**Andrés Manrique Ardila 201713672**

**Juan Felipe Parra 201714259**

**{a.manrique, jf.parrac}@uniandes.edu.co**

**Infraestructura Computacional**

**Universidad de los Andes**

**Caso 2**

1. Identifique y describa los datos que deben ser protegidos en el sistema de rastreo de unidades de distribución. Explique su respuesta en cada caso y responda la pregunta ¿Si un actor no autorizado consigue acceso al dato mencionado, ya sea en modo lectura o escritura, ¿cómo podría afectar la empresa?

El sistema sobre el cual estamos trabajando es bastante importante puesto que se trata de una empresa transportadora y las consecuencias de intercepción de datos por parte de un tercero podría afectar seriamente varias funcionalidades. A continuación, vamos a exponer la gravedad que implica el ingreso de un tercero no autorizado a las diferentes funcionalidades del sistema.

* Recepción de órdenes de recogida: alterar los datos de esta funcionalidad no resultaría muy grave para el negocio pues implicaría perder el viaje hasta la ubicación del cliente. La lectura de estos datos tampoco resultaría tan grave puesto que al ser una empresa transportadora normal y no de valores no es tan grave que se sepa que x cliente quiere que le recoja un pedido.
* Gestión de rutas: esta es un elemento clave en las empresas transportadoras pues de esto depende su eficiencia a la hora de recoger y entregar paquetes. La lectura por parte de un usuario no autorizado no resultaría tan grave pues como consecuencias alguien podría atacar el carro transportador lo cual es poco probable en una empresa transportadora. Escribir sobre estos datos es bastante grave, podría llegar a hacer perder mucho tiempo de trabajo lo cual representa un gran daño y costo para la empresa.
* Rastreo de unidades de distribución y paquetes: al igual que en gestión de rutas el ingreso de un usuario no autorizado no representaría un gran daño, la consecuencia podría ser que otra persona reciba el paquete lo cual se soluciona si el operario al momento de entregar el paquete verifica que esa sea la persona. Modificar estos archivos resulta grave para el negocio puesto que de estos datos van anclados con la parte de la dirección de envió y un cambio en esta puede hacer que llegue a un lugar equivocado y se refunda el paquete.
* Gestión administrativa contable de recursos y de clientes: en este aspecto es muy importante garantizar la seguridad puesto que se trata de información muy valiosa y personal de clientes y empleados. La lectura por parte de un usuario indebido es demasiado seria puesto que tendría acceso información como dirección, sueldo, cedula, etc. Modificar estos datos también resultaría grave puesto que alguien podría aumentarse el sueldo y cosas así.

1. Identifique cuatro vulnerabilidades de ese sistema, teniendo en cuenta únicamente aspectos técnicos o de procesos (no organizacionales). Identifique vulnerabilidades no solo en lo relacionado con la comunicación sino también con el almacenamiento y procesamiento de los datos. Explique su respuesta en cada caso.

* Almacenar información confidencial en el sistema de atención a clientes, errores en seguridad e implementación podría generar que un cliente tenga acceso a información de la empresa.
* Buffer overflow al servidor de rutas a la 1 de la mañana, esto se basa en que el servidor calcula las rutas a estas horas y un exceso en la carga de datos podría hacer que varias peticiones se cayeran.
* Manejo de claves débiles en el servidor de rastreo, un error por parte de la librería de java permitiría a cualquier atacante llegar a información sensible sin tener permiso.
* Falta de protocolo de seguridad(SSL) en la página de rastreos, la empresa sino garantiza la seguridad de la página web deja la puerta abierta a que cualquier atacante sniffee y robe información de los usuarios.

1. Para cada una de las vulnerabilidades que usted identificó en el punto anterior, proponga mecanismos de resolución
2. Para evitar que pase este problema (elevación de privilegios) toca mejorar la autenticación y validación de los datos. Otra solución es la asignación aleatoria del espacio de direcciones (para que sea más difícil para los desbordamientos de búfer ejecutar instrucciones privilegiadas en direcciones conocidas en la memoria)
3. Para prevenir este problema es conveniente establecer unos intervalos de tiempo en que la empresa no opera para aumentar el ramp-up y evitar el colapso por exceso de peticiones.
4. En la implementación realizada decidimos establecer unas llaves (KeyPair) de 1024 con esto hacemos que sea más difícil implementar un ataque de fuerza bruta.
5. Solucionar este problema es muy fácil y actualmente casi todas las paginas lo usan. La empresa tiene que comprar el certificado SSL y cambiar la implementación de su página web, de esta manera los datos enviados por los usuarios van a estar encriptados y no los atacantes no van a poder sniffear.

**Detalles de implementación**

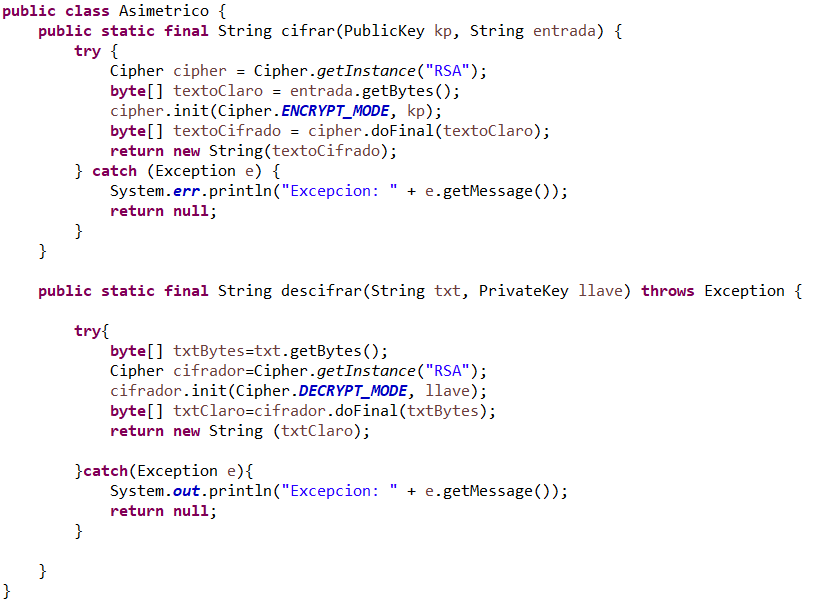
1. Cliente

La clase cliente cuenta con 2 atributos, el primero representa las llaves con las que cuenta (pública y privada). El segundo es el certificado el cual vamos a necesitar para establecer la comunicación. Al crear el cliente (inicializarlo) se crea el conjunto llaves y con este genera el certificado. Esta clase cuenta con su constructor y los métodos generarCertificado, getLlave y getCertificado. El método generarCertificado recibe como parámetro el conjunto de llaves y como resultado devuelve un certificado que en este caso tiene validez del 30/03/19 al 31/12/19. Los otros 2 métodos presentes como sus nombres lo dicen, retornan la llave (conjunto de llave privada y pública) y el certificado.



1. Asimétrico

Esta clase es la encargada de realizar el cifrado de forma asimétrica, cuenta con 2 métodos el cifrar y el descifrar. El primero recibe como parámetro la llave publica y el texto claro para finalmente devolver el texto cifrado. Para lograr este objetivo hacemos uso de la librería Cipher a la cual le indicamos el algoritmo a utilizar que en este caso es RSA. Posteriormente la iniciamos en modo inscripción y obtenemos los bytes resultantes de pasarle el texto claro. Finalmente convertimos estos bytes String y los retornamos. El segundo recibe como parámetro el texto cifrado y devuelve el texto claro, realiza lo mismo que el primer método con la diferencia de que este inicia el Cipher en modo descifrar.



1. Simétrico

Esta clase es la encargada de realizar el cifrado de forma simétrica, al igual que asimétrico cuenta con los métodos cifrar y descifrar. El primero recibe como parámetro la llave publica y el texto claro para finalmente devolver el texto cifrado. Para lograr este objetivo hacemos uso de la librería Cipher a la cual le pasamos el padding que contiene la información para implementar el algoritmo AES. Posteriormente la iniciamos en modo inscripción y obtenemos los bytes resultantes de pasarle el texto claro. Finalmente convertimos estos bytes s String y los retornamos. El segundo recibe como parámetro el texto cifrado y realiza lo mismo que el primer método con la diferencia de que este inicia el Cipher en modo descifrar.

