

Angelica Yeraldin Rodriguez Gualtero – 202215816

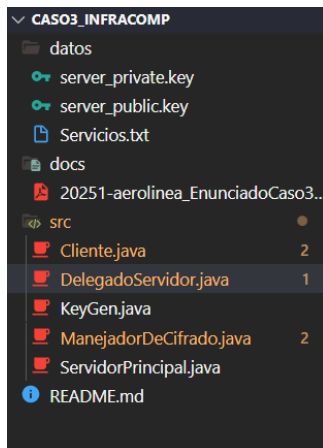
Joban Stib Mejia - 202213845

Sistema de rastreo de paquetes en una compañía transportadora

| | |
|---|---|
| Descripción de la organización del zip | 1 |
| Instrucciones para correr el servidor y cliente | 1 |
| Tabla con los datos recopilados | 2 |
| Comparación del tiempo en cifrado simétrico vs asimétrico | 3 |
| Gráficas | 3 |
| Tiempos para firmar en los escenarios | 3 |
| Tiempos para cifrar tabla en los escenarios | 4 |
| Tiempos para verificar consulta en los escenarios | 4 |
| Tiempos para caso simétrico vs caso asimétrico | 5 |
| Comentarios para las gráficas | 5 |

Descripción de la organización del zip

Para este tercer caso de infraestructura computacional, la organización de nuestro zip es la siguiente:



| Figura 1. Captura de la estructura del zip

Cómo se puede observar en la figura 1 la estructura del zip consiste en una carpeta docs que contiene los archivos solicitados, el excel con los datos recopilados y el informe con las gráficas y respuestas a las tareas planteadas.

Instrucciones para correr el servidor y cliente

1. Ejecutar el servidor:

- Abre una terminal y navega a la carpeta donde está el archivo `ServidorPrincipal.java`.
- Compila el servidor con el comando `javac ServidorPrincipal.java`.
- Ejecuta el servidor con `java ServidorPrincipal`. El servidor escuchará en el puerto 4000 y aceptará un máximo de 64 conexiones concurrentes.

2. Ejecutar el cliente:

- Abre otra terminal y navega a la carpeta donde está el archivo `Cliente.java`.
- Compila el cliente con el comando `javac Cliente.java`.
- Ejecuta el cliente con `java Cliente`. El cliente te permitirá seleccionar entre tres escenarios: una consulta, 32 consultas secuenciales o clientes concurrentes.
- Para probar con clientes concurrentes, en el menú del cliente, selecciona la opción "3" para ejecutar varios clientes simultáneamente. Y escribe el número de clientes

Tabla con los datos recopilados

Construya una tabla con los datos recopilados. Tenga en cuenta que necesitará correr cada escenario en más de una ocasión para validar los resultados.

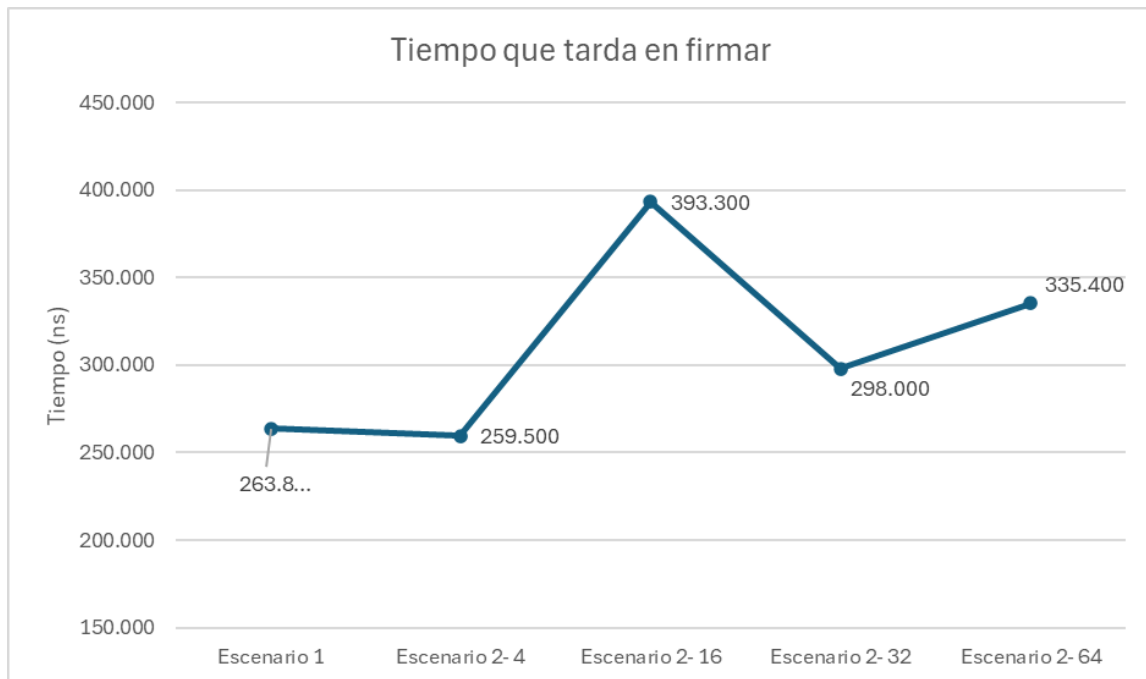
| Tiempo que tarda en firmar en los distintos escenarios | |
|---|---------|
| Escenario | Tiempo |
| Escenario 1 | 263.800 |
| Escenario 2- 4 | 259.500 |
| Escenario 2- 16 | 393.300 |
| Escenario 2- 32 | 298.000 |
| Escenario 2- 64 | 335.400 |
| Tiempo que tarda en verificar la consulta en los distintos escenarios | |
| Escenario 1 | 263.800 |
| Escenario 2- 4 | 3.500 |
| Escenario 2- 16 | 7.100 |
| Escenario 2- 32 | 5.300 |
| Escenario 2- 64 | 9.000 |
| Tiempo que tarda en cifrar la tabla en los distintos escenarios | |
| Escenario 1 | 263.800 |
| Escenario 2- 4 | 23.700 |
| Escenario 2- 16 | 37.400 |
| Escenario 2- 32 | 27.100 |
| Escenario 2- 64 | 26.800 |

Comparación del tiempo en cifrado simétrico vs asimétrico

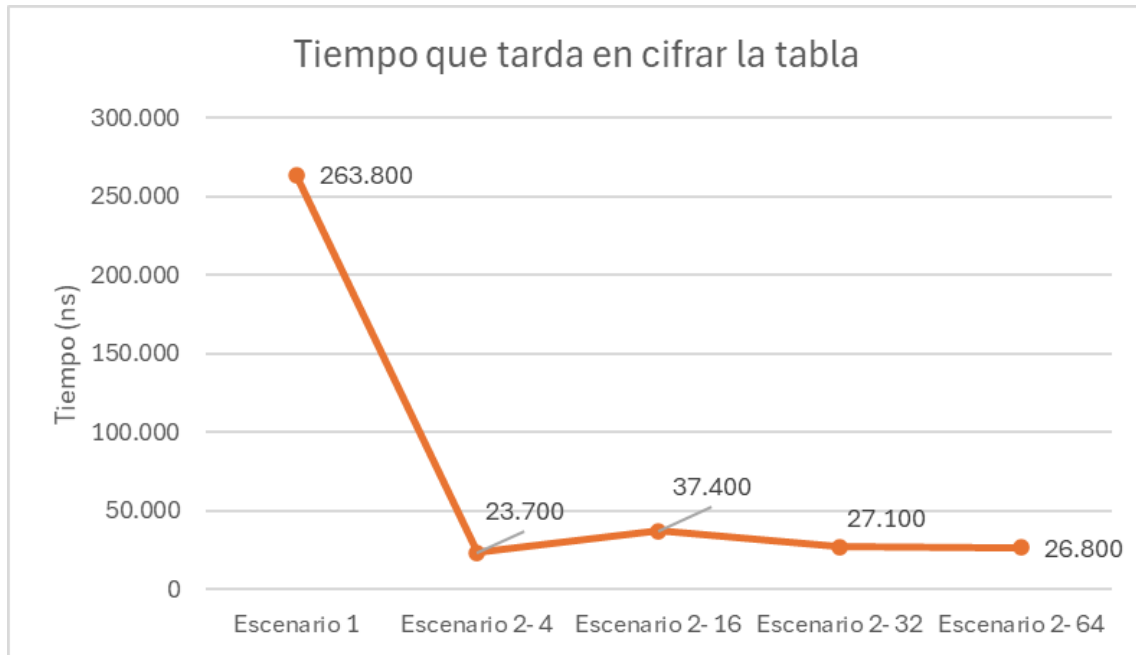
3. Compare el tiempo que el servidor requiere para cifrar la respuesta con cifrado simétrico y con cifrado asimétrico (con su llave pública). Observe que el cifrado asimétrico de la respuesta no se usa en el protocolo, solo se calculará para posteriormente comparar los tiempos.

Gráficas

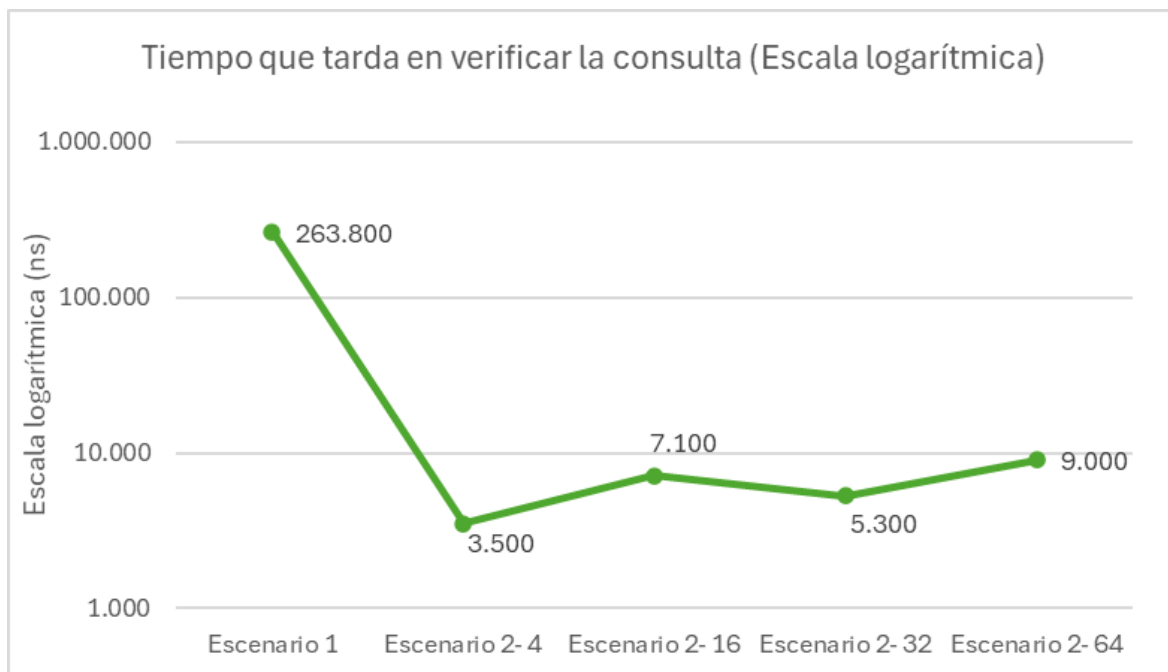
Tiempos para firmar en los escenarios



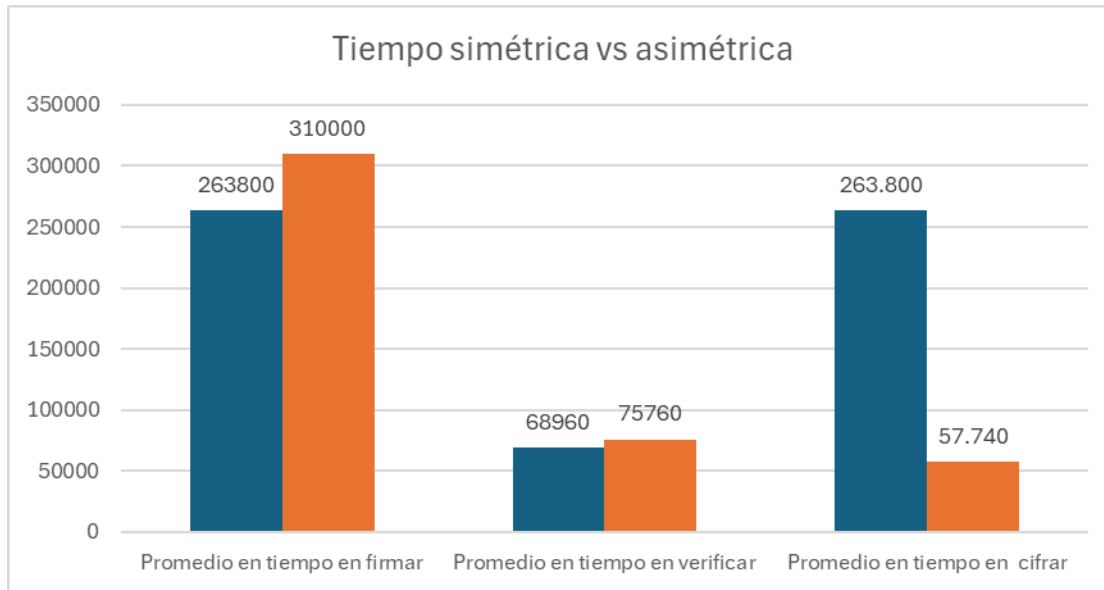
Tiempos para cifrar tabla en los escenarios



Tiempos para verificar consulta en los escenarios



Tiempos para caso simétrico vs caso asimétrico



Comentarios para las gráficas

Comentario sobre la gráfica de tiempos de firma:

En el primer escenario (cliente iterativo), el tiempo que tomó firmar fue de unos **263 ms**, y al usar 4 delegados concurrentes, el tiempo fue muy parecido (**259 ms**), lo que muestra que con pocos delegados no hay problema de rendimiento.

Cuando se sube a 16 delegados, el tiempo sube bastante hasta **393 ms**, lo que indica que el servidor tarda más en firmar porque tiene más trabajo al mismo tiempo. Curiosamente, con 32 y 64 delegados, el tiempo baja un poco (a **298 ms** y **335 ms**) en lugar de seguir subiendo. Esto puede pasar porque el servidor aprovecha mejor los recursos del computador cuando hay más carga.

En general, se puede decir que la firma funciona bien con pocos delegados, pero se empieza a notar la carga cuando hay muchos, especialmente alrededor de 16 delegados al tiempo.

Comentario sobre la gráfica de cifrado de la tabla:

En el escenario con un solo cliente (iterativo), el tiempo de cifrado fue de **263 ms**, mientras que en los escenarios con varios delegados simultáneos, el tiempo fue mucho menor, entre **23 ms y 37 ms**. Esto muestra que el algoritmo de cifrado simétrico (AES) funciona de forma muy eficiente cuando se usa con varios delegados a la vez.

El comportamiento sugiere que en los escenarios concurrentes el sistema logra aprovechar mejor los recursos del procesador, distribuyendo el trabajo de forma óptima. Además, los tiempos son bastante consistentes, incluso cuando se aumenta el número de delegados concurrentes de 4 a 64.

En general, se concluye que **el cifrado de la tabla es rápido y se mantiene estable incluso cuando hay alta concurrencia.**

Comentario sobre la gráfica de verificación de consulta:

El tiempo en el escenario con un solo cliente fue de **263 ms**, mientras que en los escenarios concurrentes fue mucho más bajo, variando entre **3.5 ms y 9 ms**. Este cambio muestra que el proceso de verificación se realiza de manera más rápida cuando hay varios delegados al mismo tiempo.

Esto puede deberse a que el servidor, al manejar múltiples solicitudes de forma paralela, aprovecha mejor la capacidad del sistema y reduce la espera por delegado. Aunque hay pequeñas variaciones en los tiempos, todos los valores están dentro de un rango corto, lo cual indica **una buena eficiencia y estabilidad del proceso de verificación en ambientes concurrentes.**

Comentario sobre la gráfica "Tiempo simétrica vs asimétrica":

La gráfica muestra que el **cifrado simétrico es mucho más rápido que el asimétrico**. En el caso de cifrado, el tiempo promedio con el método simétrico fue de **57 ms**, mientras que el asimétrico tomó **263 ms**, es decir, casi cinco veces más.

En la firma y verificación también se nota diferencia. Aunque firmar y verificar con RSA es más lento en promedio, la diferencia no es tan grande como en el cifrado. Esto confirma que, aunque **RSA es seguro, no es eficiente para manejar grandes volúmenes de datos.**