Universidad de los Andes
Ingeniería de Sistemas y Computación
ISIS 2203 Infraestructura Computacional

Caso de Estudio 2 – Canales Seguros
Sistema de Gestión Empresarial y Operativa de una Compañía Transportadora

Objetivos

- Identificar los requerimientos de seguridad de los canales usados para transmisión de la información en el sistema de gestión empresarial y operativa de una compañía transportadora.
- Construir un prototipo a escala del sistema que permita satisfacer algunos de los requerimientos de seguridad identificados. Entendiendo las garantías de seguridad y las limitaciones de la implementación propuesta.

Problemática:

Como se indicó en el documento que describe el contexto del caso, las principales tareas del sistema son la recepción de órdenes de recogida, gestión de rutas, rastreo de unidades de distribución y paquetes, y gestión administrativa contable de recursos y de clientes.
En este contexto, surgen diferentes problemas de seguridad para algunas de las transacciones que el sistema soporta, tanto a nivel de transmisión, como en procesamiento y almacenaje de datos. Como consecuencia, es necesario evaluar riesgos y determinar medidas para mitigar los problemas detectados. Su tarea en este caso es actuar como consultor de seguridad y analizar la seguridad de las tareas relacionadas con el rastreo de unidades de distribución.

Tareas:

Suponga que la arquitectura del sistema incluye tres servidores en la oficina principal: uno se encarga del manejo y rastreo de unidades de distribución y paquetes, el segundo del manejo de órdenes de recogida, y el último se encarga del manejo administrativo y contable de recursos y clientes.

- Los puntos de atención al cliente se comunican por medio de internet con el servidor de manejo de órdenes para registrar pedidos y contratos.
- Para el rastreo de unidades de distribución y paquetes y optimización de rutas, las unidades se comunican cada 180 segundos con el servidor para informar su estado. El servidor recibe la información y la procesa. Por otro lado, el servidor de manejo de unidades de distribución calcula diariamente a la 1 a.m. las rutas del día. En condiciones excepcionales, los conductores pueden cambiar las rutas asignadas pero deben informar.
- El servidor de manejo de órdenes se comunica con el de rastreo y rutas: las rutas se calculan con base en los puntos de atención que han recibido paquetes.
- El servidor de manejo administrativo contable no atiende consultas de clientes vía web; solamente responde a consultas iniciadas en la intranet de la compañía.
- Todos los servidores implementan control de acceso a nivel del sistema operativo y ejecutan transacciones solo para usuarios autenticados, de acuerdo con los permisos asignados.
- Las aplicaciones también manejan usuarios, cada una maneja su propio archivo de configuración de usuarios y soporta operaciones de cifrado con una librería que implementa algoritmos propios.

A. [20%] Análisis y Entendimiento del Problema

Considerando el sistema descrito en el párrafo anterior:

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?
2. 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.
3. Para cada una de las vulnerabilidades que usted identificó en el punto anterior, proponga mecanismos de resolución.

a. Los mecanismos propuestos deben ser explicados. Por ejemplo, si se habla de cifrado sobre un canal de comunicaciones, debe identificar los participantes en la comunicación, y si es cifrado simétrico o asimétrico (y justificar la decisión).
b. Además, debe explicar por qué resuelve la vulnerabilidad identificada

(*) *Sus explicaciones DEBEN corresponder al contexto planteado (de forma explícita). NO se aceptarán respuestas para contextos genéricos*.

B. [80%] Implementación del Prototipo

En esta parte del proyecto nos centraremos únicamente en el sistema de rastreo de unidades de distribución.

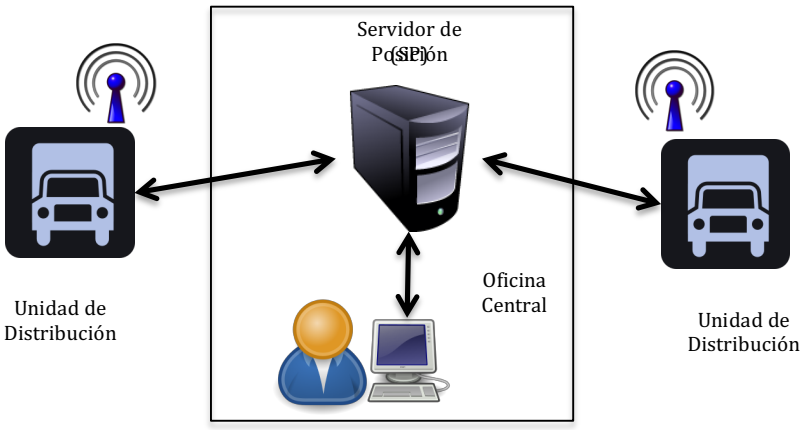

Figura 1. Sistema de distribución

Su tarea consiste en construir el cliente de una unidad de distribución que se comunique con el servidor de rastreo para reportar su estado (ubicación geográfica). El cliente y el servidor se comunicarán siguiendo el protocolo descrito en la Figura 2.

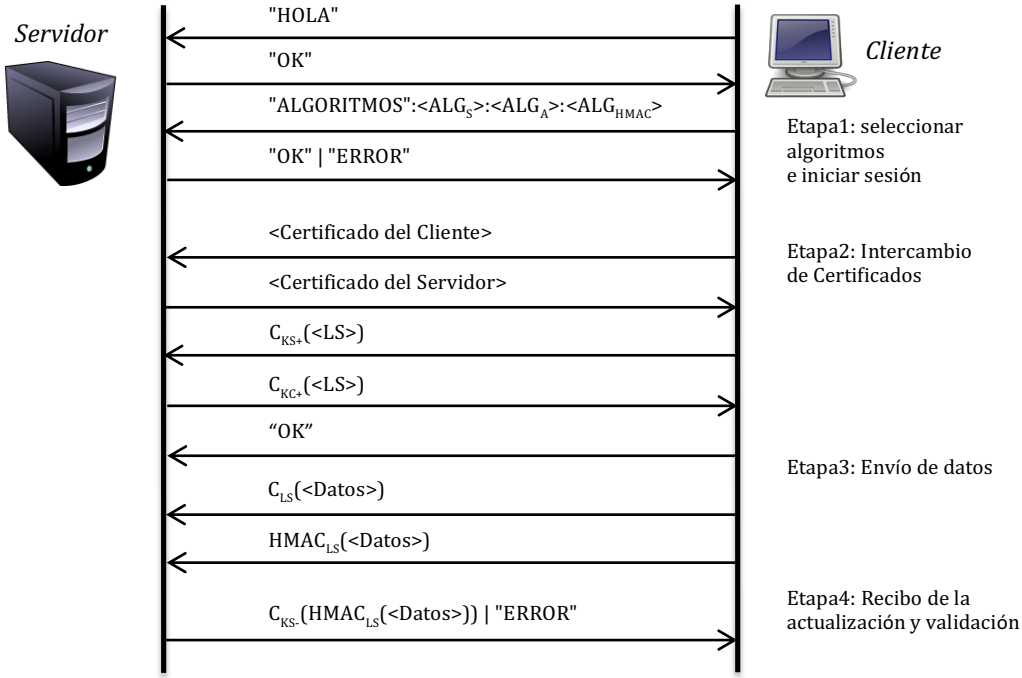

Figura 2. Protocolo de comunicación entre cliente y servidor (con seguridad).

También manejaremos un servidor sin seguridad. El cliente y el servidor sin seguridad se comunicarán siguiendo el protocolo descrito en la Figura 3.

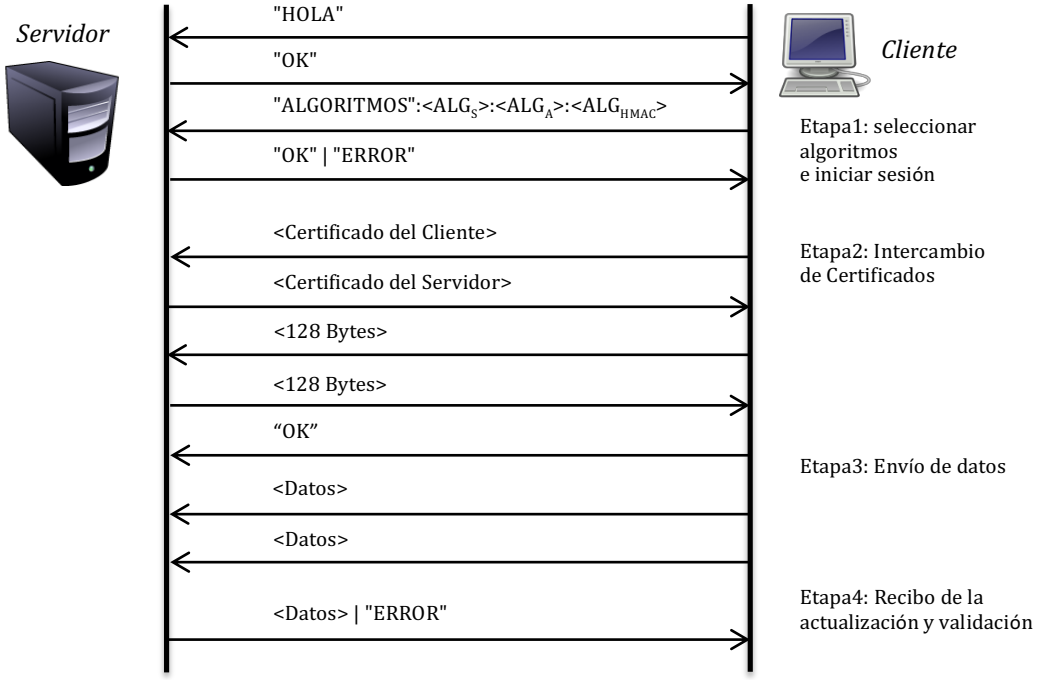

Figura 3. Protocolo de comunicación sin seguridad.

TENGA EN CUENTA:

- El protocolo de comunicación maneja:
  - Cadenas de Control: "HOLA", "ALGORITMOS", "OK", "ERROR".
  - Separador Principal: " : "
- A continuación se presentan los algoritmos disponibles en el servidor para manejo de integridad y confidencialidad. Es decir, usted debe seleccionar un algoritmo para reemplazar cada una de las cadenas: $<ALG_S>$, $<ALG_A>$ y $<ALG_{HMAC}>$ en el protocolo. Los algoritmos disponibles son:
  - Simétricos (ALG_S):
    - AES (Configuración por defecto: Modo ECB, esquema de relleno PKCS5, llave de 128 bits).
    - Blowfish (Configuración por defecto: Cifrado por bloques, llave de 128 bits).
  - Asimétricos (ALG_A):
    - RSA. (Configuración por defecto: Llave de 1024 bits.)
  - HMAC (ALG_{HMAC}):
    - HmacSHA1
    - HmacSHA256
    - HmacSHA384
    - HmacSHA512

  Las cadenas que identifican cada uno de los algoritmos son: "AES", "BLOWFISH", "RSA", "HMACSHA1", "HMACSHA256", "HMACSHA384", "HMACSHA512".
- Utilizaremos el estándar X509 para los certificados digitales (CD). La idea es que el cliente y el servidor comprueben la identidad del servidor a partir de un CD (en un caso real este debería ser expedido por una entidad certificadora pero aquí se va a generar localmente). El CD debe contener, entre otros, la llave pública para usarla en el proceso de comunicación.
- Usaremos la librería BouncyCastle para el manejo de certificados. Se recomienda usar el método X509v3CertificateBuilder.
- La comunicación se realiza a través de sockets de acuerdo con el protocolo de comunicación definido.

- La cadena <DATOS> debe reemplazarse por un identificador de cliente y una posición. El identificador va separado por el carácter ; de la posición. La posición se manejará como dos parejas de números (grados y minutos en decimal), separados por una coma ",". Ejemplo de dato (para el cliente con id 15): 15;41 24.2028,2 10.4418 (coordenadas usadas por Google).
- Dado que existen problemas en la transmisión de los bytes cifrados, manejaremos encapsulamiento con cadenas hexadecimales para transmisión de datos de este tipo. Es necesario codificar en hexadecimal la información cifrada, para luego enviarla. El servidor hará lo mismo. Use DatatypeConverter para construir los métodos apropiados:
```
DatatypeConverter.printHexBinary(byteArray);
DatatypeConverter.parseHexBinary(cadena);
```
- A continuación se muestra un ejemplo del código para enviar el certificado. Primero se encapsula y luego se envía.
```
java.security.cert.X509Certificate certificado = generarCertificado(llaves);
byte[] certificadoEnBytes = certificado.getEncoded( );
String certificadoEnString = printHexBinary(certificadoEnBytes);
socketParaComunicacion.println(certificadoEnString);
```
- El cliente debe validar el recibo que envía el servidor en la última etapa de la comunicación.
- Las librerías de BouncyCastle y el .jar del servidor serán publicadas en SICUA+.
- Además, se publicará el .jar servidor sin seguridad.

Entrega:

- Cada grupo debe entregar un zip de un proyecto Java con: La implementación correspondiente al cliente (descrito en la parte B). En el subdirectorio docs debe haber un archivo que incluya el informe con las respuestas a la parte A. **Al comienzo del informe, deben estar los nombres y carnés de los integrantes del grupo**. Si un integrante no aparece en el documento entregado, el grupo podrá informarlo posteriormente. Sin embargo habrá una penalización: la calificación asignada será distribuida (dividida de forma equitativa) entre los integrantes del grupo..
- El trabajo se realiza en grupos de **2** personas. No debe haber consultas entre grupos.
- El grupo responde solidariamente por el contenido de todo el trabajo, y lo elabora conjuntamente (no es trabajo en grupo repartirse puntos o trabajos diferentes). Se puede solicitar una sustentación a cualquier miembro del grupo sobre cualquier parte del trabajo. Dicha sustentación será parte de la calificación de todos los miembros.
- El proyecto debe ser entregado por Sicua+ por uno solo de los integrantes del grupo.
- **La fecha límite de entrega es el 4 de abril, 2019 a las 23:55 p.m.**

Referencias:

- *Cryptography and network security*, W. Stallings, Ed. Prentice Hall, 2003.
- *Computer Networks*. Andrew S. Tanenbaum. Cuarta edición. Prentice Hall 2003, Caps 7, 8.
- *Blowfish*. Página oficial es: http://www.schneier.com/blowfish.html
- *RSA*. Puede encontrar más información en: http://www.rsa.com/rsalabs/node.asp?id=2125
- *CD X509*. Puede encontrar la especificación en: http://tools.ietf.org/rfc/rfc5280.txt