Seguridad y Privacidad en Entorno de Aplicaciones

Máster Profesional en Ingeniería Informática

Universidad de Málaga

Curso académico 2018-2019

End-to-end Encryption. MQTT

López Olmedo, Oliver - Fernández Cruz, Carmen Jackeline

25 de marzo de 2019

1. Resumen

La práctica consiste en realizar un cliente y servidor que utilizando el protocolo mqtt acuerden mediante un canal público una clave común de cifrado. La clave se acuerda utilizando el algoritmo Diffie-Helman para el intercambio seguro de claves en medios inseguros. Sin embargo, este método permite únicamente acordar una clave común entre dos extremos, no así la autenticación de ninguno de ellos. Por tanto, más adelante analizaremos los distintos escenarios y posibles métodos para autenticarse.

2. Código

Se incluyen dos archivos, correspondiendo al código que ejecuta el servidor y el que ejecuta el cliente. Comenzaremos explicando los puntos claves del código del servidor:

- Cuando se inicia el servidor genera los parámetros públicos que compartirá para que el cliente pueda generar su par de claves. Una vez creados, genera su par propio de claves y transmite cada 5 segundos los parámetros por el topic "Pu_IoT_DH".
- Previamente, se había suscrito a los tres topics que se utilizarán: "Pu_IoT_DH", "K_Exc", y "Cypher", por tanto, cada vez que recibe un mensaje de cualquiera de esos tres topics se comprueba de qué topic procede.
- Si el mensaje es recibido por el topic "K_Exc"quiere decir que alguien ha enviado su clave pública. El servidor en primer lugar publica por el mismo topic su clave pública, y a continuación, utilizando la clave pública recibida y su clave privada deriva la clave compartida y envía un mensaje de prueba al topic "Cypher".
- Cuando se recibe por el topic "Cypher" se descrifra el mensaje y se realiza un "print". A partir de este punto se puede pedir que el usuario escriba un mensaje y se cifra automáticamente para enviarlo por dicho topic, aunque no está implementado debido a que no pudimos testearlo.

En el caso del cliente el funcionamiento implementado es el que sigue:

- Cuando el cliente se conecta se suscribe a los mismos topics que el servidor.

- En cuando recibe los parámetros enviados por el servidor genera su par de claves y envía su clave pública para que el servidor pueda derivar la clave compartida.
- Cuando recibe un mensaje por el topic "K_Exc"supone que es la clave pública enviada por el servidor e intenta derivar la clave compartida.
- En este punto se queda escuchando al topic "Cypher" para utilizar la clave compartida derivada y descifrar los mensajes que reciba por él.

Un detalle importante a tener en cuenta es que cuando se envía un mensaje por un topic y a la vez se está suscrito a él, el propio extremo también lo recibe. Por tanto, tanto cliente como servidor se aseguran que la clave recibida por el canal "K_Exc"no es la suya propia.

3. Escenarios

Como hemos comentado anteriormente no es posible autenticar al otro extremo durante el intercambio de claves ni una vez la hemos derivado. Por tanto, según el escenario podemos realizar ciertas acciones para intentar verificar que estamos comunicándonos con quién esperamos.

3.1. Cliente únicamente con input

Si el cliente únicamente dispusiera de input, como por ejemplo un teclado, una vez se ha derivado la clave compartida y se puede mantener una comunicación segura se podría acordar que el primer mensaje intercambiado fuera una clave o mensaje acordado previamente que demostrase que el cliente es quién se espera que sea.

3.2. Cliente únicamente con output

En este caso la autenticación se podría realizar de forma inversa. El servidor podría enviar una clave o mensaje acordado previamente y comprobar en el lado del cliente que el mensaje recibido es el esperado.

3.3. Cliente sin I/O

Este es el caso en el que la autenticación se dificulta, incluso podría entenderse como imposible. Si en el cliente no tenemos ninguna forma que permita introducir manualmente información, ya sea con teclado, botones, etc, ni tampoco tenemos ningún método que nos indique que el mensaje recibido es el esperado no tenemos garantías de que el dispositivo es el que dice ser. Se podría incluir algún tipo de certificado en el dispositivo que de alguna forma permitiera autenticarlo, aunque habría que tener presente que podría ser modificado o suplantado, pero de otra forma no tendríamos ninguna garantía directamente.