Juan José Osorio Cálad **202021720**

Alejandro Gómez Colorado **202015122**

Infraestructura Computacional

Caso 3: Parte 2

1. Descripción de la organización de los archivos en el ZIP.

El ZIP contiene tres carpetas principales, bin, docs y src. En bin se encuentran los archivos .class de las clases de java, en src están todos los archivos .java y en docs se encuentran los pdfs del caso, incluyendo este.

2. Instrucciones para correr el prototipo del cliente (Concurrencia)

Para correr el prototipo del cliente primero debe tener instalado Java, particularmente la versión 17 (JDK 17) en la cual se desarrolló este caso. Primero debe ejecutar el archivo ServidorMain.java y luego se debe ejecutar ClienteMain.java. Al ejecutar ClienteMain.java, debe ingresar 1 para 4 clientes, 2 para 16 y 3 para 32. Luego, el programa va a imprimir los resultados de fallo y éxito. No se imprimen los tiempos de generar el código de autenticación, verificación de la firma y Gy puesto que no decía en el enunciado.

3. Preguntas

3.1 Responda las siguientes preguntas:

a. En el protocolo descrito el cliente conoce la llave pública del servidor (K_w+). ¿Cuál es la manera común de enviar estas llaves para comunicaciones con servidores web?

Para comunicarse con un servidor web, comúnmente se genera la llave con el algoritmo RSA de cifrado asimétrico. El intercambio de llaves se realiza mediante un canal confiable, como TCP/IP o entregando la llave en persona (por ejemplo, darle una USB con la llave a la persona que va a descifrar el mensaje) en la mayoría de los casos. Sin embargo, como esta es una llave pública, hay que tener en cuenta que cualquier persona tiene acceso a esta; es por esto que enviar la llave pública por diferentes medios no importa, a pesar de que estos sean inseguros. Si fuera una llave de cifrado simétrico, sí sería necesario asegurarse de que el medio por el cual se comparte la llave es completamente seguro.

b. El protocolo Diffie-Hellman garantiza "Forward Secrecy", explique en qué consiste esta garantía.

Es una característica que garantiza que las claves de sesión no se verán comprometidas incluso si los secretos a largo plazo utilizados en el intercambio de claves de sesión son comprometidos. Forward Secrecy protege las sesiones pasadas contra futuros compromisos de claves o contraseñas. Al generar una clave única para cada sesión que inicia un usuario, el descubrimiento de una clave de sesión única

no afectará a ningún otro dato que no sea el intercambiado en la sesión específica protegida por esa clave en particular. Por lo tanto, las claves no se almacenan, se usan dentro de una sesión y se desechan.

3.2 Corra su programa en diferentes escenarios y mida los tiempos que el cliente demora para:

Los datos obtenidos en los distintos escenarios fueron tomados en un computador con un Intel core i7-10750H de 2.6 GHz.

c. Cifrar la consulta

A. Cifrar la Consulta					
		Tiempo promedio (ns) en la ejecución #:			
		1	2	3	Promedio Total
# Clientes	4	950.200,00	2.634.100,00	615.233,00	1.399.844,33
	16	839.320,00	882.572,00	168.190,00	630.027,33
	32	1.193.561,00	1.121.738,00	729.771,00	1.015.023,33

d. Genere el código de autenticación

B. Generación Código Autenticación					
		Tiempo promedio (ns) en la ejecución #:			
		1	2	3	Promedio Total
# Clientes	4	888.566,00	7.603.600,00	511.775,00	3.001.313,67
	16	378.300,00	247.220,00	293.700,00	306.406,67
	32	172.985,00	560.377,00	153.368,00	295.576,67

e. Verificación de la firma

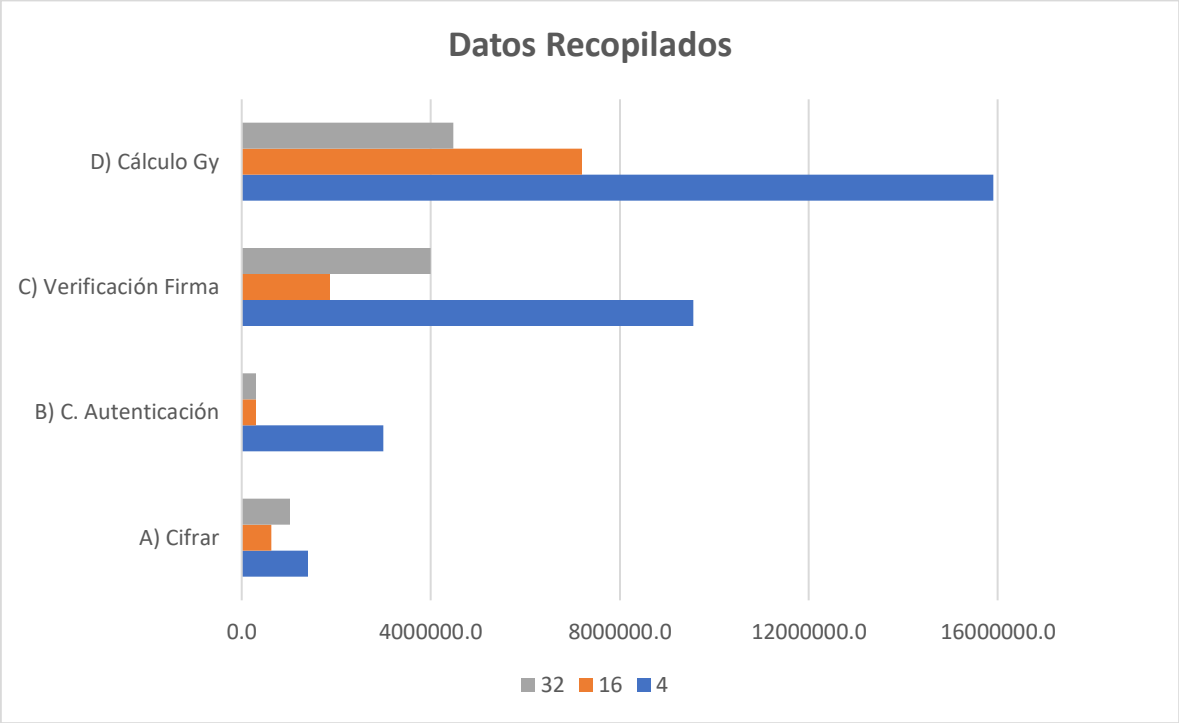
C. Verificación de la Firma					
		Tiempo promedio (ns) en la ejecución #:			
		1	2	3	Promedio Total
# Clientes	4	11.201.366,00	8.405.066,00	9.050.325,00	9.552.252,33
	16	1.095.630,00	3.553.650,00	957.000,00	1.868.760,00
	32	1.154.400,00	9.007.793,00	1.825.047,00	3.995.746,67

f. **Calcular Gy**

D. Cálculo Gy					
		Tiempo promedio (ns) en la ejecución #:			
		1	2	3	Promedio Total
# Clientes	4	18.223.300,00	14.020.233,00	15.471.800,00	15.905.111,00
	16	6.231.550,00	10.458.563,00	4.896.740,00	7.195.617,67
	32	4.041.795,00	5.651.123,00	3.723.309,00	4.472.075,67

3.3 Construya una tabla con los datos recopilados (tenga en cuenta que necesitará correr cada escenario en más de una ocasión para validar los resultados) y luego construya una gráfica con los datos de la tabla.

Datos Recopilados					
		Tiempo promedio (ns) en la ejecución #:			
		A) Cifrar	B) C. Autenticación	C) Verificación Firma	D) Cálculo Gy
# Clientes	4	1399844,3	3001313,7	9552252,3	15905111,0
	16	630027,3	306406,7	1868760,0	7195617,7
	32	1015023,3	295576,7	3995746,7	4472075,7



Gráfica de tiempo en nanosegundos para las 4 operaciones con 4 (azul), 16 (naranja) y 32 (gris) clientes.

3.4 Escriba sus comentarios sobre los resultados.

En general, el cálculo de Gy es el que toma más tiempo, puesto que es una operación computacionalmente costosa; se realiza una potencia y operación modulo con números de hasta 1024 bits. Por otro lado, la operación relativamente más sencilla es la de cifrado, puesto que se realiza la operación XOR por bloques mediante el algoritmo CBC, por ello sus tiempos tan bajos. En el caso de la verificación con la firma digital se utiliza el algoritmo asimétrico RSA, que es más demorado que los algoritmos simétricos que conocemos como CBC y EBC; es por esto que tiene un tiempo mediano en comparación a los demás.

Nótese que a medida que el número de clientes aumenta, los tiempos de Cálculo de Gy, generación del código de autenticación y cifrado, los tiempos en general disminuyen, puesto que el procesador juega entre los turnos que otorga a los clientes para llevar a cabo cada uno de los procedimientos. Cuantos más clientes hay, se aprovecha más la concurrencia y es menor el tiempo que un cliente ocupa el procesador para hacer cálculos que podría hacer en segundo plano.

3.5 Identifique la velocidad de su procesador, y estime cuántas consultas puede cifrar su máquina, cuántos códigos de autenticación puede calcular y cuántas verificaciones de firma, por segundo. Escriba todos sus cálculos.

Para las mediciones se tomó un aproximado de 16 consultas (clientes concurrentes) para estimar los tiempos de velocidad del procesador para procesar consultas por segundo. De esta manera, usamos un punto medio de concurrencia según lo mencionado anteriormente.

A) Cifrado:

$$630027.3\ ns = 0{,}00063\ s$$

$$\mathbf{0{,}00063\ s \rightarrow 16\ consultas\ de\ cifrado}$$

$$\mathbf{1s \rightarrow X\ consultas\ de\ cifrado}$$

$$\frac{1s}{0{,}00063\ s} = 1587{,}30s$$

$$X = 1587{,}30s * 16 = 25.396{,}82\ consultas\ de\ cifrado\ por\ segundo$$

B) Código de Autenticación:

$$306406{,}67\ ns = \ 0{,}00031\ s$$

$$\mathbf{0{,}00031s \rightarrow 16\ códigos\ de\ autenticación\ generados}$$

$$\mathbf{1s \rightarrow X\ consultas}$$

$$\frac{1s}{0{,}00031s} = \ 3.225{,}80s$$

$$X = 3.225{,}80s * 16 = 51612{,}90\ códigos\ de\ autenticación\ generados\ por\ segundo$$

C) Verificación Firma:

$$1868760\ ns = 0{,}00187\ s$$

$$\mathbf{0{,}00187\ s \rightarrow 16\ verificaciones\ de\ firma}$$

$$\mathbf{1s \rightarrow X\ verificaciones\ de\ firma}$$

$$\frac{1s}{0{,}00187} = \ 534{,}75s$$

$$X = 534{,}75s * 16 = 8556{,}14\ verificaciones\ de\ firma\ por\ segundo$$

Desempeño de mi procesador		
# Operaciones por segundo:		
A) Cifrar	B) C. Autenticación	C) Verificación Firma
25.396,82	51.612,90	8556,14