a la base de datos on-premise: Utiliza mecanismos de conexión segura, como la conexión a través de SSL/TLS, para asegurar la comunicación entre el script y la base de datos on-premise. Esto ayuda a proteger los datos mientras se transmiten.

4. Manejo adecuado de errores y excepciones: Implementa una gestión adecuada de errores y excepciones en el script para evitar fugas de información sensible o mensajes de error que puedan exponer detalles del sistema o datos confidenciales.

5. Seguridad de la base de datos en la nube: Configura y utiliza medidas de seguridad en la base de datos en la nube AWS, como Amazon RDS. Esto puede incluir el cifrado de datos en reposo, el acceso basado en roles y políticas de seguridad, el monitoreo y la auditoría de actividades, y la implementación de actualizaciones de seguridad y parches.

6. Acceso y permisos adecuados en la nube: Asegúrate de que las credenciales utilizadas para acceder y almacenar datos en la base de datos en la nube AWS tengan los permisos mínimos necesarios. Limita el acceso solo a las cuentas y usuarios requeridos y sigue las mejores prácticas de seguridad recomendadas por AWS.

7. Monitoreo y registro de actividades: Implementa un sistema de monitoreo y registro de actividades en la nube AWS para detectar y responder rápidamente a cualquier actividad sospechosa o intento de acceso no autorizado. Utiliza servicios como AWS CloudTrail y Amazon CloudWatch para monitorear eventos y registros de la plataforma.

La implementación de estas medidas de seguridad debe adaptarse a las necesidades y requisitos específicos de la empresa de servicios públicos y cumplir con los estándares y regulaciones aplicables en la industria. Es recomendable consultar a expertos en seguridad de la información para obtener orientación y asesoramiento adicional.

Al migrar grandes volúmenes de datos con el script, pueden presentarse algunos problemas y desafíos relacionados con el rendimiento. Algunos de los posibles problemas y recomendaciones para mejorar el rendimiento del script durante la migración:

1. Tiempo de ejecución prolongado: Cuando se manejan grandes volúmenes de datos, el tiempo de ejecución del script puede aumentar significativamente. Esto puede afectar la eficiencia y la velocidad de la migración. Para mejorar el rendimiento, considera implementar técnicas de

procesamiento paralelo o distribuido. Esto puede involucrar dividir los datos en lotes más pequeños y ejecutar múltiples hilos o procesos en paralelo para procesarlos.

2. Uso de memoria y recursos del sistema: El manejo de grandes volúmenes de datos puede requerir una gran cantidad de memoria y recursos del sistema. Esto puede causar cuellos de botella y ralentizar la migración. Para optimizar el uso de memoria, considera procesar los datos en lotes más pequeños en lugar de cargar todos los datos en la memoria al mismo tiempo. Además, asegúrate de liberar recursos y cerrar conexiones de base de datos adecuadamente después de procesar cada lote de datos.

3. Optimización de consultas: Si la consulta de datos desde la base de datos on-premise es lenta, es posible que debas optimizar las consultas para mejorar el rendimiento. Asegúrate de tener índices adecuados en las tablas que se están consultando y utiliza cláusulas WHERE adecuadas para limitar el número de filas recuperadas. También puedes considerar la desnormalización de datos o la creación de vistas materializadas para acelerar la extracción de datos.

4. Paralelización de la carga de datos: Durante la carga de datos en la base de datos en la nube AWS, considera dividir la carga en múltiples procesos o hilos paralelos. Esto permitirá aprovechar mejor los recursos disponibles y acelerar la migración. Sin embargo, ten en cuenta los límites y la capacidad de rendimiento de la base de datos en la nube y ajusta el paralelismo en consecuencia.

5. Utilización de servicios de importación y exportación: Si la migración de datos implica grandes volúmenes de información, puedes considerar el uso de servicios de importación y exportación de AWS, como AWS Snowball o AWS Database Migration Service (DMS). Estos servicios permiten transferir grandes volúmenes de datos de manera eficiente utilizando dispositivos físicos y reducen el tiempo de migración.

6. Monitoreo y ajuste: Durante la migración de grandes volúmenes de datos, es importante monitorear el rendimiento del script y realizar ajustes según sea necesario. Utiliza herramientas de monitoreo y registros disponibles en AWS, como Amazon CloudWatch, para supervisar el uso de recursos, el tiempo de ejecución y detectar cuellos de botella. Basado en los resultados del monitoreo, realiza ajustes en la configuración, como el tamaño de lote, la asignación de recursos y la configuración de paralelismo, para mejorar el rendimiento.

En el archivo de especificaciones se detallan las transformaciones que se deben realizar en los datos:

- Se renombran las columnas existentes para que coincidan con los nombres esperados en la base de datos en AWS.
- Se realizan conversiones de unidades para ajustar los valores a las unidades requeridas en la base de datos en AWS.
- Se agrega una nueva columna calculada que representa el total de consumo de agua.
- Finalmente, se ordenan los datos por el ID del cliente en orden ascendente.