Proyecto Final: Descripción de la solución metodología y diseño.

Resumen:

El trabajo presenta una implementación de generación de llaves criptográficas basado en señales de Electrocardiograma(ECG). Este sistema sería implementado en una red de sensores corporal de tipo IoT donde se requiere un acuerdo de llaves entre dispositivos sin un tercero.

Se implementó el acuerdo de estas llaves entre 2 dispositivos. Manteniendo confidencialidad uno al otro sobre los datos biométricos registrados. El resultado principal del proyecto es un esquema de acuerdo de llaves, donde estas son generadas principalmente por los datos biométricos en común de los sensores participante.

Dadas las capacidades de dispositivos de IoT, se realizará este acuerdo de llaves para encriptación simétrica. Dada las capacidades de los dispositivos actuales, donde el consumo de energía de relevante para la duración de la transferencia de datos, se requiere un esquema que equilibre la seguridad y la potencia de procesamiento.

Descripción del problema:

1. La poca capacidad de procesamiento de algunos dispositivos usados en el IoT. Estos dispositivos se centran sus capacidades en las principales caraterísticas de la telecomunicación como lo son: tiempo, distancia y energía.

2. Muchas de las llaves en dispositivos son seleccionadas por el fabricante. Se presenta el problema de acuerdo de llaves sin un tercero. Estos dispositvos pueden estar sujetos a manipulación electrónica lo que sugiere que no sean seguras las llaves brindadas por un fabricante.

3. Balance entre procesamiento y seguridad. Muchas de las funciones hash, así como el cifrado de llave publica requieren mayor procesamiento y gestión de llaves.

4. Manipulación y/ó secuestro de datos sensibles que podrían llevar a un mal diagnóstico o retrasar alguna intervención médica.

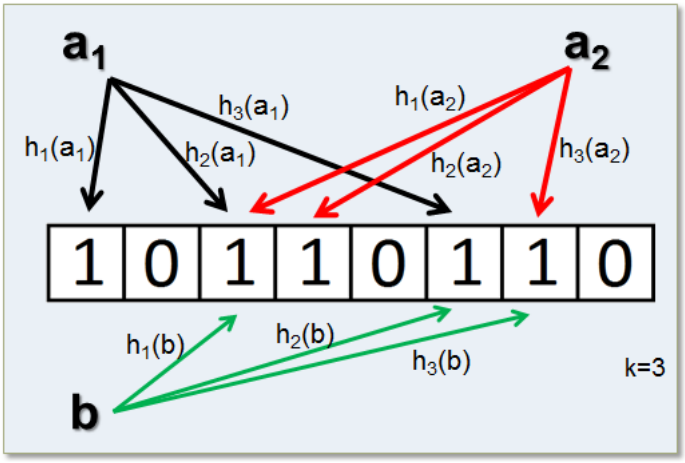
Marco teórico:

1. Filtro de Bloom(1970)

Un filtro de Bloom es una estructura de datos probabilística, concebida por Burton Howard Bloom en 1970, que es usada para verificar si un elemento es miembro de un conjunto. Los falsos positivos son posibles pero los falsos negativos no.

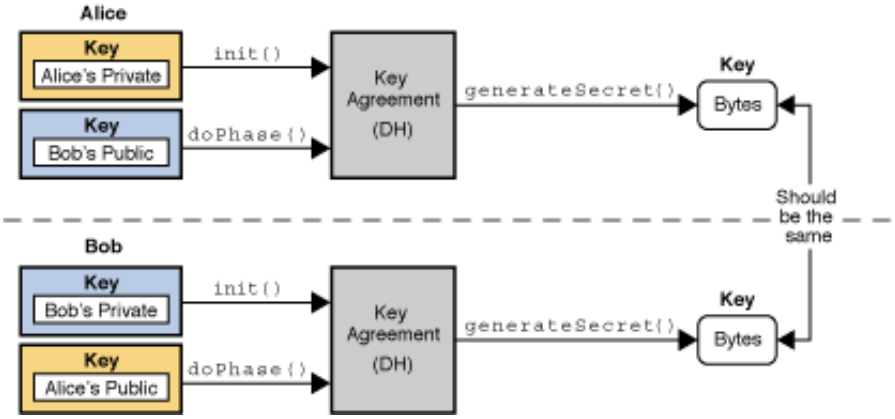
Los filtros de Bloom se suelen usar para:

1. Acelerar la búsqueda de elementos. Si el filtro devuelve false entonces es seguro que el elemento no se encuentra en el conjunto
2. Realizar búsquedas sin especificar claramente lo buscado (protegiendo su privacidad)



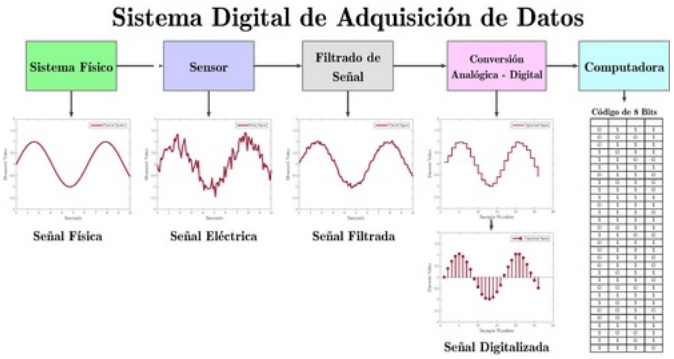
1. Acuerdo de llaves.

También llamados **protocolos de intercambio de claves** (en inglés *key exchange protocols*) es un protocolo criptográfico en el que se establece una secuencia de pasos entre dos o más participantes a través de la cual los participantes se ponen de acuerdo en el valor de una información secreta compartida. A la información secreta compartida se le suele llamar clave debido a que esa información se suele usar como clave de algún algoritmo criptográfico.



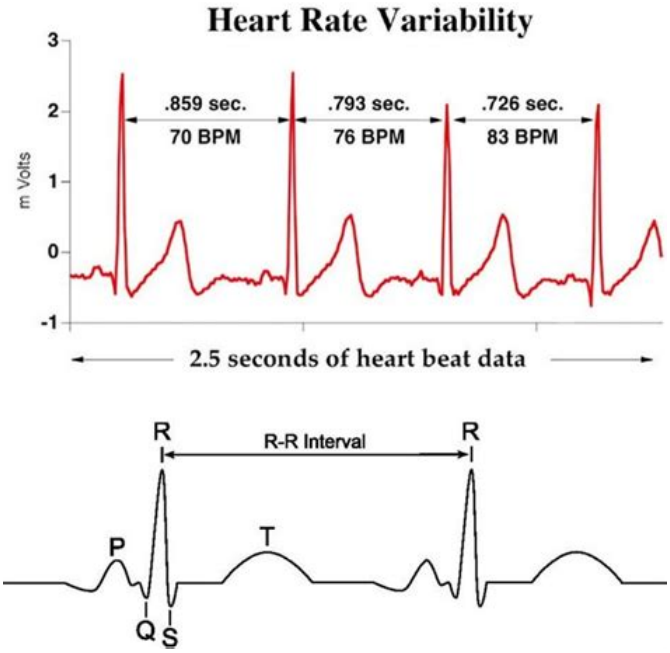
2. Señales fisiológicas.

En el cuerpo existen diversas señales electricas, muchas de ellas aportan información más alla de la forma de onda. En este caso de utilizará señales de electrocardiograma(ECG) preadquiridas de una base de datos.



3. ECG - IPI - Inter Pulse Interval

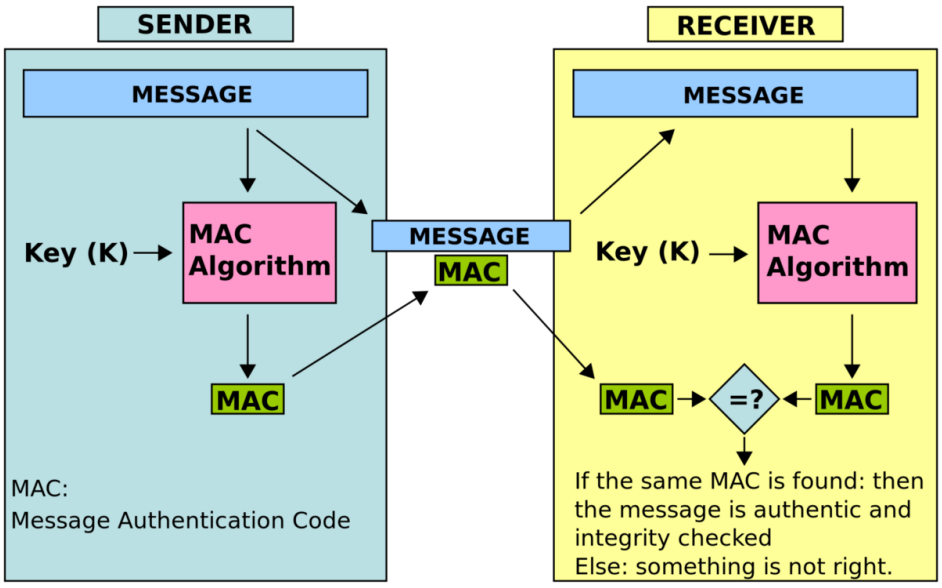
Los sistemas de clasificación de señales cardiacas normalmente actúan junto a módulos de adquisición y pre-procesamiento de las señales eléctricas del corazón. Estos módulos son requeridos debido a que dichos sistemas necesitan que la señal obtenida sea lo más clara posible y libre de errores en el proceso de adquisición. Las señales eléctricas de corazón, o el empleo de un electrocardiograma (ECG), permiten conocer el comportamiento de dicho órgano. Estas señales indican un registro toda la actividad realizada por este músculo y a través de ellas es posible identificar cuando el corazón funciona normalmente o tiene alguna falla.



4. Funciones HMAC.

En la criptografía, un **código de autentificación de mensajes en clave-hash** (**HMAC**) es una construcción específica para calcular un código de autentificación de

mensaje (MAC) que implica una función hash criptográfica en combinación con una llave criptográfica secreta. Como cualquier MAC, puede ser utilizado para verificar simultáneamente la *integridad de los datos* y la *autentificación* de un mensaje. Cualquier función hash criptográfica, tales como MD5 o SHA-1, puede ser utilizada para el cálculo de un HMAC; el algoritmo MAC resultante se denomina HMAC-MD5 o HMAC-SHA1 en consecuencia. La fuerza criptográfica del HMAC depende de la potencia criptográfica de la función de hash subyacente, el tamaño de su salida de hash y el tamaño y calidad de la llave.



Descripción de solución

1. Tomar ventaja de la similaridad de algunas señal fisiológica muestreadas en un mismo intervalo en el mismo cuerpo.

2. Creación de llaves con estas señales para la comunicación segura entre 2 dispositivos. Utilizar una esquema de bóveda difusa como el filtro de Bloom y los índices de aquellos valores en común de un vector de daros de las señales de A y B. Dónde A y B son 2 sensores cualesquiera en el cuerpo y possen la capacidad de registrar ECGs.

3. Evitar transferencia de datos de la señales o pista alguna sobre estos durante el acuerdo de llaves. Estos datos pueden ser secuestrados o modificados.

Metodología

1. Generación de patrones.
    1. Digitalización de la señal en Sensor A y B.
    2. Extracción de patrones de la señal en sensor A y B.
2. Intercambio de patrones
    1. Los sensores hacen conocer sus datos de manera difusa.
        1. Creación de Filtro de Bloom con datos de sensor A
        2. Búsqueda de datos de B en FB de A.
3. Generación de llaves.
    1. Generación de llave en sensor B.

    2. Conocimiento de patrones en común en A.

    3. Dado el paso b, A genera y verifica que su llave coincida con la de B.

    4. B determina si el proceso falló o fue exitoso.

Diseño

1. **Generación de patrones**
    1. Digitalización de la señal en Sensor A y B.
        1. Se obtuvieron señales de ECG de la base de datos de PhisyioBank. Específicamente de la base de datos MIT-BIH Arrhythmia.
        2. El array de patrones representa la concatenación de los valores de IPI en binario, de 3 segmentos recorridos por una muestra. (30 patrones) (Recorrido total de 90 muestras)
    2. Extracción de patrones de la señal en sensor A y B.
        1. Se seleccionó el IPI como patrón a obtener de la señales.
            1. Dada la señal, se debe aplicar un pre - procesado. En este caso de utilizará algún kernel para hacer correlación cruzada y obtener picos en la señal más finos.
            2. Se definirá un umbral, se registrará la posición de las muestras de lo sobrepasen. Dados los índices se obtendrá el diferencial entre ellos.

            1. Se obtendrá el promedio entre valores que pasen el umbral no consecutivos.

2. Intercambio de patrones

1. Los sensores hacen conocer sus datos de manera difusa.
    1. Creación de Filtro de Bloom con datos de sensor A.

    1. Dada una cantidad estimada de elemento y una probabilidad de falsos positivos de determina una longitud para el filtro de Bloom. El filtro indica que algún elemento **podría** estar o **definitivamente** no estar en el filtro.
    2. Crear objeto de Filtro de Bloom, que dados estos parámetros establezca su longitud en bits, tendrá la capacidad de añadir y verificar, además A envía un número aleatorio a R.

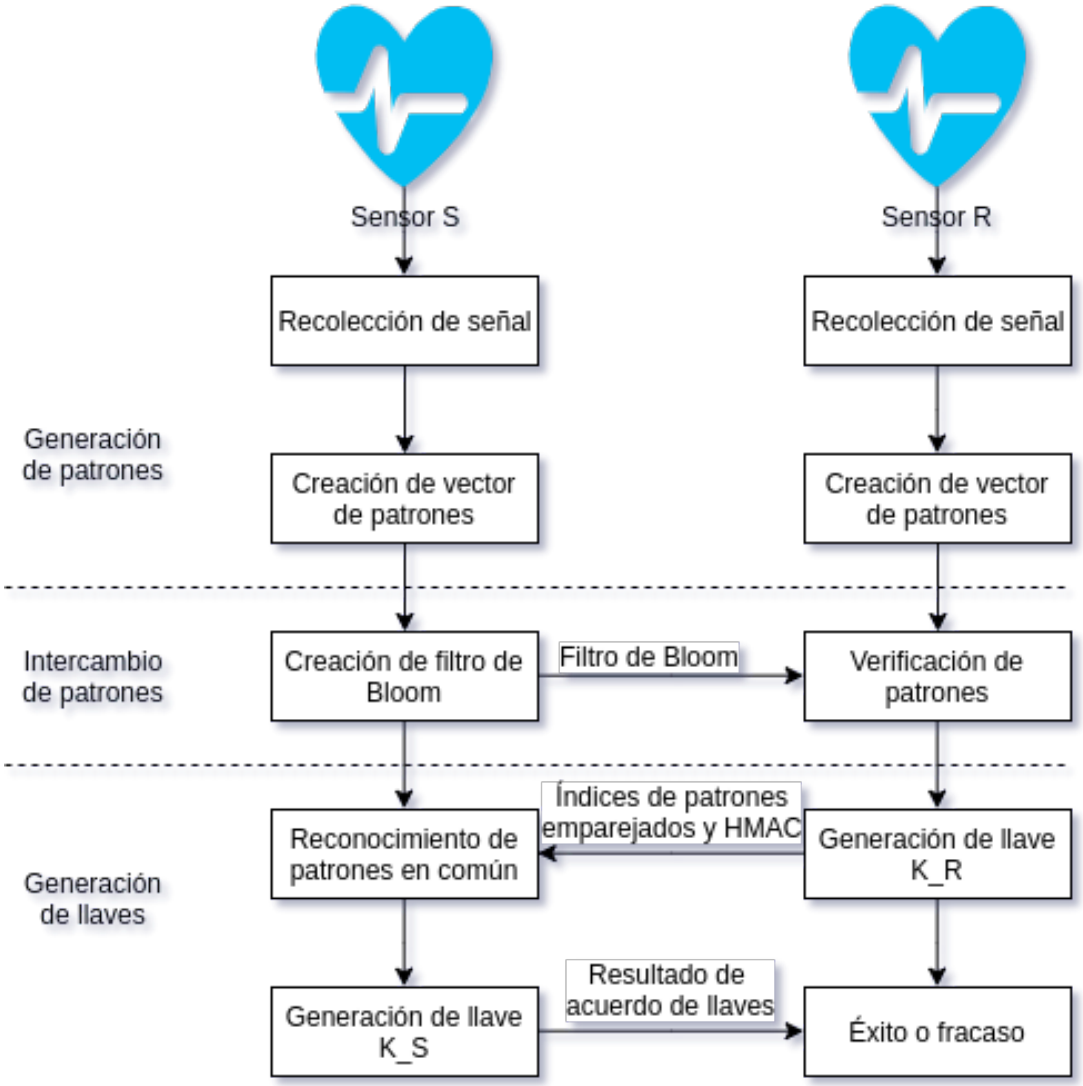
2. Búsqueda de datos de B en FB de A.

    1. B puede verificar qué datos muestreados por él están en A sin compartirlos. Haremos uso del objeto de filtro de Bloom y su método de verificar. Se obtendrán lo índices de los elementos que estén en dicho filtro.

3. Generación de llaves

1. Generación de llave en B
    1. Dada la concatenación de los patrones con los índices en común y el número aleatorio, el hash SHA-1 de la concatenación de estos datos, daría como resultado la llave en B.
    2. En este paso realizaremos un HMAC compuesto por KeyB y el mensaje Indices || Numero aleatorio. Este en enviado.
2. Conocimiento de patrones en común en A.
    1. B envía sus índices en común con A. A genera la misma concatenación con lo recibido.
3. A genera y verifica que su llave coincida con la de B.
    1. A genera su llave con los índices que recibió de B. La llave de A es el resultado del mismo proceso en B y un hash SHA-1 de este.
    2. Procede a componer un HMAC con su llave KeyA y el mensaje Indices || Numero aleatorio. Si los HMACs coinciden, entonces Key A == Key B. Se envía otro HMAC, donde Key A y el mensaje (IDa || IDb || Numero Aleatorio)

4. B determina si el proceso falló o fue exitoso. Puede verificar el HMAC con su llave.



Implementación

El desarrollo será en Python, lo más modular posible para cada paso en el diseño.

1. Librerías usadas
    1. Filtro de Bloom
        1. Math
        2. Hashlib
        3. Bitarray
    2. Calculo de IPI
        1. Numpy
        2. os

3. shutil
4. wfdb

3. Acuerdo de llaves
    1. hmac(extensión de hashlib)
    2. numpy
    3. random

Pruebas finales

1. Prueba existosa

Representación en hexadecimal de HMAC-R: 7207df095bc3f1583dee49c3909fce65813994ba

Representación en hexadecimal de HMAC-S: 7207df095bc3f1583dee49c3909fce65813994ba

Si las MACs coinciden entonces, se mantiene Kr == Ks, sin que ninguno conozca la del otro. Correcto acuerdo de llaves Correcto acuerdo de llaves…

1. Prueba fallida a propósito

Representación en hexadecimal de HMAC-R: c4827ef43d855d4e2ce7d31db03f8402348fb6db Representación en hexadecimal de HMAC-S: ecdbf3419cff6fd03bd957423e4d073267300bfe Sin coincidencia en MACs... C no coincide, fallo en protocolo. Fallo en la verificación de llaves.

Más detalles en el código anexado y en https://github.com/SenoReload/PhysioKeyGen

Referencias

1. *Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov PCh, Mark RG, Mietus JE, Moody GB, Peng C-K, Stanley HE. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals (2003). Circulation. 101(23):e215-e220.*
2. *Moody GB, Mark RG. The impact of the MIT-BIH Arrhythmia Database. IEEE Eng in Med and Biol 20(3):45-50 (May-June 2001). (PMID: 11446209)*
3. *Using Bloom Filter to Generate a Physiological Signal-Based Key for Wireless Body Area Networks(Yao et al, 2019)*
4. *Key Establishment Protocol for a Patient Monitoring System Based on PUF and PKG(Diaz)*
5. *Alfred J. Menezes, Paul C. Van Oorschot, Scott A. Vanstone. "Handbook of applied cryptography". CRC Press, 1997.*