Juan Camilo Ruiz – 201617394

Jorge Andres Gomez – 201618492

Infraestructura Computacional

Universidad de los Andes

**Caso 2**

1. **Análisis y entendimiento del problema**
2. **Identificación datos por proteger**

Los datos que deben ser protegidos en el sistema de rastreo de unidades de distribución son la información que es enviada por cada dispositivo asociado a una unidad de distribución al servidor encargado del manejo de unidades, donde dicha información corresponde a la ubicación geográfica (posición) de la unidad. Dicha información debe ser protegida debido a que, si un actor no autorizado consigue acceso a el, en el caso de la escritura, podría mandar datos falsos sobre la ubicación de las unidades haciendo creer a la empresa que hizo todas las entregas cuando en realidad no, o que no efectuó ninguna entrega siendo que, si se hicieron, en otras palabras, causaría inconsistencias en el sistema. Por otro lado, en el caso de que un actor no autorizado tuviera acceso en modo lectura, también afectaría bastante a la empresa debido a que dicho actor sabría las rutas y paradas de las unidades distribuidoras, lo cual pone en riesgo no solo a la empresa sino a los propios trabajadores que conducen dichas unidades y los paquetes que son transportados. De este modo, de manera general, si un actor no autorizado consigue acceso a los datos, y ocurre bien sea o fugas de información o inconsistencias por información falsa, la empresa sería afectada de diversas formas, perdiendo tanto clientes como asociados.

1. **Vulnerabilidades**
   * **Spoofing:** Al no verificar la autenticidad de los dispositivos ni proteger el canal usado entre el servidor y los dispositivos, es posible que ocurra el ingreso de un dispositivo falso.
   * **Information disclausure:** La comunicación entre el dispositivo y el servidor de manejo de unidades no está protegida, lo cual permite que la información sea interceptada, causando fugas de información.
   * **Integridad de la información:** Como la comunicación entre el servidos y los dispositivos no está protegida, un actor no autorizado podría tener acceso y mandar información falsa, lo cual afectaría la integridad de la información.
   * **Floofing:** Por la falta de protección entre la comunicación entre el servidor y las unidades podría llegar a ocurrir una saturación de recursos, donde por envíar tanta información se consume toda la memoria disponible.
2. **Soluciones propuestas**
   * **Certificado digital:** Usamos un certificado digital para autenticarnos con el servidor y evitar spoofing. Ademas, de esta manera hacemos llegar una llave publica (de un par de llaves K+ K-), valida en un periodo de tiempo equivalente a la validez del certificado, de modo que podamos intercambiar mensajes de manera segura en un sentido con encripción asimétrica. Además, el servidor nos envía un certificado digital, para obtener su llave publica y poder mediante encripción asimétrica enviar mensajes en esta dirección.
   * **Encriptación:** Con la llave publica nuestra, el servidor nos envía un mensaje cifrado por RSA en el cual esta una llave para encripción simétrica. La desciframos con la llave privada nuestra y así podemos empezar a enviar mensajes cifrados con encripción simétrica. Particularmente, enviaremos las coordenadas por este medio, de modo que solo el servidor y el cliente conocen la llave para descifrar las coordenadas.
   * **Hashing:** Para evitar que el mensaje cifrado de manera simétrica sea intervenido y editado, aplicamos sobre el mensaje original una función de hash. Ya que la función de hash es difícilmente reversible y su probabilidad de colisión es baja, sabemos que si se edita el mensaje cifrado antes de que llegue al servidor, el resultado del hash sobre el mensaje descifrado no será el mismo que el hash, permitiendo asegurar la integridad de la información. Además, como una medida de seguridad extra, enviamos el hash al servidor cifrado de manera asimétrica con la llave publica que nos envió en el certificado digital. Esto con el fin de que en caso de que el atacante le sea aun mas complicado reversar la función de hash.