IIC3253 — Criptografía y Seguridad Computacional

Tarea 3

Pregunta 1

En clases se mencionaron ciertas consecuencias negativas que conlleva el autentificar y autorizar a usuarios en la Web en base a sus nombres de usuario (o correos electrónicos) y contraseñas.

(a) Explique en detalle cuáles son estas consecuencias, y describa casos hipotéticos que evidencien que son negativas.

Respuesta: Las principales consecuencias de usar usuario y contraseña en la Web son que se manda en texto plano la llave secreta al servidor, por lo que se basa en la confianza de como manejan los datos en el servidor o en la empresa de la aplicación que se está utilizando, ya que no sabemos si están siguiendo los protocolos para guardar las claves, más aún depende de la confianza de los empleados de la empresa, ya que al autenticarse podrían ver algún log con los datos que se envían y podrían obtener la contraseña fácilmente, otra consecuencia es que no existe auditabilidad, es decir, no hay forma de comprobar si un mensaje válido no fue enviado por el usuario, por ejemplo, una aplicación que tenga almacenada la tarjeta de crédito de un usuario podría hacer un cargo comprando un servicio de la aplicación y no se podría comprobar que no fue el usuario quien realizo esa acción.

(b) Diseñe una alternativa para autentificar y autorizar usuarios en la web que no traiga consigo las consecuencias negativas mencionadas. Explique cómo funcionaría su sistema de autenticación/autorización, y qué problemas prácticos podría traer consigo. Finalmente, explique qué medidas tomaría para prevenir dichos problemas.

Respuesta: Una alternativa posible es un sistema con firmas digitales basado en llave pública y llave privada, como Schnorr, y mandar en cada solicitud al servidor el mensaje más la firma, así para autenticarse solo seria necesario mandar el nombre de usuario y luego mandar este en el header de cada request para que el servidor pueda buscar la llave pública del usuario fácilmente y para la autorización no hay ningún problema, ya que cada mensaje va firmado por lo que el servidor puede comprobar la identidad del usuario. Los problemas que tiene esta solución son: El usuario tiene que generar la llave pública y llave privada, cómo se guarda la llave privada, el usuario tiene que ejecutar el algoritmo para firmar los mensajes. Para solucionar esto el usuario podría ejecutar un script que genere las llaves y las guarde en alguna parte del computador, luego el algoritmo para firmar se puede implementar desde el código javascript de la web y solo pedir la llave privada, además después de autenticarse

se podría generar un token que sirva para autorizar las acciones menos importantes de la aplicación y solo pedir la firma para la acciones importantes.