UNIVERSIDAD POLITECNICA DE VALENCIA

DEPARTAMENTO DE COMUNICACIONES

Máster Universitario en Tecnologías, Sistemas y Redes de Comunicaciones

Destilación de Clave Cuántica

Autor: Juan Vicente Pradilla Cerón

Director 1: José Mora Almerich Ph.D.

Director 2: José Capmany Francoy Ph.D.

Fecha de comienzo: 16/06/2012

Lugar de trabajo: Grupo de Comunicaciones Ópticas y Cuánticas del iTEAM

Objetivo — Implementar y testear un sistema de destilación de clave cuántica compatible con el sistema de distribución de clave cuántica desarrollado por el Grupo de Comunicaciones Ópticas y Cuánticas del iTEAM basado en el protocolo BB84.

Metodología — Analizando los resultados experimentales del sistema de distribución de clave anteriormente desarrollado, se ha diseñado e implementado el proceso de destilación de clave cuántica del protocolo BB84. Posteriormente, se han realizado pruebas simuladas para valores concretos de tamaño de clave y tasas de errores cuánticos (QBER), determinando el comportamiento del tamaño de la clave, del tiempo de procesamiento y de los errores en la clave, caracterizando así el funcionamiento del protocolo BB84.

Desarrollos teóricos realizados — Simulaciones del comportamiento del sistema de destilación de clave cuántica para tamaños de clave inicial de 128, 256 y 512 qbits (bits cuánticos) y valores de QBER entre 1% y 14%. Análisis de los resultados de las simulaciones y propuesta para la amplificación de la tasa de bit para distribución de clave cuántica basa en funciones hash criptográficas.

Resultados — Caracterización del funcionamiento del sistema de destilación de clave cuántica y sus subcapas, propuesta para el aumento de la tasa de bit para distribución de clave cuántica y las primeras pruebas teóricas.

Líneas futuras — Implementación de otros protocolos para la distribución de claves cuánticas, desarrollo de diferentes algoritmos para la corrección de errores, verificación experimental de los mecanismos de detección de errores, comprobación matemática de la propuesta del usos de funciones hash criptográficas para ampliar la tasa de bit cuántica, integración en una red de distribución de clave cuántica, tele-portación de información y distribución de clave cuántica sobre medios de transmisión no guiados.

Publicaciones — Artículo en memoria y póster en el congreso internacional IEEE Topical Meeting on Microwave Photonics (MWP 2013), Oct 28-31 de 2013, Alexandria - USA. Artículo en memoria y poster en la VIII Reunión Española de Optoelectrónica, Jul 10-12 de 2013, Alcalá de Henares – España. También se ha iniciado el proceso de publicación en la revista internacional IEEE/OSA Journal of Lightwave Technology, en la Revista Óptica Pura y Aplicada (OPA) y en el Primer Congreso Nacional de I+D en Defensa y Seguridad, Nov 2013.

Resumen — Esta tesina presenta un sistema de destilación de clave cuántica para el protocolo BB84. La finalidad es implementar y testear el sistema para integrarlo en una solución completa de distribución de clave cuántica de alto rendimiento. Además, se pretende identificar y proponer puntos de mejora en el proceso de destilación a fin de incrementar la tasa de bit de distribución de clave cuántica.

Abstract — This dissertation presents a quantum key distillation system for the BB84 protocol. The purpose is to implement and test the system to be integrated into a complete solution of high performance quantum key

distribution. Furthermore, identify and propose improvements in the distillation process to enhance the bit rate of quantum key distribution.

Autor: Juan Vicente Pradilla Cerón, email: juaprace@teleco.upv.es
Director 1: José Mora Almerich Ph.D., email: jmalmer@iteam.upv.es
Director 2: José Capmany Francoy Ph.D., email: jcapmany@iteam.upv.es

Fecha de entrega: 15-07-13

ÍNDICE

I. Introducción

En la actualidad, la seguridad en las comunicaciones se perfila como uno de los temas de mayor relevancia que se están desarrollando en el mundo. Aplicaciones como las comunicaciones en centrales nucleares, las operaciones interbancarias y las redes de organismos de inteligencia militar requieren que se garantice la máxima seguridad para su operación diaria.

Para dar respuesta a estos requerimientos crecientes de seguridad han comenzado a emerger protocolos para generación y distribución de claves que utilizan los principios de la física cuántica (conocidos como *Quantum Key Distribution*, QKD) con el objetivo de garantizar una seguridad incondicional. Ejemplos de estos protocolos son: el BB84, que fue propuesto por Bennett y Brassard [1]; el B92, que es una modificación al protocolo BB84 propuesta en 1992 por Bennett [2], la cual no brinda grandes ventajas sobre su predecesor, por lo que su interés no va más allá del académico; y el SARG04, propuesto en el año 2004 por Scarani, Acín, Ribordy y Gisin [3], como una alternativa al protocolo BB84, para evitar el ataque por división del número de fotones, PNS.

Todos estos protocolos utilizan los principios de la física cuántica. En primer lugar, el principio de incertidumbre de Heisenberg, que asegura que es imposible determinar, con precisión absoluta y de forma simultánea, el valor de dos magnitudes conjugadas de un sistema elemental. Por otro lado, el teorema de la no clonación propuesto por Dieks, Wootters y Zurek, asegura que no se puede clonar de forma exacta un estado cuántico desconocido manteniendo el original sin modificaciones. Finalmente, estos protocolos también se basan en que las correlaciones cuánticas obtenidas de medidas separadas de estados entrelazados violan la desigualdad de Bell y por tanto impiden crear un acuerdo antes de la medida [4]. El hecho de que la seguridad esté basada en los principios de la física sugiere la posibilidad de seguridad incondicional que implica la posibilidad de demostrar que el sistema es seguro sin ninguna restricción en las capacidades que tenga un atacante, salvo los límites fijados por la física [1].

II. Protocolo BB84

II.1 *Fundamentos del protocolo BB84*

El protocolo BB84 para la distribución de clave cuántica, que ha sido el escogido para esta tesina, se modela como la interacción de tres actores: un emisor (Alice), un receptor (Bob) y un atacante (Eve). Para comunicarse Alice y Bob emplean dos canales de comunicación: uno cuántico, que les permitirá compartir señales cuánticas, y un canal clásico, en el cual pueden enviar mensajes de forma clásica [4].

El canal clásico necesita ser autentificado, lo que implica que Alice y Bob deben identificarse entre ellos directamente o a través de un tercero (entidad certificadora). Eve, por su parte, puede escuchar la conversación clásica, pero no participar en ella. Sin embargo, el canal cuántico está abierto a cualquier manipulación por parte de Eve. De forma que, la tarea de Alice y Bob es garantizar la seguridad, teniendo en cuenta el libre acceso de Eve para manipular el canal cuántico y escuchar la transmisión del canal clásico. La figura 1 muestra un esquema general de esta técnica.

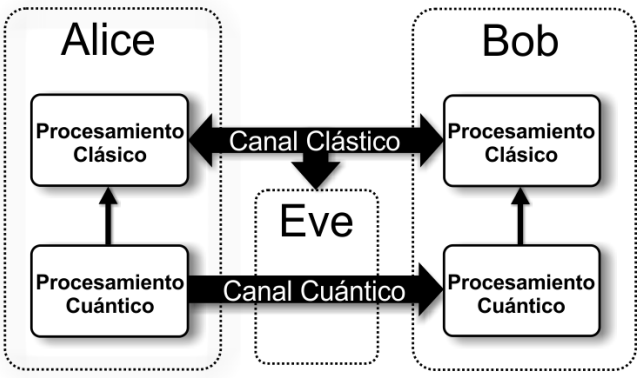

Fig. 1: Modelo de un protocolo de QKD

Sobre este modelo, el protocolo BB84 se puede dividir en dos partes, según el canal que se esté empleando. Estas dos partes de presentan en profundidad a continuación.

II.2 *Intercambio de la clave a través del canal cuántico*

En la primera parte del protocolo BB84 se emplea el canal cuántico, a través del cual, Alice envía a Bob un conjunto aleatorio de qbits (bits cuánticos) codificados según cuatro estados. Estos cuatro estados se agrupan formando dos bases con estados ortogonales: los dos primeros estados forman una primera base ($|\psi_{+0}\rangle$,$|\psi_{+1}\rangle$) y los otras dos ($|\psi_{X0}\rangle$,$|\psi_{X1}\rangle$) forman una segunda, logrando que las condiciones $\langle\psi_{+0}|\psi_{+1}\rangle = 0$ y $\langle\psi_{X0}|\psi_{X1}\rangle = 0$, correspondientes al producto escalar entre estados, sean satisfechas. Al mismo tiempo, los estados de diferentes bases de la tabla 1 no son ortogonales, ya que $\langle\psi_{+0,+1}|\psi_{X0,X1}\rangle \neq 0$. Así, un estado queda absolutamente determinado al proyectarlo sobre su correspondiente base, mientras que el resultado será totalmente aleatorio si se proyecta en la otra base.

Base	Estado	Descripción
$Base_+$	ψ_{+0}	Preparar el qubit en la base de polarización horizontal/vertical con el valor 0
	ψ_{+1}	Preparar el qubit en la base de polarización horizontal/vertical con el valor 1
$Base_x$	ψ_{X0}	Preparar el qubit en la base de polarización oblicua con el valor 0
	ψ_{X1}	Preparar el qubit en la base de polarización oblicua con el valor 1

Tabla. 1: Estados en función de la base y el valor asignado.

Por su parte, Bob determina el estado que Alice le ha enviado escogiendo aleatoriamente, para cada uno de los estados recibidos, una de las dos posibles bases de $\langle \psi_{+0,+1} | \psi_{X0,X1} \rangle \neq 0$, midiendo y almacenando el resultado. Es de esperar que Bob escoja la misma base que Alice en el 50% de los casos.

II.3 *Destilación de clave cuántica mediante el canal clásico*

Una vez que el envío de la clave finaliza por parte de Alice, comienza el proceso de destilación de clave que se lleva a cabo a través del canal clásico entre los dos extremos. Este proceso está compuesto por cuatro capas sucesivas representadas en la figura 2: Reconciliación de bases (*Sifted*), detección de errores (*Error Detection*), corrección de errores (*Error Correction*) y amplificación de la privacidad (*Privacy Amplification*).

En la capa de reconciliación de bases, Alice envía a Bob las bases que ha seleccionado para codificar cada uno de los qbits, pero sin revelar en ningún momento el estado seleccionado. De igual forma Bob comparte que bases ha empleado para realizar las medias en cada uno de los qbits que ha recibido, sin indicar el resultado de la medida. Alice y Bob retienen los qbits donde la base elegida por ambos coincide y descartan el resto. Como resultado obtienen una cadena de bits de longitud aproximada la mitad de la original, hecho que inmediatamente reduce a la mitad la tasa de bit.

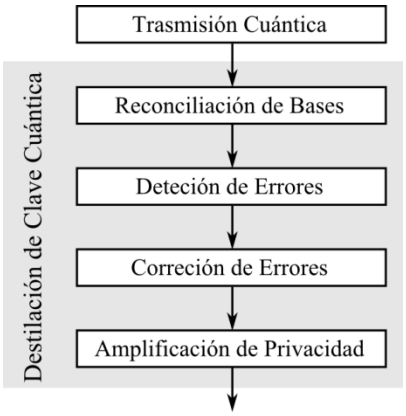

Fig.2: Capas del proceso de destilación de claves

La relación entre los qbits enviados y los errores detectados es conocida como la tasa de errores cuánticos (QBER, *Quantum Bit Error Rate*), magnitud que resultará imprescindible para decidir si se puede transmitir una clave con seguridad incondicional con un número acotado de bits conocidos por un hipotético espía. Esta cota se debe ajustar a los requerimientos de la aplicación que usara la clave y las características del canal.

Por su parte, la capa de detección de errores comparte una pequeña parte de la clave en crudo entre Alice y Bob para tratar de determinar la cantidad de errores que el canal cuántico ha introducido en la comunicación. Los bits compartidos son eliminados de la clave, de forma que la clave resultante junto con la estimación realizada se pasa la capa de corrección de errores para intentar mitigar los errores introducidos por el canal.

En la capa de corrección de errores se pueden emplear diversos algoritmos que corrijan el mayor número de errores y a su vez minimicen la cantidad de información que se revela mediante el canal clásico. Los algoritmos más utilizados incluyen el *Cascade* [5], el *Winnow* [6] y el uso de *LDPC* [7,8].

Para el caso del algoritmo *Cascade*, que es el más empleado en conjunción con el BB84, la clave se divide en bloques de igual tamaño. Se calcula la paridad a cada uno de los bloques y se intercambia la misma entre Alice y Bob. Si la paridad es la misma se asume que ese bloque no contiene errores, de lo contrario se realiza una búsqueda binaria del error en el bloque. La búsqueda binaria consiste en dividir el bloque en dos y para cada uno de los sub-bloques calcular la paridad y compartirla con el otro extremo de la comunicación. Si las paridades son iguales el bloque no se subdivide. De no ser así, el proceso se repite hasta encontrar el bit erróneo y se corrige.

Siguiendo con el algoritmo *Cascade*, después de corregir los errores en los bloques iniciales mediante la búsqueda binaria, se reorganizan de forma aleatoria los bits de la clave, y se divide la clave nuevamente para repetir el proceso completo. Se hacen máximo cuatro iteraciones aumentando al doble, en cada ocasión, el tamaño de los bloques iniciales. Se considera que el algoritmo ha corregido todos los errores si llega a las cuatro iteraciones o si el tamaño inicial de bloque supera al tamaño total de la clave.

Contando con una clave idéntica en Alice y Bob se pasa la clave a la capa de amplificación de la privacidad, en esta se emplea comúnmente una función hash para codificar los bits. Las funciones hash utilizadas en la amplificación de la privacidad son conocidas como hash criptográficas y deben cumplir los siguientes criterios [9]:

- Unidireccionalidad (Resistencia pre-imagen): dado un hash **H**, debe ser computacionalmente inviable encontrar un mensaje **M** tal que **H = hash(M)**
- Resistencia a colisiones: dado **M1** debe ser computacionalmente inviable encontrar **M2** tal que **H = hash(M1) = hash(M2)**

- Uniformidad: dado **H1 = hash(M1)** y **H2 = hash(M2)**; **H1** y **H2** deben diferir en aproximadamente el 50% de sus bits, cuando M2 es igual a M1 con un bit modificado
- Facilidad de cálculo: Debe ser fácil calcular **hash(M)** a partir de un mensaje M
- Resultado constante: para un mensaje **M** la función **hash(M)** debe siempre entregar la misma cantidad de bits (n), sin depender del tamaño de **M**

II.4. *QBER* y *Límite de seguridad*

El QBER (*Quantum Bit Error Rate)* es la relación entre los qbits enviados y los errores detectados. Este valor es de suma importancia porque determina el límite en el que la clave deja de ser segura. Para calcular el valor máximo de QBER admisible se realiza un análisis de la información mutua que se comparte entre las distintas entidades (Alice - A, Bob - B y Eve - E).

Se define la información mutua entre Alice y Bob como: $I(A, B) = H(A) - H(A|B) = H(B) - H(B|A)$ donde $H(x)$ es la entropía de Shannon y $H(x|y)$ es la entropía condicional. Y de forma análoga se define $I(A,E)$ e $I(B,E)$ como la información mutua entre Alice/Eve y Bob/Eve donde se puede deducir que $I(A,B) + I(A,E) \leq 1$.

Por otra parte, se conoce que para que la clave que se comparte entre Alice y Bo sea secreta debe cumplir con la condición: $I(A,B) \geq \text{máx}\{I(A,E), I(B,E)\}$. De los dos resultados anteriores se infiere que: $I(A,B) \geq 1/2$.

Expresándolo en términos del QBER (Q), es obtiene que $I(A,B) = 1 - Q\log_2(Q) - (1 - Q)\log_2(1 - Q) \geq 1/2 \rightarrow Q \leq 11\%$ [10]. Con lo que se llega a que el límite de seguridad o máximo QBER para que Alice y Bob compartan una clave segura es del 11%. En el análisis, se usa un 14% para dejar un margen de reserva.

III. Análisis del comportamiento del proceso de destilación de clave cuántica

III.1. *Metodología*

El análisis de comportamiento del proceso de destilación de la clave cuántica se lleva a cabo en dos fases: la ejecución de múltiples instancias del proceso con una tarea transversal de recolección de datos, mediante sondas en cada nivel de la arquitectura, y el análisis posterior de los datos mediante una herramienta estadística.

Para la primera fase, se definen cinco puntos de medición. En la Fig. 3 se muestra los puntos de medición para cada una de las capas del proceso de destilación, en los que se adapta una sonda de recolección de datos con el fin de almacenar: el tiempo empleado en el procesamiento, el tamaño de la clave y el *Bit Error Rate* (BER).

Para cada una de las instancias del proceso, se define un tamaño inicial de clave de longitud variable (128, 256 y 512 qbits) y un *Quantum Bit Error Rate* (QBER) inicial que varía entre 0% y 14% del tamaño de la clave en pasos de 0.1%. Para cada una de las configuraciones posibles se toman 10 muestras. De esta forma se recolectan un total de 61.500 muestras correspondiente a 12.300 instancias del proceso (12.300 destilaciones de clave cuántica - 4.100 claves de 512 qbits, 4.100 claves de 256 qbits y 4.100 claves de 128 qbits).

Finalmente, las muestras medidas por las sondas se almacenan en un archivo .CVS separado por punto y coma “;” para su posterior análisis en un programa estadístico.

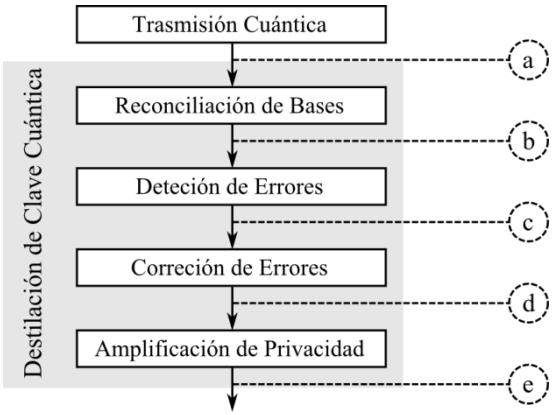

Fig.3. Puntos de medición en el proceso de destilación de clave cuántica.

En la segunda fase, se toman las muestras almacenadas en el archivo .CSV y se cargan en un programa de análisis estadístico. Posteriormente, se analizan estas muestras mediante scripts en lenguaje R, escritos especialmente para ente fin. De esta forma, se puede examinar el comportamiento del proceso de destilación en términos de: tamaño de la clave, BER y duración de procesamiento, en cada uno de los puntos de medición.

III.2. *Resultados*

La presentación de los resultados se apoya ampliamente en la presentación de graficas descriptivas y se divide en tres apartados: tamaño de la clave, BER y duración de procesamiento. Sobre esta división se obtendrá dos puntos de vista: consolidado y detallado. De esta forma, se podrá analizar el comportamiento del proceso de destilación de clave cuántica con una visión global y posteriormente pasar a conocer el comportamiento de cada una de las capas.

III.2.1 *Consolidados*

Las gráficas consolidadas para el tamaño de la clave se muestran en las figuras 4, 5 y 6. Estas figuras muestran la evolución del tamaño de la clave al ir siendo procesada por las diferentes capas para una clave inicial de 128, 256 y 512 qbits como se aprecia en la figura 4(a), 5(a) y 6(a), respectivamente.

En primer lugar, se observa que el tamaño de la clave disminuye cerca de un 50% en el proceso de reconciliación de bases (figuras 4(b), 5(b) y 6(b)). Por otra parte, se observa a través de las figuras 4(c), 5(c) y 6(c) que el proceso de detección de errores solo descarta una cantidad pequeña de la clave. La disminución de la clave de cerca de un 50% es un resultado esperable y asociado a la aleatoriedad con la cual se eligen las bases para la medición qbit a la entrada del proceso de destilación de clave cuántica, mientras que la pequeña reducción del proceso de corrección de errores se vincula directamente a la porción de la clave que se comparte para estimar el BER y que es descartada después de ser compartida.

De forma similar, se observa en las figuras 4(d), 5(d) y 6(d) que el comportamiento de la corrección de errores mantiene el tamaño de la clave. Esto se atribuye al hecho de que no se descartan bits durante el procesado. Es interesante recalcar que al no eliminar bits de paridad se aumenta la información disponible para el atacante. Sin embargo, este comportamiento es posteriormente corregido con la amplificación de la privacidad con lo cual no constituye un riesgo significativo a la integridad de la clave.

Finalmente, las figuras 4(e), 5(e) y 6(e) muestran el resultado más relevante de este análisis: el tamaño de la clave a la salida del proceso de destilación es independiente del tamaño de clave inicial y del QBER inicial. En este caso, se obtiene un valor de 512 para las tres longitudes de bits. La razón de este comportamiento radica en las funciones de amplificación de la privacidad que en este caso funcionan como una función hash que independientemente de la entrada entrega una cadena de 512 bits.

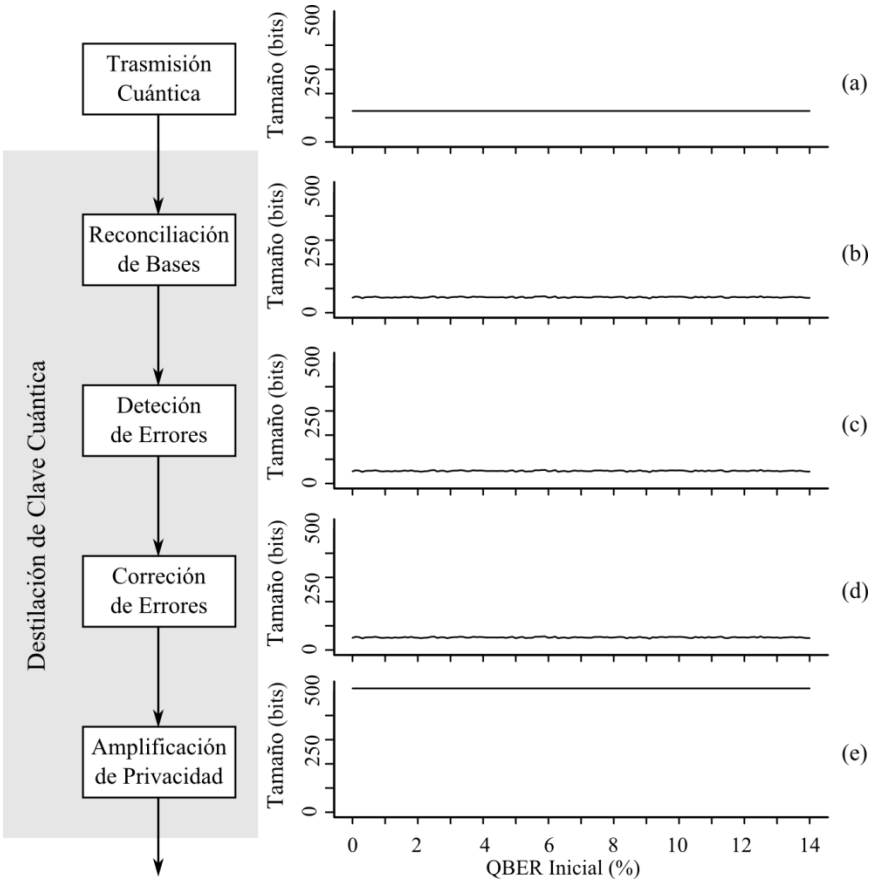

Fig.4. Evolución del tamaño de la clave en el proceso de destilación para una clave inicial de 128 qbits

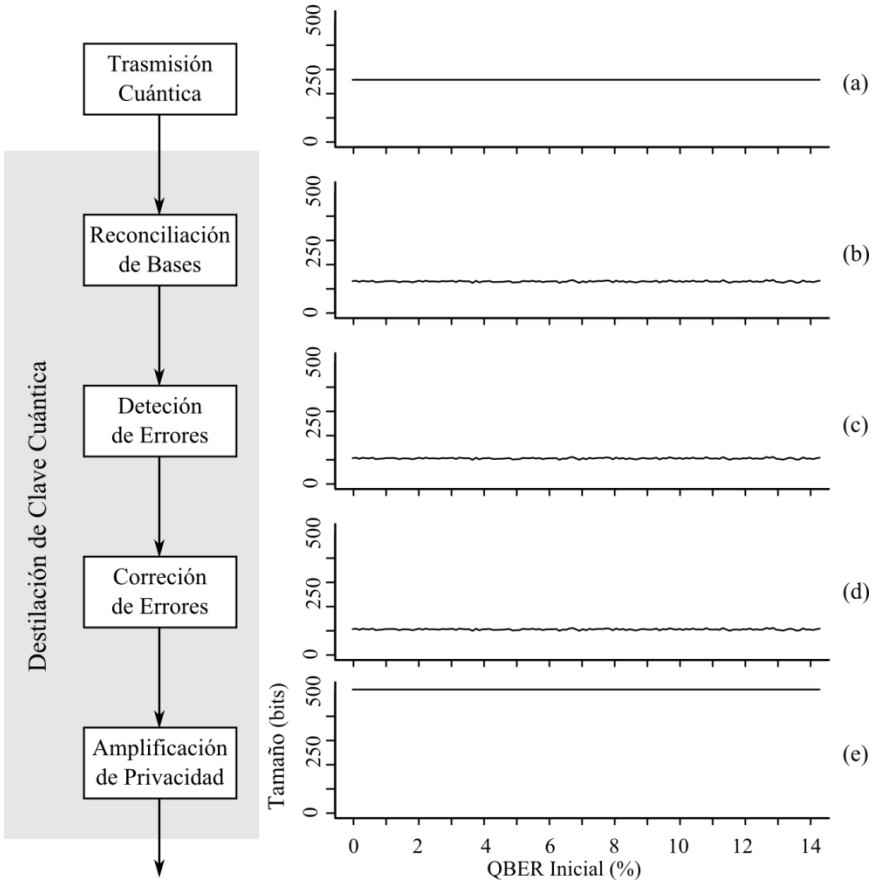

Fig.5. Evolución del tamaño de la clave en el proceso de destilación para una clave inicial de 256 qbits

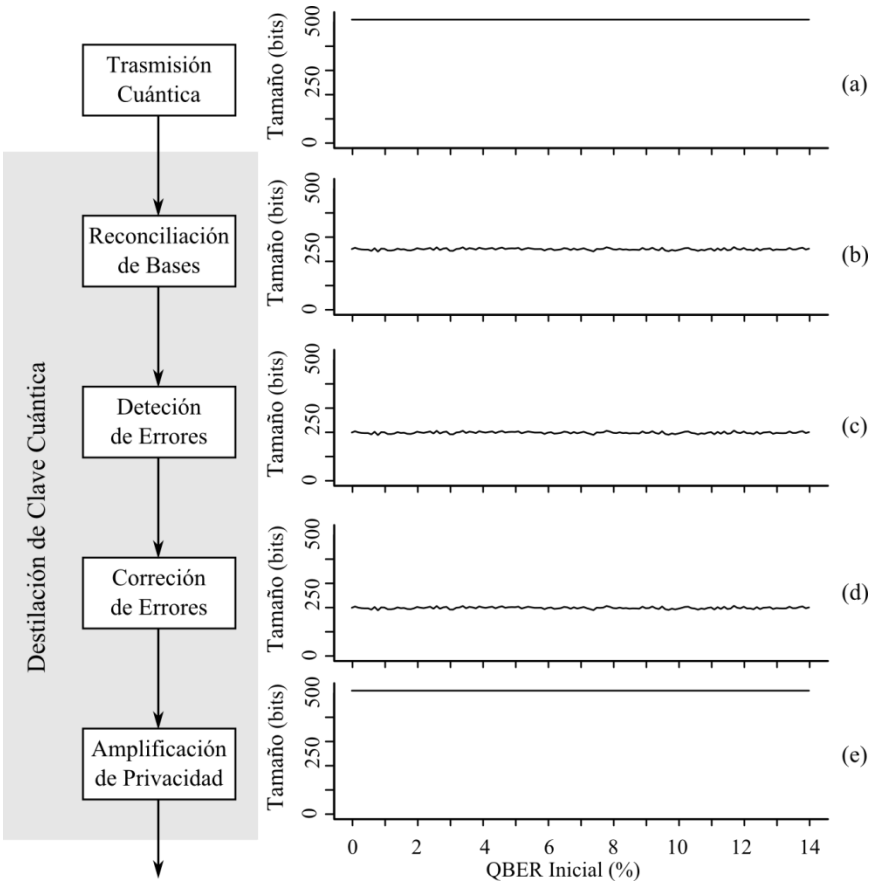

Fig.6. Evolución del tamaño de la clave en el proceso de destilación para una clave inicial de 512 qbits

Las figuras 7, 8 y 9 muestran el BER de la clave en cada una de las capas del proceso de destilación de clave cuántica para distintas longitudes de clave. En estas gráficas, se evidencia que el BER de las claves en la capa trasmisión cuántica se encuentra alrededor del 50%, lo que es de esperar debido a la elección aleatoria de las bases de medición (figuras 7(a), 8(a) y 9(a)).

Además, se encuentra que el BER de las capas de reconciliación de bases (figuras 7(b), 8(b) y 9(b)) y detección de errores (figuras 7(c), 8(c) y 9(c)) se corresponde con el porcentaje del QBER inicial que se ha insertado en cada clave transmitida, evidenciando así que el proceso de reconciliación de bases ha funcionado de forma correcta y ha logrado quedarse con los fragmentos de la clave comunes a las dos entidades. La similitud entre estas dos capas radica en que el fragmento que se intercambia para la detección de errores es escogido de forma aleatoria, lo que permite que el BER promediado no se vea afectado significativamente.

Por su parte, la capa de corrección de errores (figuras 7(d), 8(d) y 9(d)) presenta el comportamiento más interesante al reducir de forma drástica el BER, llevándolo en casi todos los casos a cero, lo que implica que las claves son idénticas en las dos entidades (Alice y Bob) y por tanto, pueden ser empleadas para cifrar la información que se vaya a compartir entre los dos. Los pequeños picos que se observan en la gráfica son muestra de que algunas claves no fueron

corregidas totalmente, lo que revela que hay espacio para la mejora en la implementación de este protocolo.

Finalmente, la capa de amplificación de la privacidad, mantiene el BER en cero cuando se ha podido corregir todos los errores en los dos extremos de la comunicación pero amplifica los errores en las claves que mantienen un ver bajo. Este efecto es beneficioso para aumentar la seguridad de las claves cuando partes de esta han podido ser intervenidas por un atacante. En el análisis de este comportamiento se profundizará más adelante.

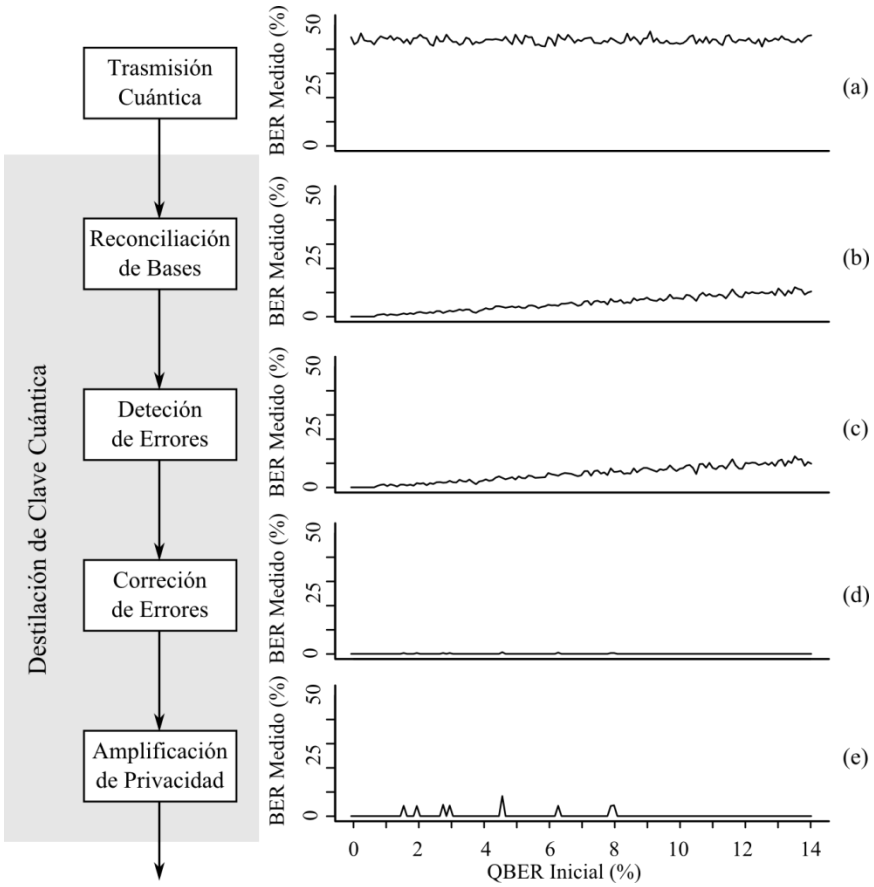

Fig.7. Evolución del BER en el proceso de destilación para una clave inicial de 128 qbits

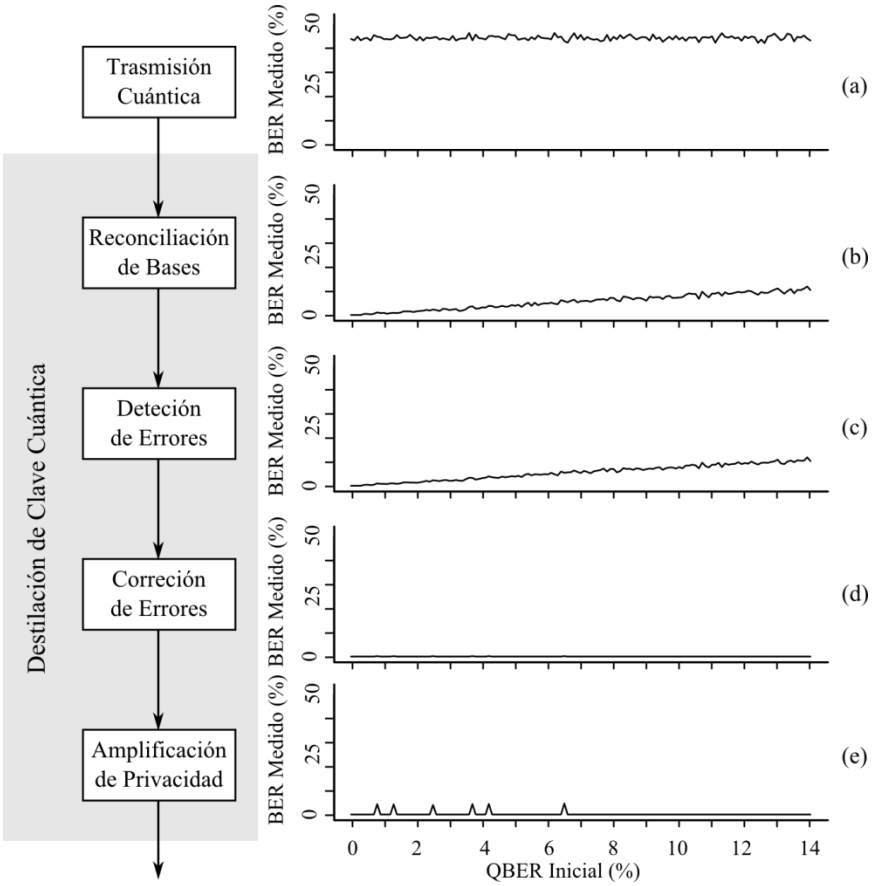

Fig.8. Evolución del BER en el proceso de destilación para una clave inicial de 256 qbits

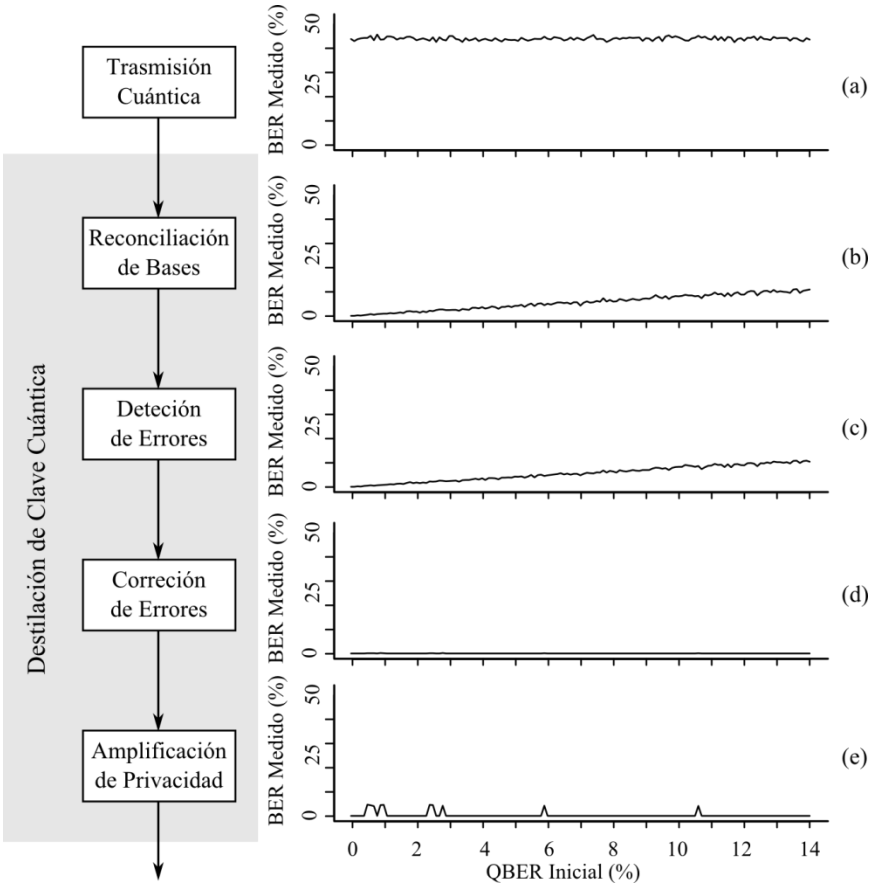

Fig.9. Evolución del BER en el proceso de destilación para una clave inicial de 512 qbits

Finalmente, las figuras 10, 11 y 12 muestra la duración del procesamiento en cada uno de los puntos de medición, siendo lo más interesante conocer que el proceso de detección de errores es el que tiene un mayor consumo de tiempo (punto de medición *c* en las figuras). Es necesario aclarar que en este punto de las pruebas se ejecutan las dos entidades (Alice y Bob) en un solo ordenador, con lo que el tiempo de transmisión no está contabilizado. Se observa que en proceso que utiliza más recursos computacionales es el de detección de errores que emplea 40 ms. También, resulta de interés el hecho de no evidenciar variaciones significativas en el tiempo de procesamiento con los tamaños de clave que se han utilizado. Sin embargo, este comportamiento seguramente se verá alterado cuando se realicen pruebas con claves de mayor tamaño y en el momento en el que el tiempo de transmisión se contabilice.

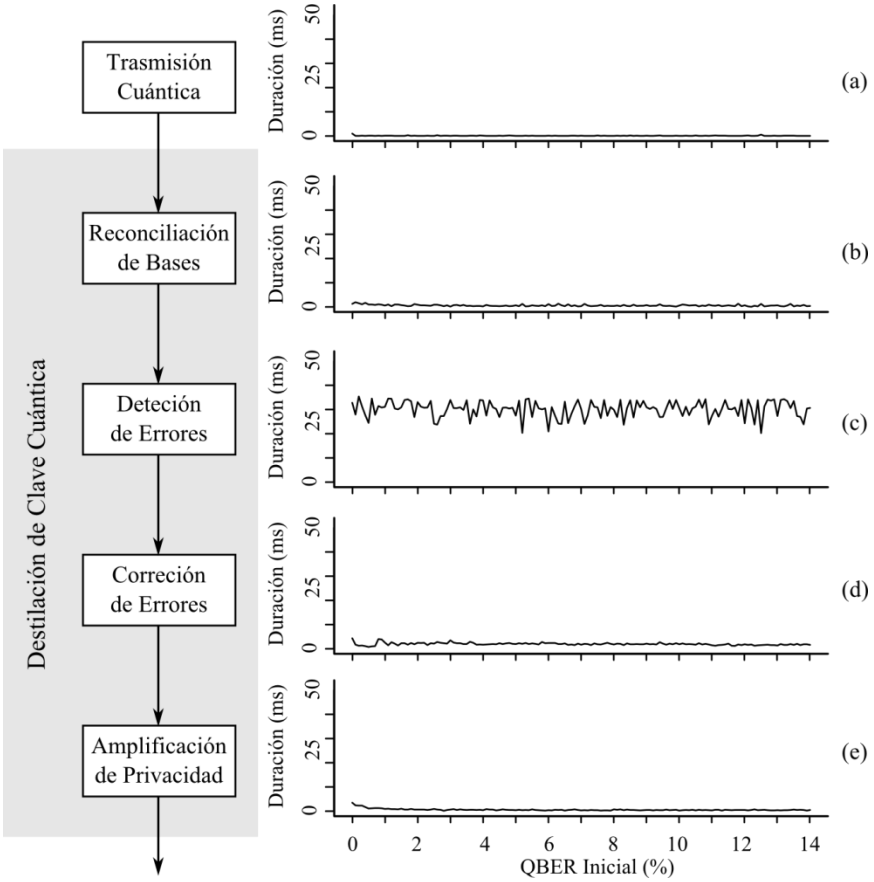

Fig.10. Evolución de la duración en el proceso de destilación para una clave inicial de 128 qbits

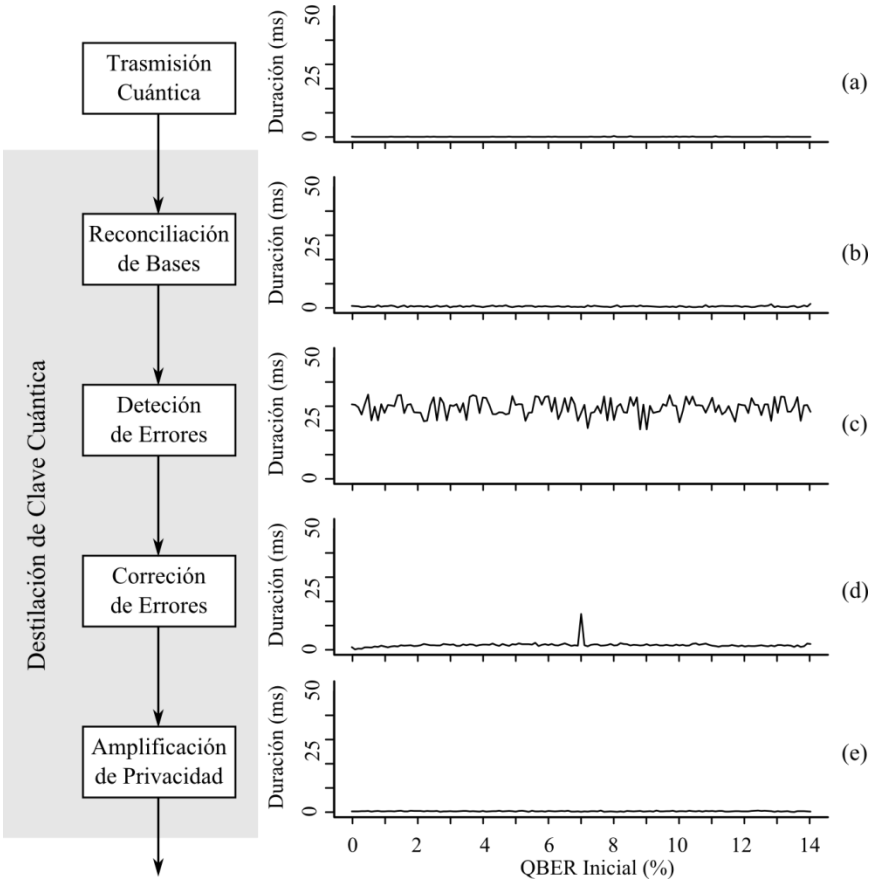

Fig.11. Evolución de la duración en el proceso de destilación para una clave inicial de 256 qbits

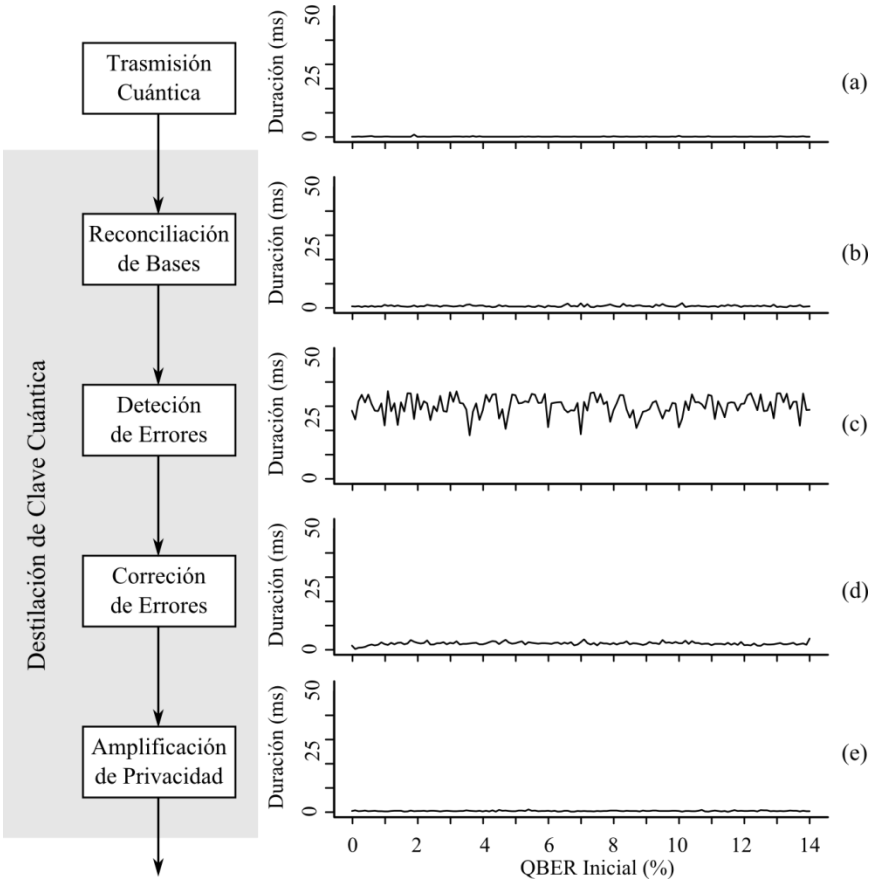

Fig.12. Evolución de la duración en el proceso de destilación para una clave inicial de 512 qbits

III.2.2 *Detallados*

A continuación se examinan los tres resultados: tamaño de la clave, BER y duración de procesamiento, para cada una de las cinco capas del proceso: transmisión cuántica (*a*), reconciliación de bases (*b*), detección de errores (*c*), corrección de errores (*d*) y amplificación de la privacidad (*e*).

En el caso de la capa de transmisión cuántica (*a*) la figura 13 muestra el tamaño de la clave para claves iniciales de 128, 256 y 512. Se puede observar que se recibe una longitud de clave de forma correcta según el tamaño asociado. Por su parte la figura 14, ilustra el BER en la capa y que es acorde al esperado (cercano al 50%) según la escogencia aleatoria de las bases de medición. De igual forma, la figura 15 muestra que la duración del procesamiento en esta capa se encuentra en su mayoría por debajo de los 0.2 ms y que el procesamiento de las primeras claves lleva más tiempo. Una posible explicación para este comportamiento es que en la primera ejecución se deben reservar los recursos en memoria y arrancar la máquina virtual de JAVA (lenguaje de programación empleado para el desarrollo) mientras que las ejecuciones posteriores aprovechan que estos recursos ya están cargados.

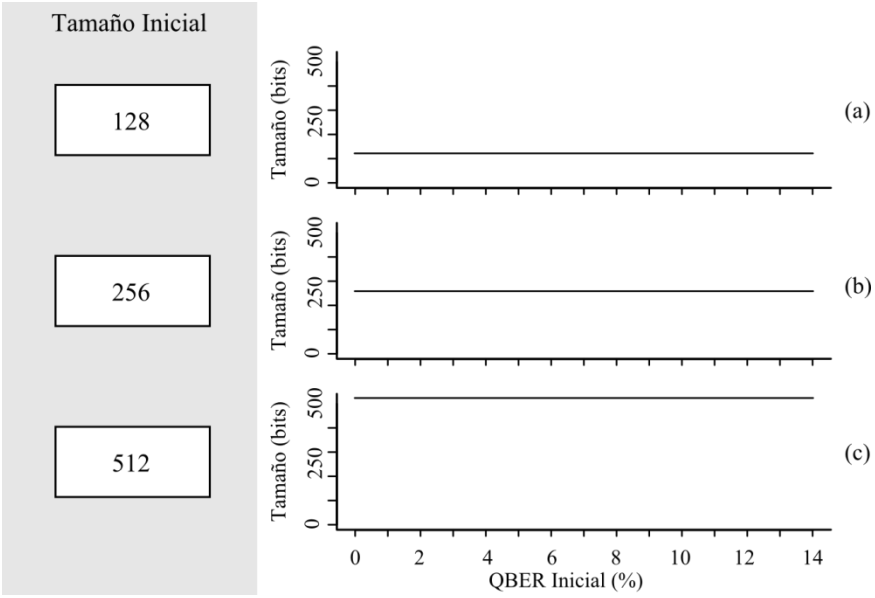

Fig.13. Tamaño de la clave en la capa de transporte cuántico para claves de 128(a), 256(b) y 512(c) qbits

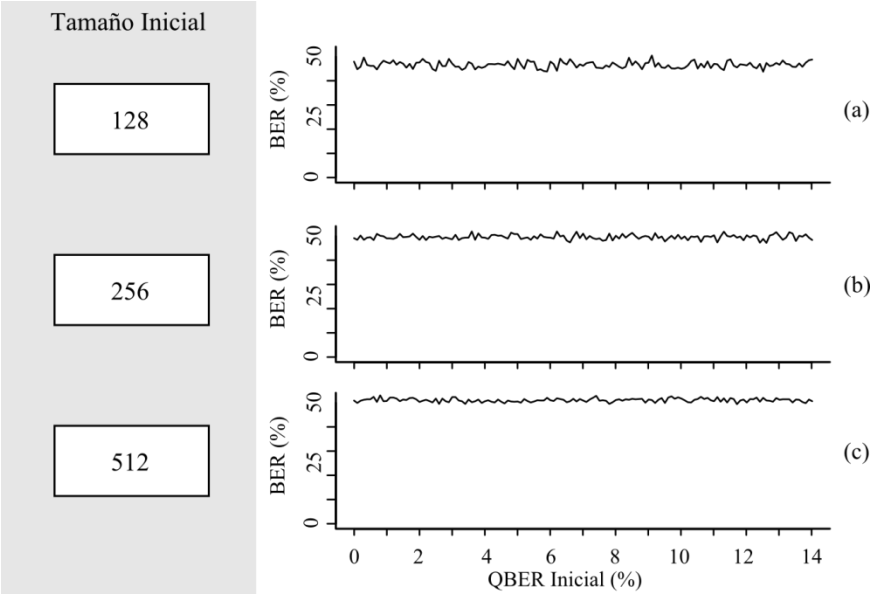

Fig.14. BER de la clave en la capa de transporte cuántico para claves de 128(a), 256(b) y 512(c) qbits

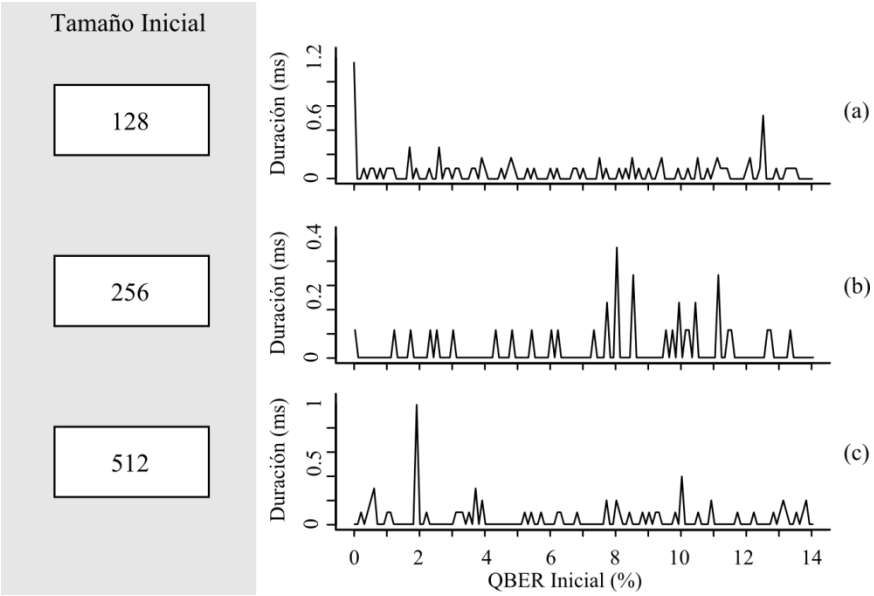

Fig.15. Duración de la clave en la capa de transporte cuántico para claves de 128, 256 y 512 qbits

De forma análoga, en la capa de reconciliación de bases se puede observar el tamaño de la clave para claves iniciales de 128, 256 y 512 (figura 16) es cercana al 50% de la clave inicial. Este hecho se atribuye al descarte de los qbits que fueron medidos con bases diferentes en Alice y Bob. Este efecto se encuentra estrechamente relacionado con la aleatoriedad de la elección de las bases. Por su parte, el BER de la figura 17 muestra una relación directa con el BER de la clave inicial, indicando de esta forma que el proceso de reconciliación de bases funciona de forma correcta al descartar los qbits medidos con bases distintas y conservando la distribución de los errores. Asimismo, se observa en la figura 18 que la duración de procesamiento en esta capa es inferior a los 2 ms y que al igual que en la capa anterior presenta un pequeño transitorio al destilar las primeras claves cuánticas.

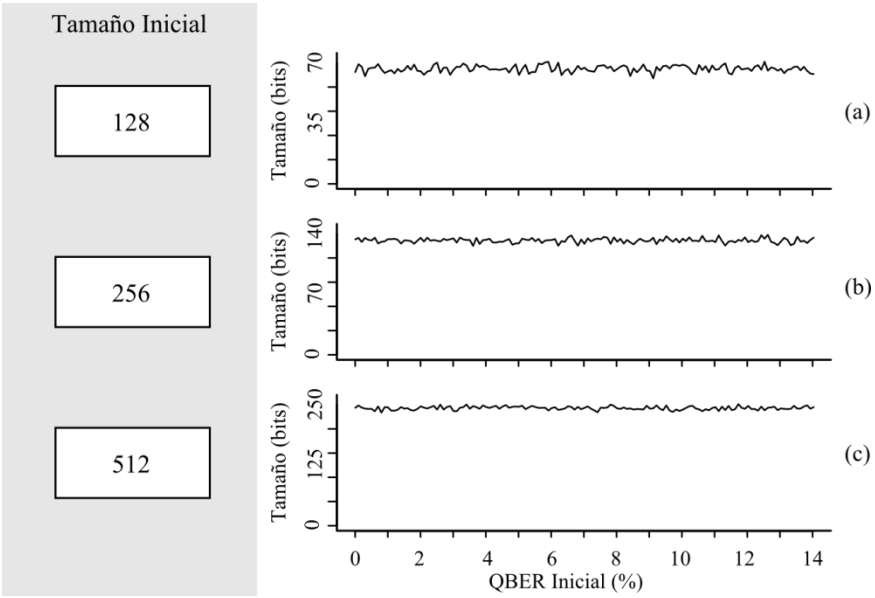

Fig.16. Tamaño de la clave en la capa de reconciliación de bases para claves de 128(a), 256(b) y 512(c) qbits

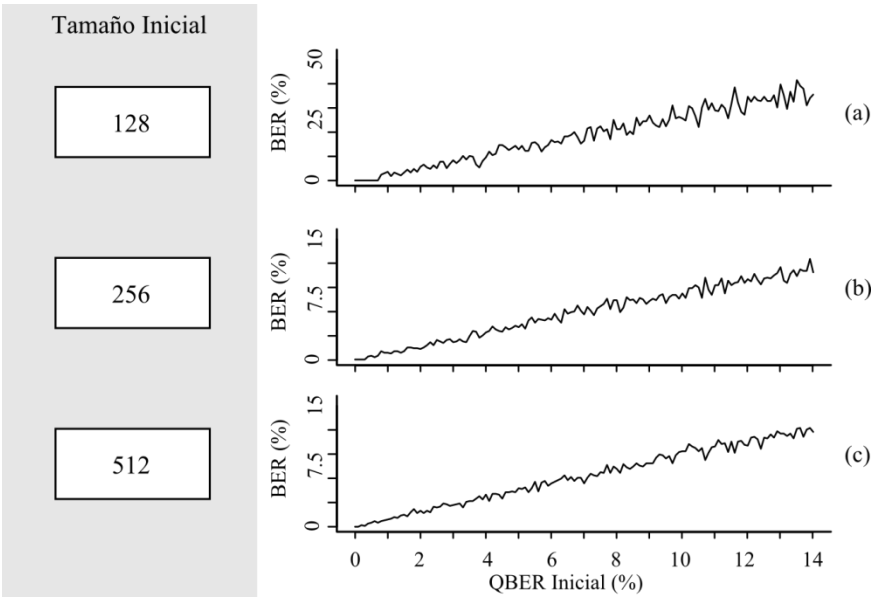

Fig.17. BER de la clave en la capa de reconciliación de bases para claves de 128(a), 256(b) y 512(c) qbits

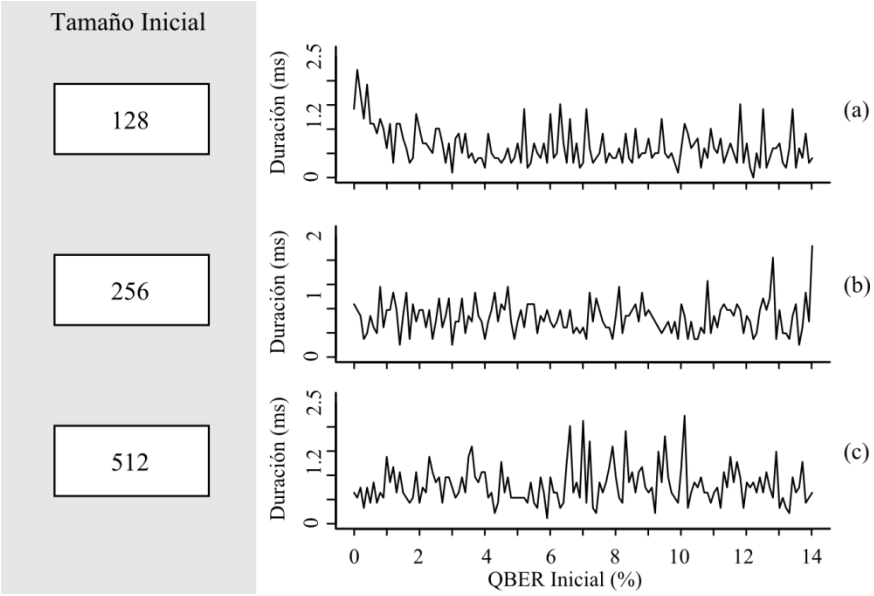

Fig.18. Duración de la clave en la capa de reconciliación de bases para claves de 128(a), 256(b) y 512(c) qbits

Por otra parte, la capa de detección de errores, mantiene un comportamiento muy similar a la capa de reconciliación de bases. El tamaño, representado en la figura 19, es un poco inferior debido a que un fragmento aleatorio de la clave es compartido mediante el canal clásico y por tanto debe ser descartado. Sin embargo, el tamaño de la clave en este punto es cercano al 40% de la clave inicial. De igual forma el BER permanece casi intacto (la elección de los qbits a compartir se hace de forma aleatoria) y se corresponde con el BER inicial (figura 20). Respecto la duración del procesamiento en esta capa (figura 21), representa la mayor duración de todo el proceso de destilación de clave cuántica y es cercana a los 40 ms. En versiones posteriores de la implementación se espera mejorar este comportamiento.

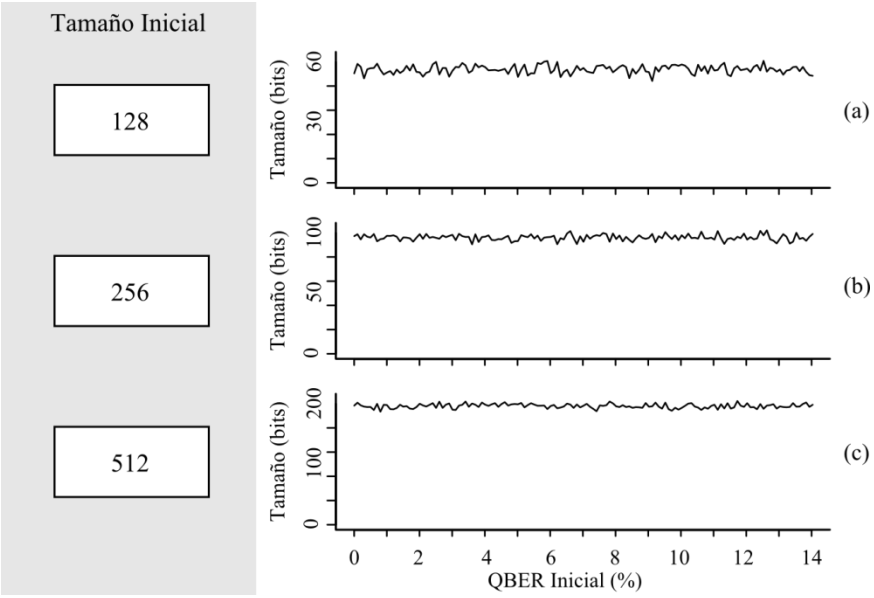

Fig.19. Tamaño de la clave en la capa de detección de errores para claves de 128(a), 256(b) y 512(c) qbits

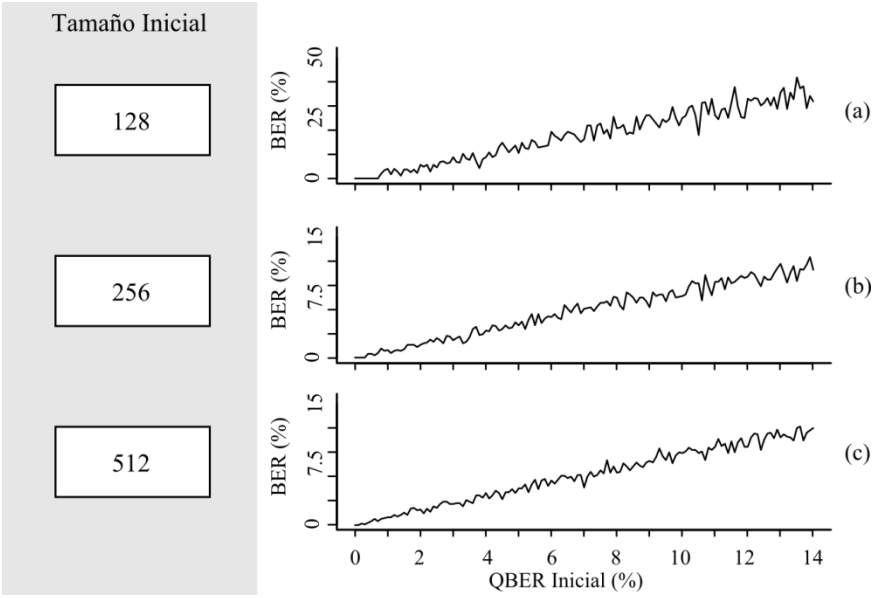

Fig.20. BER de la clave en la capa de detección de errores para claves de 128(a), 256(b) y 512(c) qbits

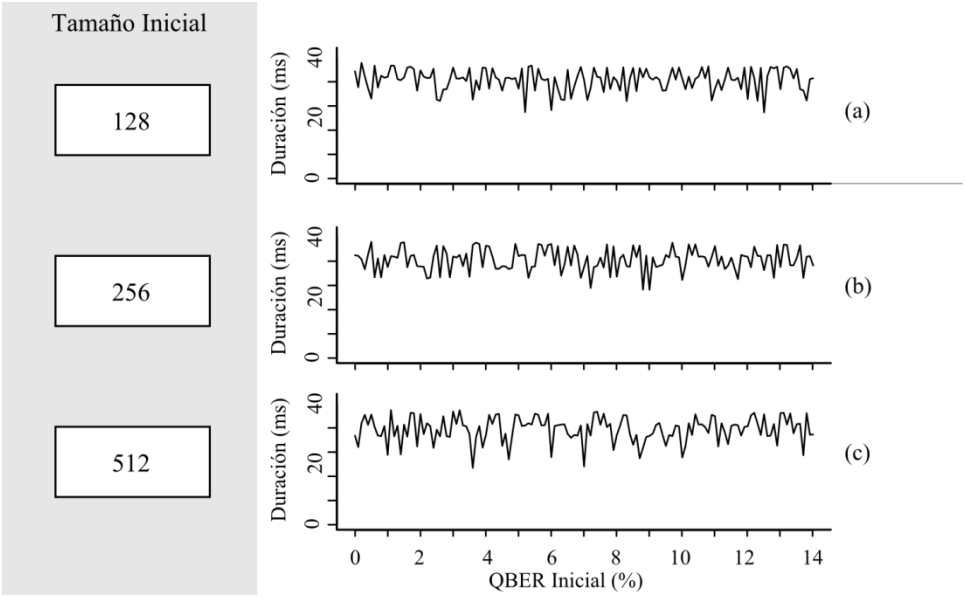

Fig.21. Duración de la clave en la capa cuántica para claves de 128(a), 256(b) y 512(c) qbits

Para la capa de corrección de errores se observa (figura 22) que el tamaño de la clave permanece igual que en la capa predecesora debido a que no se descartan los bits en el proceso. El BER presenta una disminución significativa (figura 23) llevándolo en la mayoría de los casos a cero. Es necesario recordar en este punto que para que la clave sea útil para ser utilizada el BER debe ser muy próximo a cero de lo contrario la información cifrada no podrá ser descifrada en el otro extremo. Al observar con detenimiento el resultado del ver, se confirma que el algoritmo de corrección corrige la mayoría de las claves y en el peor de los casos el BER presente en la clave se

reduce a menos del 1%. Además, se observa que la duración del proceso (figura 24) suele ser de menos de 5 ms, aunque es un proceso de uso intensivo de recursos.

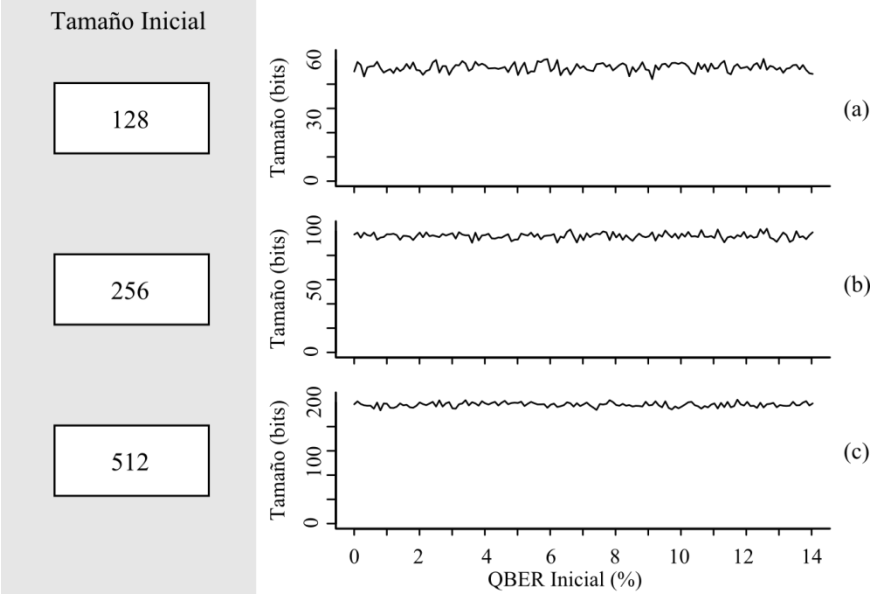

Fig.22. Tamaño de la clave en la capa de corrección de errores para claves de 128(a), 256(b) y 512(c) qbits

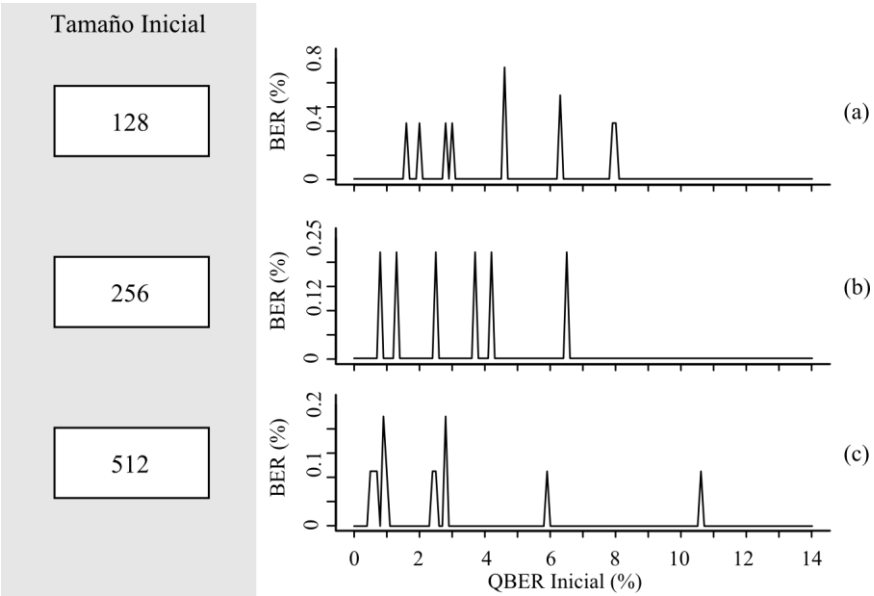

Fig.23. BER de la clave en la capa de corrección de errores para claves de 128(a), 256(b) y 512(c) qbits

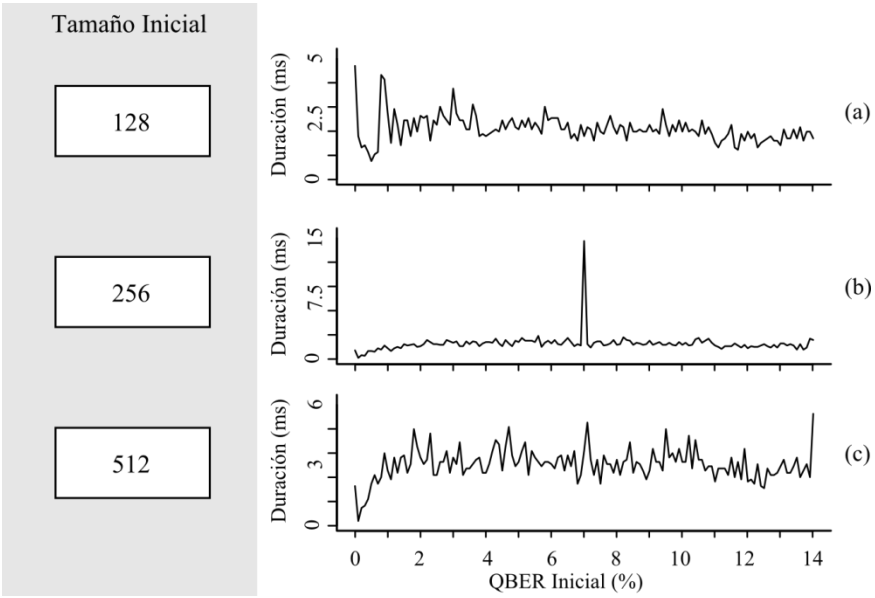

Fig.24. Duración de la clave en la capa de corrección de errores para claves de 128(a), 256(b) y 512(c) qbits

Finalmente, en la capa de amplificación de la privacidad se observa un tamaño constante de 512 bits (figura 25) como efecto directo de emplear una función hash SHA-512 que entrega una cadena de 512 bits ante una entrada de cualquier longitud de bits. La influencia en la integridad de la clave de este efecto se profundizará posteriormente. De igual forma, el BER en esta capa (figura 26), en su mayoría es cero dado que ya en el proceso de corrección de errores se entregan claves con un BER muy próximo a cero. En el caso en que el BER entregado por la capa anterior sea mayor que cero, el mismo, se ve amplificado hasta un 50% aproximadamente (es necesario considerar que el BER máximo entregado por la capa anterior es inferior al 1%). Finalmente, la duración en esta capa es aproximadamente de 0.5 ms (figura 27), lo que suele ser natural debido a que las funciones hash suelen estar muy optimizadas.

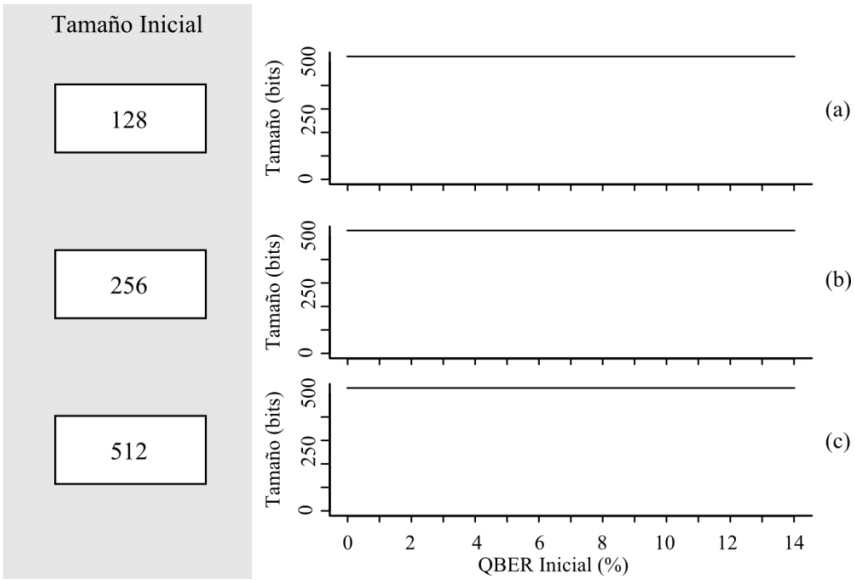

Fig.25. Tamaño de la clave en la capa de amplificación de la privacidad para claves de 128(a), 256(b) y 512(c) qbits

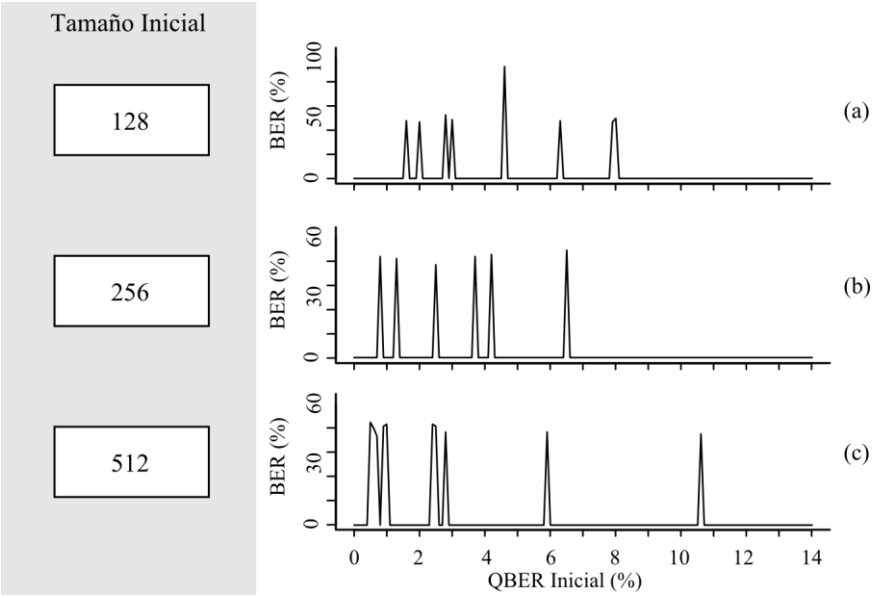

Fig.26. BER de la clave en la capa de amplificación de la privacidad para claves de 128(a), 256(b) y 512(c) qbits

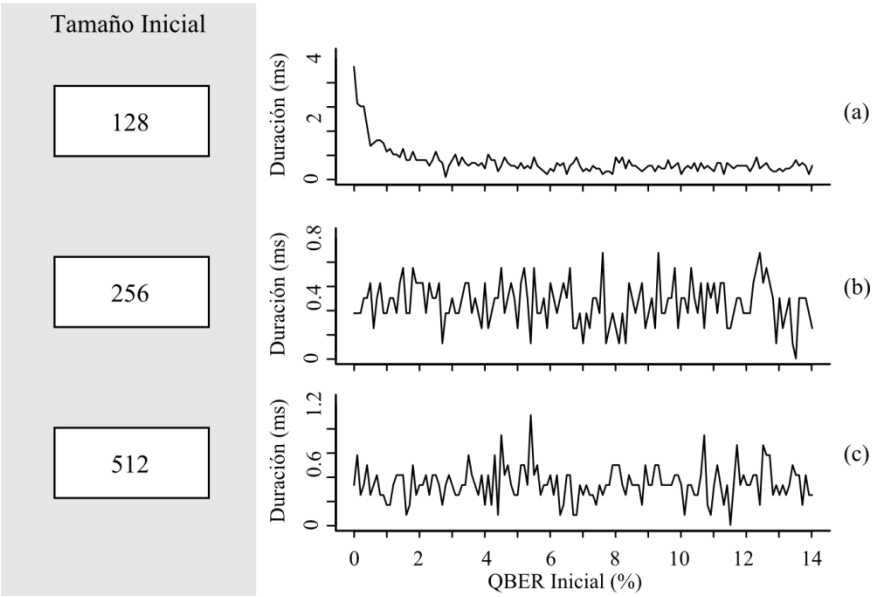

Fig.27. Duración de la clave en la capa de amplificación de la privacidad para claves de 128(a), 256(b) y 512(c) qbits

III.3. *El efecto de la amplificación de la privacidad en la integridad de la clave*

Para observar el efecto de la amplificación de la privacidad en la integridad de la clave se planteó la realización de seis experimentos que compararan el BER entre la clave que Bob obtiene mediante el proceso de destilación de clave cuántica y una clave hipotética obtenida por el atacante, Eva, que solo difiere en un porcentaje de BER inicial de la que ha obtenido Bob.

Así, el primer experimento, que sirve como punto de comparación con los otros cuatro, toma una clave antes de la amplificación de la privacidad (resultado intermedio del proceso de destilación de clave cuántica) de 512 bits para Bob. Por su parte, la clave de Eva se obtiene al duplicar la clave de Bob y adicionar de forma aleatoria un BER que varía entre el 1% y el 14% en pasos de 0.1%. De esta forma, se emula la posibilidad de que el atacante pueda acceder a la clave de Bob y que esta solo difiera en un porcentaje pequeño (entre 1% y 14%). Posteriormente, se aplica el proceso de amplificación de la privacidad a las dos claves (la de Eva y la de Bob) con una función hash que entrega una clave de 512 bits (de igual tamaño que a la entrada) y se comparan las dos claves obtenidas para establecer el BER entre estas dos claves.

El resultado de este primer experimento se representa en la figura 28, donde se puede observar las medidas obtenidas para los diferentes niveles de QBER inicial, junto con la media de todas las medidas. Es evidente, que la función hash entrega una clave uniformemente distribuida, lo que deriva en un BER entre las dos claves de cerca del 50%.

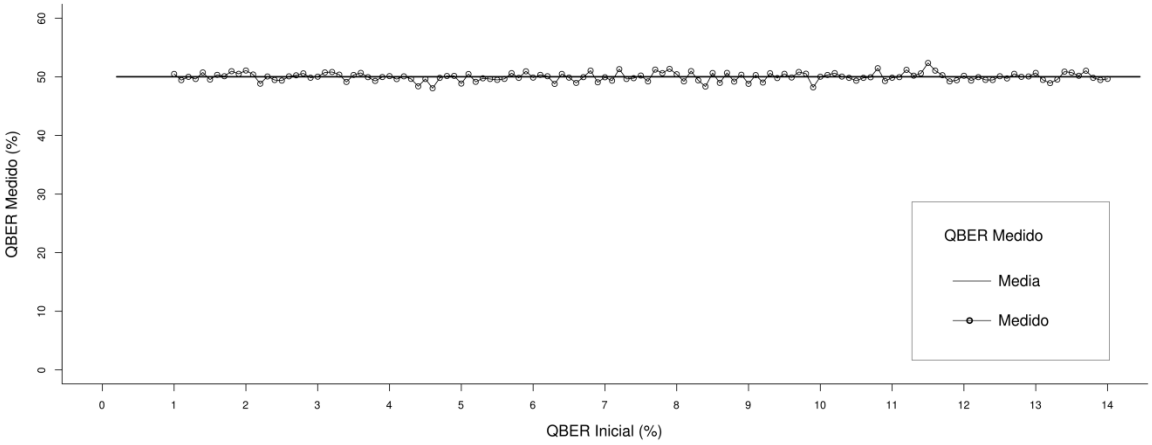

Fig.28. Efecto de la amplificación de privacidad para una clave de 512 bits con un hash de 512 bits

El segundo experimento, sigue la metodología empleada para el primero pero en esta ocasión se emplea una clave inicial para Bob y Eva de 128 bits y una función para la amplificación de la privacidad que entrega una clave final de 128 bits. Este experimento es también un punto de comparación para los experimentos posteriores. Los resultados del mismo, se observan en la figura 29, donde de forma análoga al anterior se encuentra que el BER entre las dos claves finales es cercano al 50%.

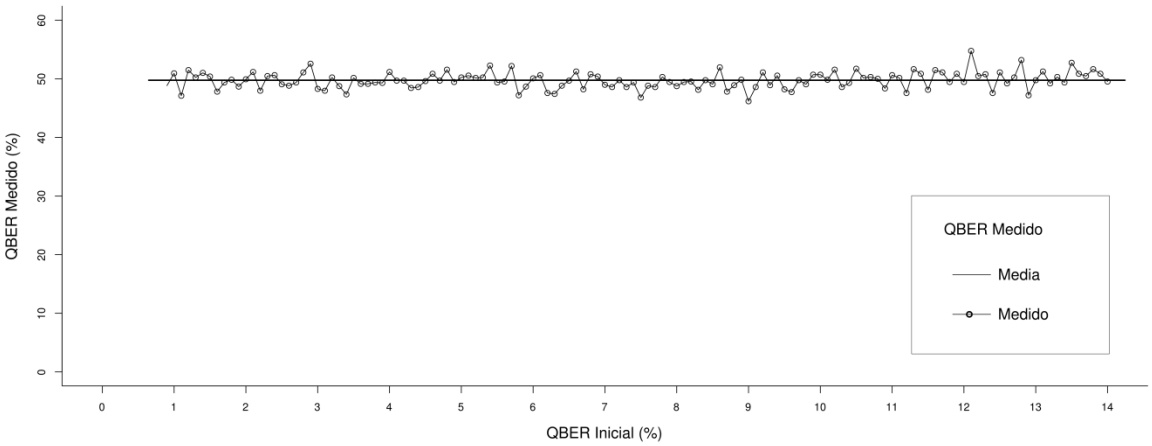

Fig.29. Efecto de la amplificación de privacidad para una clave de 128 bits con un hash de 128 bits

Los dos primeros experimentos sirven como referencia en términos de la integridad de las claves y muestra que es indiferente si se generan claves de 128 o 512 bis; es decir, se puede escoger transmitir claves de128 o 512 bits según los requerimientos de la aplicación sin tener implicaciones perjudiciales a nivel de integridad de la clave (en los dos casos es una clave segura).

En contraposición al experimento anterior, el tercer experimento toma cuatro claves de 128 bits, a cada una le aplica una función hash de 128 bits y luego se concatena las cuatro claves finales formando una sola clave de 512 bits. Posteriormente se comparan las claves de 512 bits obtenidas de esta forma en Bob y Eva, luego se grafica el BER entre estas dos claves. El resultado se muestra en la figura 30 y de nuevo se obtiene que el BER es uniforme y cercano al 50%. Con este experimento se demuestra que es posible obtener una clave segura de mayor longitud concatenando claves más pequeñas.

Por su parte, el cuarto experimento toma una ruta diferente. Inicia con una clave de 512 bits a la cual se le inyecta el QBER inicial. Luego la clave se divide en cuatro claves de 128 bits y se calcula un hash de 128 bits, para posteriormente concatenar las cuatro claves resultantes en una sola clave de 512 bits. Como se muestra en la figura 31 la seguridad de la clave se ve comprometida cuando el BER inicial es pequeño pero tiene un buen comportamiento (BER cercano al 50%) a niveles de BER mayores a 2%. Este efecto puede ser atribuido a que en presencia de pocos errores estos pueden ser localizados en solo una de las porciones de 128 bits con lo que al aplicar la función hash sobre los otros fragmentos estos serán idénticos para Bob y para Eva.

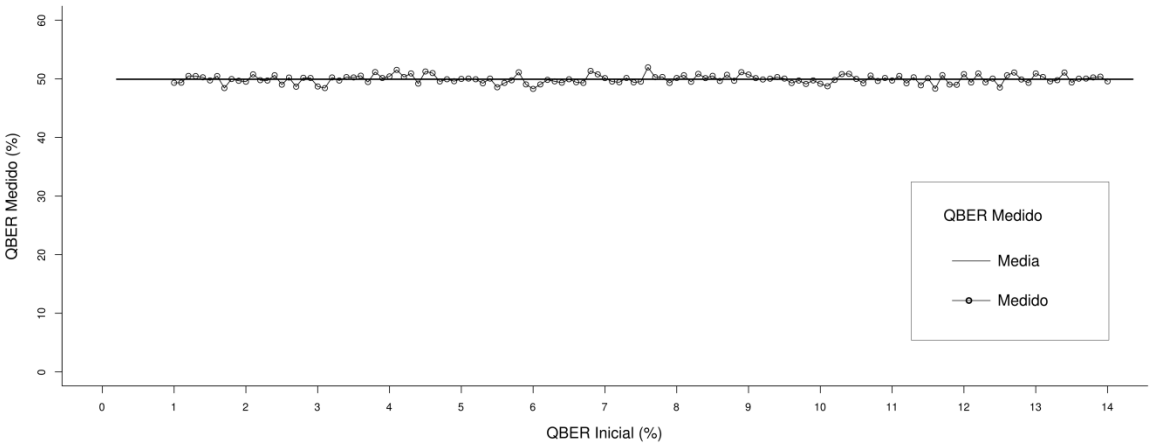

Fig.30. Efecto de la amplificación de privacidad para cuatro claves de 128 bits con un hash de 128 bits concatenadas en una sola clave de 512

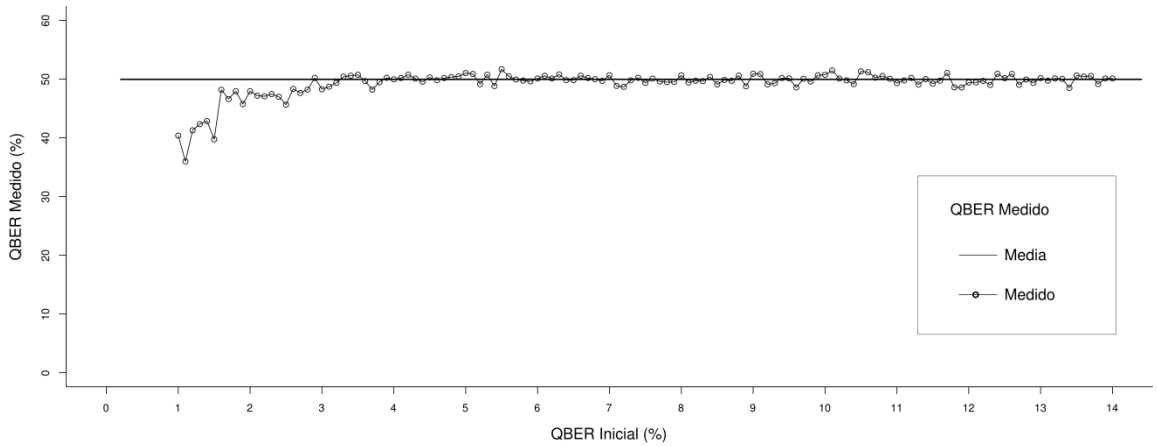

Fig.31. Efecto de la amplificación de privacidad para una clave de 512 bits dividida en cuatro claves de 128 con un hash de 128 bits concatenadas en una sola clave de 512

De forma análoga, el quinto experimento busca aprovechar los resultados anteriores para aumentar el tamaño de la clave final sin perder integridad en la clave generada. Para este experimento se toma una clave inicial de 512 bits a la que se le inyecta el QBER inicial y posteriormente se divide en 4. Luego se emplea una función hash que genera una clave de 512 bits para cada una de las cuatro partes. Finalmente, las cuatro partes se concatenan generando una clave final de 2048 bits (4 veces la clave inicial), al comparar este el resultado generado en Bob y Eva se observa el BER graficado en la figura 32. En esta figura se aprecia que el BER final se encuentra un poco por debajo del 50% para todos los casos en que el BER inicial es superior al 2% de la clave. Sin embargo, hay una perdida en la seguridad de la clave para valores de BER inicial inferiores al 2%. Esta técnica puede resultar interesante al tener en cuenta que un atacante que decida escuchar el canal cuántico, necesariamente introducirá en él un BER elevado.

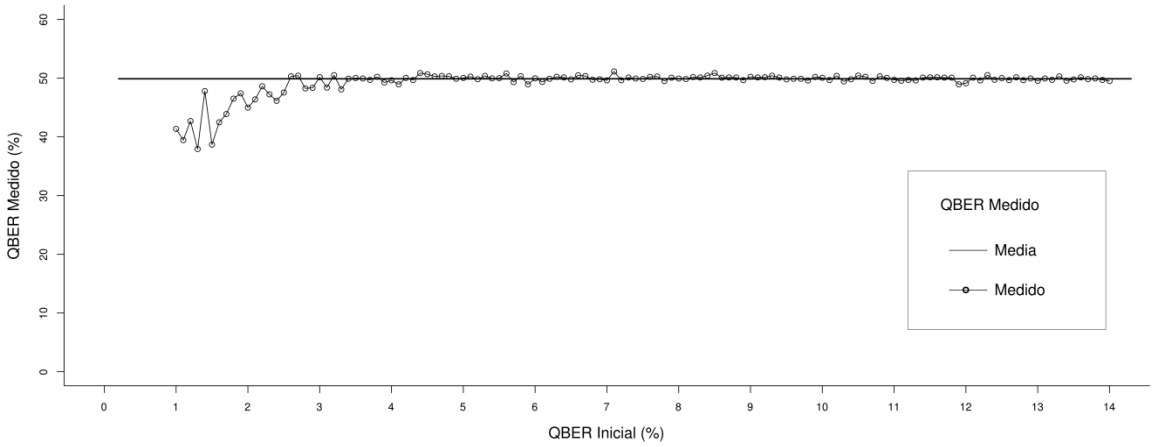

Fig.32. Efecto de la amplificación de privacidad para una clave de 512 bits dividida en cuatro claves de 128 con un hash de 512 bits concatenadas en una sola clave de 2048

Finalmente, el sexto experimento sigue la línea del experimento anterior. La diferencia radica en que en lugar de tomar una sola clave de 512 bits y dividirla, se inicia con cuatro claves de 128 bits, donde cada a una se le ha inyectado el QBER inicial. Luego se pasa cada clave por una función hash que entrega 512 bits, para finalmente, concatenarla en una única clave de 2048 bits (4 veces más grande que la suma de las claves iníciales). Como se observa en la figura 33, esta configuración permite obtener una ganancia en el tamaño final de la clave, manteniendo la seguridad de la clave.

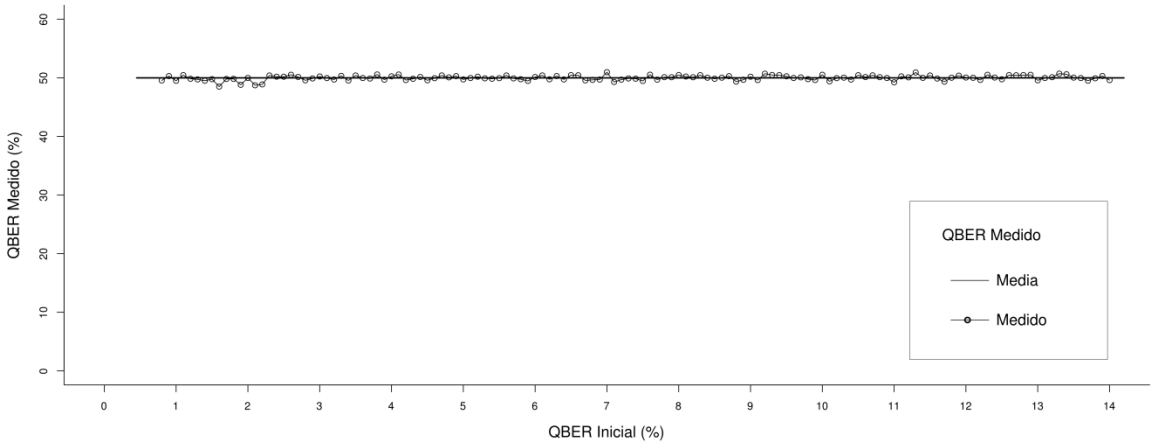

Fig.33. Efecto de la amplificación de privacidad para cuatro claves de 128 bits con un hash de 512 bits concatenadas en una sola clave de 2048

IV. Aplicaciones de prueba

Para explorar la aplicabilidad de los sistemas de distribución de clave cuántica se crearon dos aplicaciones de prueba: la primera es un sistema que crea un flujo de datos encriptado bajo un modelo cliente/servidor y la segunda es la creación de una red privada virtual (VPN - Virtual Private Network) punto a punto sobre IPSec [11] (Internet Protocol security).

La aplicación cliente/servidor crea un flujo de datos mediante un flujo de datos en Java sobre el cual se envían mensajes basados en UTF-8: mensajes (texto plano), archivos de configuración (.xml), páginas web (.html) y archivos de texto (.txt). El flujo funciona a nivel de aplicación y usa una encriptación Triple DES [12] (Triple Data Encryption Algorithm) empleando una clave intercambiada con anterioridad mediante el sistema QKD entre el cliente y el servidor. Así, para cada comunicación que el cliente realiza, primero se obtiene una clave compartida con el servidor mediante QKD, posteriormente se establece un flujo de datos cifrados usando la clave compartida, acto seguido se envía el mensaje mediante el flujo y finalmente se cierra el flujo y se descarta la clave empleada (figura 34).

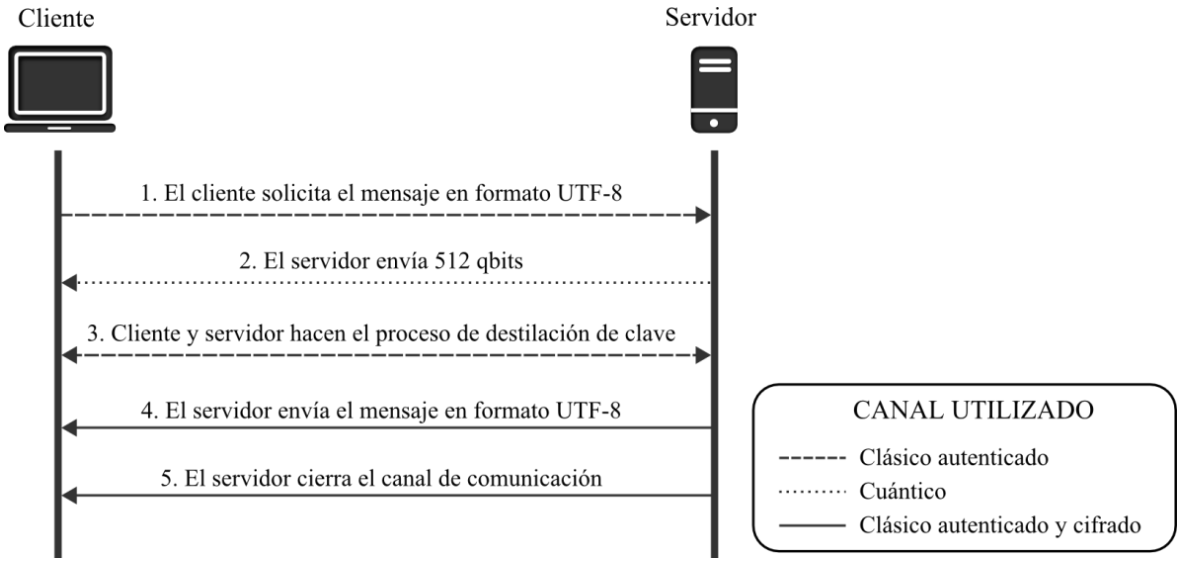

Fig.34. Flujo de datos para la aplicación cliente/servidor

Para la creación del flujo de datos cifrado se emplean las librerías Java Cryptography API – JCA (disponible desde la versión 1.2 de JAVA) y Java Cryptography Extensions – JCE (disponible desde la versión 1.4 de JAVA sin restricciones del gobierno de Estados Unidos de America). En especial las clases CipherOutputStream y CipherInputStream, en conjunción a las clases estándar OutputStream e InputStream.

Un esquema del flujo de información y las clases empleadas, en esta primera aplicación, se puede observar en la figura 35; en donde el servidor usa la clave compartida mediante el sistema QKD para cifrar el flujo de información que transmitirá el mensaje, mediante la clase

CipherOutputStream, para posteriormente transmitir el flujo encapsulándolo en un OutputStream. El proceso inverso tiene lugar en el cliente de forma análoga.

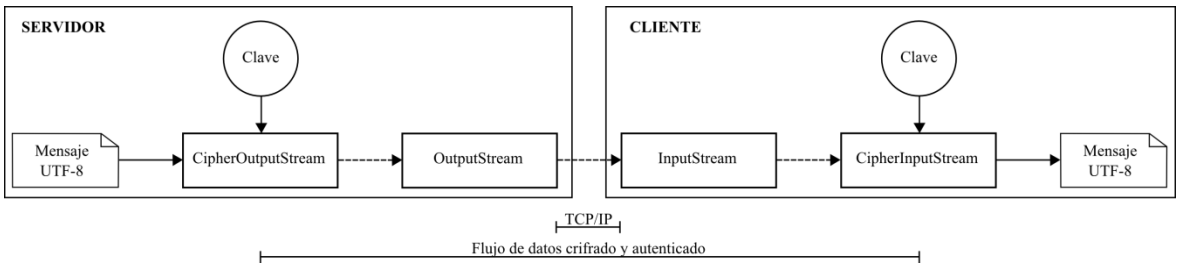

Fig.35. Flujo de información y clases empleadas en la aplicación cliente/servidor

Por su parte, la aplicación de VPN emplea la pila IPSec propia del núcleo de Linux (versiones superiores a la 2.5.47) y la configura para cada una de las conexiones que se quieren establecer entre dos equipos. Con este fin, se obtiene una clave compartida entre los dos equipos mediante el sistema QKD. Luego, se modifican los archivos de configuración para establecer una conexión con difusión manual de claves en modo túnel usando la herramienta *setkey* y privilegios del grupo administrador. Se inicializa la VPN cifrando todo el tráfico con encriptación Triple DES. Mediante esta conexión se envía cualquier tipo de información que se quiera compartir (texto, imágenes, programas de ordenador, etc.). Por último se finaliza la VPN y se reinician los archivos de configuración (figura 36).

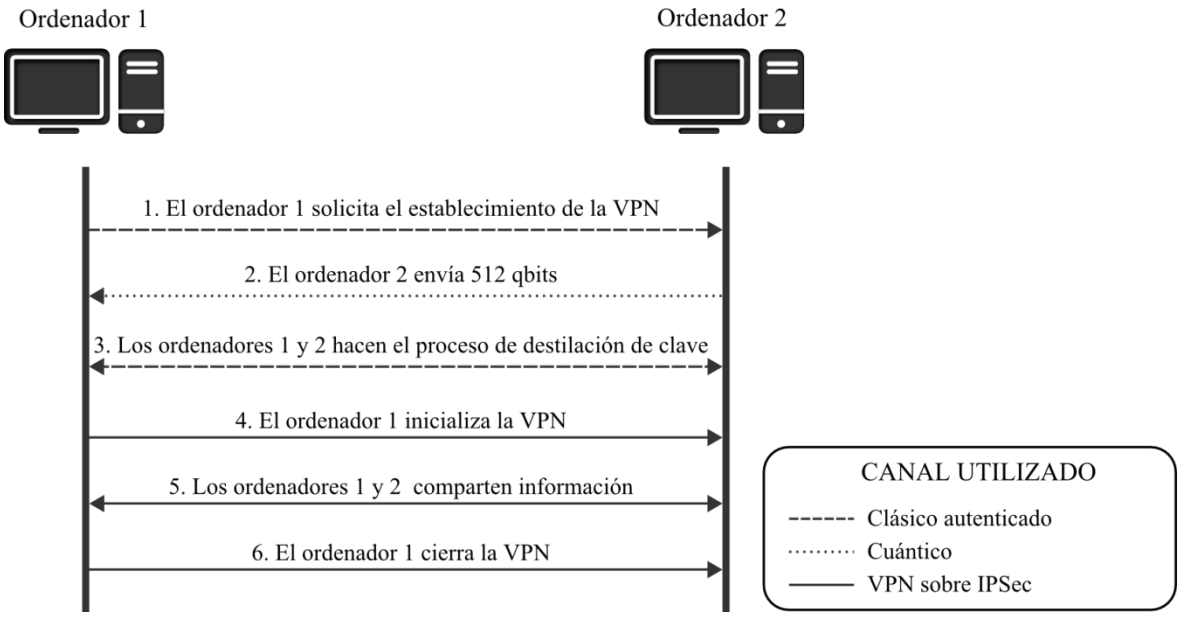

Fig.36. Flujo de datos para la aplicación de VPN

Dos de los conceptos mencionados y que describen el funcionamiento de la VPN sobre IPSec son el de “difusión manual de claves” y el de “modo túnel”. El primero hace referencia al hecho de no emplear los protocolos para el establecimiento de claves de cifrado presentes en IPSec (como el

Internet Key Exchange – IKE [13]), si no el sistema de distribución de clave cuántica y un pequeño script que actualiza los archivos de configuración.

El modo túnel implica que todo el paquete IP (cabeceras incluidas) se cifra y por tanto debe ser encapsulado en un nuevo paquete IP para que pueda ser enrutado por la red; es el modo empleado para las comunicaciones sobre Internet. En contraposición el modo transporte solo cifra los datos mientras que las cabeceras del paquete IP permanecen inalteradas, es el modo usado comúnmente al conectar dos ordenadores.

V. Conclusiones y líneas futuras

V.1. Conclusiones

En este trabajo, se ha diseñado y analizado un sistema de destilación de clave cuántica para el protocolo BB84. Se ha corroborado que el proceso de destilación de clave asegura que el intercambio de claves mediante el canal cuántico sea aprovechable por las aplicaciones debido a que permite la corrección de errores, inherentes a todo canal de comunicaciones, manteniendo la seguridad incondicional lograda mediante las propiedades físicas de la transmisión cuántica.

El tamaño transmitido mediante el canal cuántico suele reducirse drásticamente (hasta un 75%) en el proceso de destilación de clave cuántica. Dado que el proceso de transmisión mediante el canal cuántico presenta unas tasas de bit muy bajas en comparación a las tasas de bit de trasmisión de datos (del orden de los Mbps), se hace necesario encontrar mecanismos que logren mejorar considerablemente estas tasas sin comprometer la seguridad.

Se han analizado cada uno de los procesos que integran la destilación de clave cuántica. Concretamente, se ha comprobado que el algoritmo *Cascade* es capaz de reducir de forma considerable los errores presentes en las claves transmitidas mediante el canal cuántico, haciéndolos cero en la mayoría de los casos. Esto permite que se puedan aprovechar gran parte de las claves que se generan y tramiten mediante el sistema QKD. Sin embargo, el algoritmo Cascade tiene un costo asociado al tiempo de búsqueda binaria y al de transmisión de paridades de bloque para comparación, una optimización puede ser el uso de LDPC en esta sub-capa del proceso de destilación de clave cuántica.

Dentro del proceso de destilación de clave cuántica la capa de detección de errores es la que mayor tiempo consume, este fenómeno indica un aspecto de mejora a considerar para futuras implementaciones, dado que si los tiempos de detección de errores se disminuyen se podrá sin perder la precisión obtenida, se podrán ofrecer tasas de distribución de clave cuántica mayores.

Para lograr mejorar aumentar la tasa de transmisión de claves cuánticas se ha propuesto el uso de las funciones hash criptográficas usadas en la capa de amplificación de la privacidad para lograr ampliar la tasa binaria apoyándose en las propiedades de estas funciones. Así, experimentalmente se han obtenido mejoras en las tasas de bit para la transmisión de claves cuánticas de hasta cuatro veces.

Este trabajo presenta una fundamentación experimental, o línea base, para próximas experimentaciones que permitan aumentar considerablemente la tasa de distribución de clave cuántica sin comprometer la seguridad incondicional. Dada la arquitectura de software sobre la que fue desarrollado cambiar las sub-capas o el protocolo completo será un trabajo dinámico que abrirá las puertas a futuros desarrollo de sistemas de distribución de clave cuántica de alto rendimiento.

V.2. Líneas futuras

En el trabajo presentado se establece una línea base y se identifican varios puntos de mejora para obtener una mayor tasa de distribución de clave cuántica. Sobre estos dos aspectos algunas líneas futuras a desarrollar son:

- Realizar nuevas implementaciones de protocolos de distribución de clave cuántica como el B92 y el SARG04.
- Usando un mismo protocolo, experimentar con modificaciones algorítmicas de las subcapas: como el cambio de Cascade por LDPC en la capa de corrección de errores o el uso de diversas funciones hash
- Verificación experimental de los mecanismos de detección de errores
- Comprobación matemática de la propuesta del usos de funciones hash criptográficas para ampliar la tasa de bit cuántica manteniendo la seguridad incondicional.
- Integración en una red de distribución de clave cuántica de alto rendimiento.

Además, con los conocimientos obtenidos y los conceptos desarrollados, la línea de investigación se puede expandir para abarcar áreas de las comunicaciones cuánticas como:

- Tele-portación de información,
- Distribución de clave cuántica sobre medios de transmisión no guiados

AGRADECIMIENTOS

Al gobierno Colombiano, quien ha apoyado este trabajo de investigación mediante la beca Francisco José de Caldas, a mis compañeros del Grupo de Comunicaciones Ópticas y Cuánticas del ITEAM así como a mis directores, José Mora y José Capmany, cuya guía constante ha sido fundamental para la realización de este trabajo.

A Dios, mi esposa, familia y amigos

BIBLIOGRAFÍA

[1] C. H. Bennett y G. Brassard, *Quantum cryptography: plubic key distribution and coin tossing*, Proceedings of IEEE International Conference on Computers, Systems and Signal Processing, IEEE Press., pp. 175-179, 1984.

[2] C. H. Bennett, *Quantum Cryptography Using any Two Nonorthogonal States*, Phys. Rev. Lett., Vol. 68, No. 21, pp. 3121, 1992.

[3] V. Scarani, A. Acín, G. Ribordy y N. Gisin, *Quantum Cryptography Protocols Robust against Photon Number Splitting Attacks for Weak Laser Pulse Implementations*, Phys. Rev. Lett, Vol. 92, 2004.

[4] V. Scarani, H. Bechmann-Pasquinucci, N.J. Cerf, N. Lütkenhaus, M. Dušek y M. Peev, *The security of practical quantum key distribution*, Reviews of modern physics 81, 1301-1310, 2009.

[5] G. Brassard y L. Salvail, *Secret-Key Reconciliation by Public Discussion*, Lecture Notes in Computer Science, Vol. 765, pp. 411-423, 1994.

[6] W. Buttler, J. Torgerson, G. Nickel, C. Donahue y C. Peterson. *Fast, efficient error reconciliation for quantum cryptography*. Physical Review A, 67(5), 2003.

[7] C. Elliot, A. Colvin, D. Pearson, O. Pikalo, J. Schlafer y H. Yeh, *Current status of the DARPA Quantum Network*, BBN Technologies, 2005.

[8] D. Pearson, *Building a QKD Network out of Theories and Devices*, Building the DARPA Quantum Network, BBN Technologies, 2005.

[9] W. Stallings, *Cryptography and Network Security: Principles and Practice 5th Edition*, Pearson, 2011

[10] N. Gisin, G. Ribordy, W. Tittel y H. Zbinden, *Quantum Cryptography*, Rev. Mod. Phys., Vol. 74, pp. 145, 2008.

[11] V. Manral, *Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH), Request for Comments: 4835*, Internet Engineering Task Force (IETF), 2007.

[12] W. C. Barker y E. Barker, *Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher*, National institute of standards and technology, 2012.

[13] C. Kaufman, P. Hoffman, Y. Nir y P. Eronen, *The Internet Key Exchange Protocol Version 2, Request for Comments: 5996*, Internet Engineering Task Force (IETF), 2010.

Anexos

[1] J. Pradilla, J. Mora y J. Capmany, *Amplification of the transmission rate for quantum key distribution based on SCM*, International IEEE Topical Meeting on Microwave Photonics (MWP 2013), Oct 28-31 de 2013, Alexandria - USA.

[2] J. Pradilla, J. Mora y J. Capmany, *Amplificación de la tasa de bit para distribución de clave cuántica basada en el protocolo BB84*, VIII Reunión Española de Optoelectrónica, Jul 10-12 de 2013, Alcalá de Henares – España.

Amplification of the transmission rate for quantum key distribution based on SCM

Juan Pradilla, José Mora and José Capmany
Grupo de Comunicaciones Ópticas y Cuánticas (GCOC)
Instituto de Telecomunicaciones y Aplicaciones Multimedia (iTEAM)
Valencia, Spain
jmalmer@iteam.upv.es

***Abstract*— The security represents one of the most important perspective future lines in telecommunications. This work is focused on new techniques and methods employing the properties of quantum physics to provide the ability to obtain an unconditional security. This article proposes the use of cryptographic hash functions used in the amplification layer of privacy to expand the bit rate relying on the properties of these functions. In this way, our approach permits to improve the quantum key transmission rate when optically is limited. We demonstrate the potentiality of the proposal with an amplification of 6 dB.**

Keywords*— *Quantum key distribution, hash functions, BB84 protocol, privacy amplification

I. INTRODUCTION

Security in telecommunications has emerged as one of the topics of greatest development and relevance in the world. Applications such as telecommunications between nuclear power plant, financials operations and military communications, need to guarantee maximum security for its daily operation [1].

A new set of protocols has appeared as a response to these requirements of unconditional security. In this way, Quantum Key Distribution (QKD) is a research line which permits to generate and distribute digital keys using the principles of quantum physics. The security based on the principles of physics suggests the possibility of unconditional security, which implies the possibility to prove that the system is safe without any restriction on the capabilities that a hacker has, except the limits determined by physics [1].

Examples of such protocols are: BB84, proposed by Bennett and Brassard in 1984 [1]; B92, a modification to the BB84 protocol, proposed in 1992 by Bennett [2], which does not provide great advantages over its predecessor, so that their interest is only academic; and SARG04, proposed in 2004 by Scarani, Acín, Ribordy and Gisin [3], as an alternative to the BB84 protocol, to prevent the photon number splitting attack, PNS. All these protocols use quantum physics principles [4]. Particularly, the uncertainty principle of Heisenberg which ensures that it is impossible to determine with absolute precision and simultaneously, the value of two conjugate variables of an elemental system. Also, the no-cloning theorem proposed by Dieks, Wootters and Zurek, which ensures that you cannot accurately clone an unknown quantum state while keeping the original unmodified. Finally, these quantum correlations obtained from separate measurements of entangled states violate Bell's inequality and prevent you from creating an agreement before the measurement. However, these protocols have an important limitation related to the final key rate. Currently, the key rates (of the order of Mbps at distances of 100 km) are low compared to the transmission data rates (of the order of Gbps or Tbps over distances of 100 km). For this reason, we focus our attention into increase the binary rate of quantum key distribution (QKD) in order to obtain faster and more secure communications systems.

This article explores the operation of the BB84 protocol, and proposes a method to increase the binary rate of the quantum key distribution systems using subcarrier multiplexing (SCM-QKD). In this way, a novel technique is presented which uses the properties of cryptographic hash functions to increase the bit rate of the quantum key distribution systems. In this work, we demonstrate the potentiality of the proposal with an amplification of 6 dB.

II. BB84 OPERATION

The BB84 protocol is represented with the action of three actors: a sender (Alice), a receiver (Bob) and a hacker (Eve). Alice and Bob use two communication channels: a quantum channel, allowing them to share quantum signals; and a classical channel, in which messages can be sent using traditional methods (Fig. 1) [4].

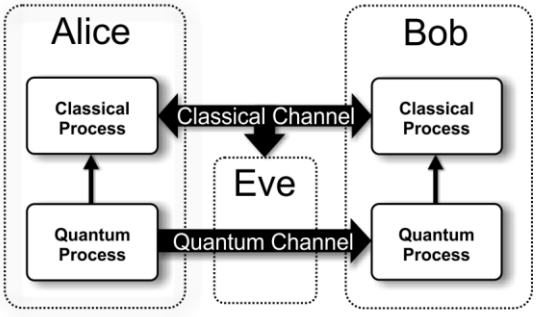

Fig. 1. *BB84 protocol model.*

The classical channel is an authenticated channel, which means that Alice and Bob must identify each other directly or through a third part (certification authority). Eve, in turn, can hear the classical conversation, but not participate. However,

the quantum channel is open to any manipulation by Eve. Therefore, the BB84 protocol can be divided into two parts, depending on which quantum or classical channel is used.

A. Key Exchange Through the Quantum Channel

In our case, the quantum key distribution was realized by applying the concept of subcarrier multiplexing (SCM) to extend the capacity of frequency coding schemes for QKD systems [5]. In this way, multiple subcarrier frequencies can be simultaneously used to increase the bit rate without compromising the security of the protocol. Figure 2 outlines the key exchange process through the quantum channel.

SCM-QKD [5] uses a faint pulse laser source emitting at frequency $\omega_0=2\pi f_0$ is externally modulated by N radiofrequency subcarriers by Alice. Each subcarrier with angular frequency $\Omega_i=2\pi f_i$, is generated by a local oscillator (LO_i) and randomly phase modulated among four possible values 0, π and π/2, 3π/2 which let the encoding of the bits and form a pair of conjugated bases required to implement a BB84 protocol. The compound signal is then sent through an optical fiber link and, upon reaching Bob's location is externally modulated by N identical subcarriers in a second modulator. These subcarriers are phase modulated among two possible values 0 and π/2, which represent the choice between the two bases, to decode the bits. As a consequence, an interference single-photon signal is generated at each of the sidebands (upper and lower) of each subcarrier.

As above mentioned, the protocol BB84 send a random set of qubits (quantum bits) coded in four states:

$$\begin{aligned} &\textit{Base } 1 \longleftrightarrow \begin{cases} |\psi_0\rangle = |0\rangle \\ |\psi_1\rangle = |1\rangle \end{cases} \\ &\textit{Base } 2 \longleftrightarrow \begin{cases} |\psi_+\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle + |1\rangle\right] \\ |\psi_-\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle - |1\rangle\right] \end{cases} \end{aligned} \qquad (1)$$

The first two states in (1) form one base of a quantum two dimensional system while the other two from a second one. Therefore, the conditions $\langle\psi_0|\psi_1\rangle = 0$ and $\langle\psi_+|\psi_-\rangle = 0$ corresponding to scalar product between stated have to be satisfied. At the same time, the states in the different bases of (1) are not orthogonal and have maximum overlapping. As a consequence, there is no measurement procedure that can determine with a 100% certainty the specific state which is prepared in Alice and it is sent to Bob. Meanwhile, Bob determines the state that Alice has sent choosing randomly one of two possible bases of $\langle\psi_{0,1}|\psi_{+,-}\rangle$ measuring and storing the result. Bob choose the same basis Alice in 50% of cases as expected.

While Alice transmits the qbits to Bob, Eve can interact with them and interfere with communication. Eve does not know which of the four states of the expression (1) has selected Alice. Therefore, Eve must choose randomly one of the two bases for its actions. In case of choosing the same basis as Alice, Eve can convey the correct state to Bob. If a different base is selected, Eve transmit to Bob a wrong state and the presence of spyware can be detected. It is important to note that Eve will not know if Bob has chosen the correct basis until communication takes place through the public channel. Therefore, there is a 50% probability that Eve has used the wrong base on its measurements.

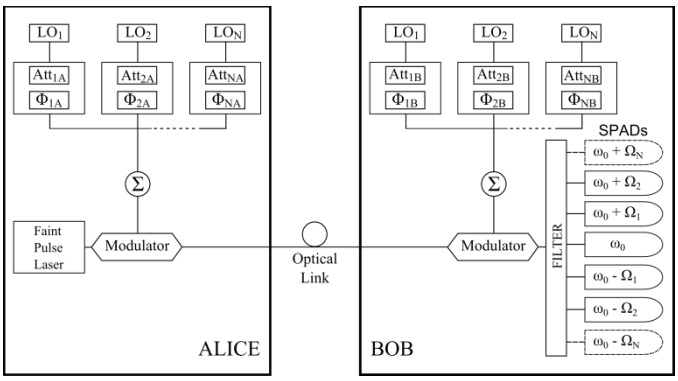

Fig. 2. *SCM-QKD system layout.*

Unfortunately, the quantum key transmission remains a new discipline within telecommunications, which entails that the systems used are still novel and therefore the key transmission rates are low.

B. Quantum Key Distillation Through the Classical Channel

Once Alice and Bob share the key through the quantum channel, the process of key distillation is carried out through the classical channel on both ends. This process is composed of four layers: Sifted, Error Detection, Error Correction and Privacy Amplification.

In order to evaluate the distillation process, Fig. 3 shows the processed key in each layer. In this case, we consider a transmitted key with 256 qbits by controlling the error rate of the sample. The relation between the discarded qubits and sent is known as quantum bit error rate (QBER). This magnitude is relevant to decide whether a key is distributed with unconditional security.

In the sifted layer, Alice sends Bob the bases selected to encode each one of the qubits, but without disclose the selected state at any time. Alice and Bob retain the qubits which were measured with the correct base and discard the rest. As a result, a key is obtained of approximately half of the original length (Fig. 3a). In case of a spy intercepts the communication, key length will be about 25% whereupon the entire key is discarded. Through the error detection, Alice and Bob have a simple way to detect the presence of Eve. This estimation is possible because an information leak is quantifiable by means of communication degradation in a quantum channel [4].

The error detection layer shares a small portion of the key between Alice and Bob to try to determine the amount of the quantum channel errors introduced in the communication. All shared bits are removed from the key (Fig. 3b). Therefore, the resulting key is introduced in the following error correction layer in order to mitigate the errors introduced by the channel.

The error correction layer can employ various algorithms to correct the most of errors, and in turn minimizes the amount of information that is revealed by the classical channel. The

algorithms used include Cascade [6], Winnow [7] and the use of LDPC [8, 9].

In the case of Cascade algorithm, which is the most used in conjunction with BB84 and used in the experiment, use block parities and binary search. This algorithm does not discard bits of the key as shown comparing Fig. 3b and Fig. 3c.

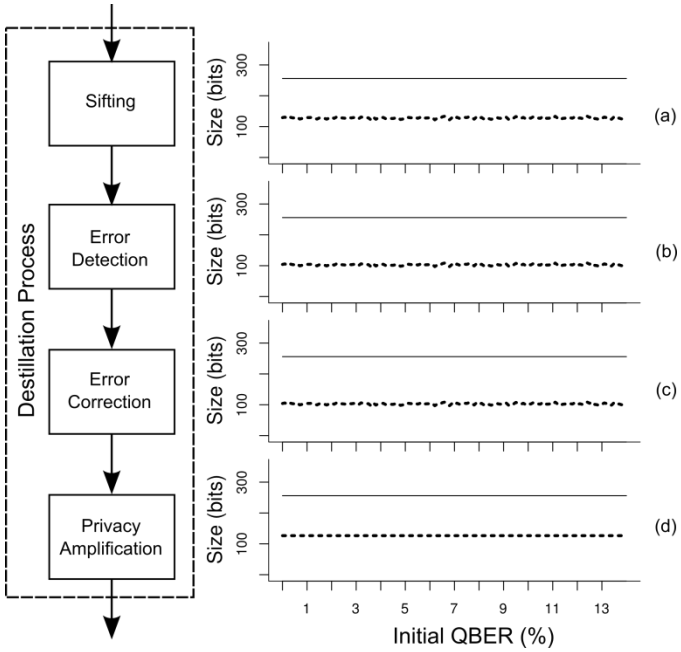

Fig. 3. *Key size in quantum key distillation layers for (a) sifting, (b) error detection (c) error correction (d)privacy amplification.The solid represents the quantum key after the transmission.*

At this point, Alice and Bob have an identical key and it is passed to the privacy amplification layer. This layer commonly use hash function to encode the key bits, so that the Eve information's about the key is reduced.

Hash functions used in the privacy amplification are known as cryptographic hashing, and must meet the following criteria: [10]:

- Unidirectionality: given a hash H, should be computationally infeasible to find a message M such that H = hash (M).
- Collision resistance: Given M1 should be computationally infeasible to find M2 such that H = hash (M1) = hash (m2).
- Consistency: given H1 = hash (M1) and H2 = hash (M2), H1 and H2 must differ in about 50% of its bits, when M2 equals M1 but modified in a bit.
- Ease of calculation: should be easy to calculate hash (M) from a message M.
- Outcome constant: for a message M the function hash (M) must always give the same amount of bits (n), without depending on the size of M.

III. Privacy Amplification and the Bit Rate Amplification

An important problem of quantum key distribution systems is the low key rate which is achieved. To expand key size without compromising security provided by the BB84 protocol is proposed use cryptographic hash functions. The proposal consists on putting a key of fewer bits in the input than the output of the privacy amplification layer, taking advantage of the properties of uniformity and constant result.

To test the proposal, different scenarios were considered in which the information that Eve retains from Alice. This Eve's information is represented by a percentage of the key retrieved by Bob which goes from 86% to 99% (worst case scenario possible). Furthermore, an initial BER was defined between the keys of Bob and Eve as the difference between their keys (the number of different bits) divided by the total size of the key which has been obtained by the BB84 protocol (number of bits in Bob's key). According to the percentage of Eve's information, this initial BER varies between 1% and 14% in steps of 0.1%.

In our distillation process, Bob and Eve have four keys of 128 bits (512 bits in total) with a given initial BER. Each key is passed as a parameter to a hash function that delivers 128 bits at the output. The four resulting keys are concatenated to obtain a final key of 512 bits for Bob and Eve. Comparing the different bits between the Bob final key's and the Eve final key's and dividing by the total size of the final key is obtained the measured QBER. To find an average behavior, the process is performed 10 times for each initial BER and averaged.

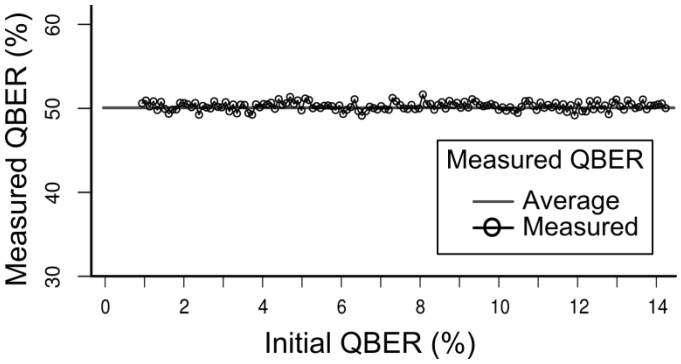

Fig. 4. *Measured QBER in funtion of the Eve's information given by an initial BER. Initial key 128 bits x 4, hash function 128 bits. Final key 512 bits.*

The result of Fig. 4 will serve as a baseline since an initial BER between 1 % and 14 % (i.e. Eve has between 86 % and 99 % of the key), the difference between Bob and Eve final key's differ by 50%. This fact is attributable to the properties of cryptographic hash functions (in this case MD5 [11]).

To observe the impact of amplifying the size of the final key, it is proposed a similar procedure generating four keys of 128, but using cryptographic hash functions with output 512 bits (function SHA-512[12]). Therefore, the resulting four keys are concatenated thus obtaining a final key of 2048 bits in Bob and Eve. Finally, the QBER is calculated between the two final keys.

Fig. 5 shows the results and indicates that the final keys differ by 50% for initial BER between 1% and 14% as previously. Note this safety is not compromised when expanding the size of the key. This result is significant since a four times improvement is achieved in the final key rate with this simple procedure which uses very few computer resources. Obviously, the security of the final key to this method depends entirely on the BB84 protocol at this point ensures unconditional security unequivocally.

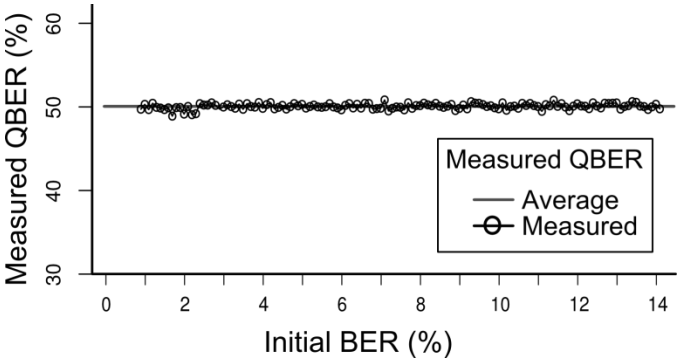

Fig. 5. *Measured QBER in funtion of the Eve's information given by an initial BER. Initial key 128 bits x 4, hash function 512 bits. Final key 2048 bits.*

IV. CONCLUSION

The security is a constantly evolving subject where new techniques and methods employing the properties of quantum physics are making their way to provide unconditional security. However, a limitation is found comparing the key rates with the current transmission rate.

In order to improve the transmission quantum key rate has been proposed the use of cryptographic hash functions. This approach used in the amplification layer of privacy permits to expand the bit rate relying on the properties of these functions. Improvements experimentally obtained on the quantum key bit rate are around of four times.

This paper presents a guide for future experiments to use the process iteratively and obtain exponential performance about this effect. Finally, in either case to achieve this improvement in bit rate without compromising the security of the key which is necessary unconditional security is ensured by the quantum protocol.

ACKNOWLEDGMENT

This work was supported by the help of the Polytechnic University of Valencia in the Proof of Concept SP20120588 and Colciencias by Francisco José de Caldas scholarship.

REFERENCES

[1] C. H. Bennett and G. Brassard, "Quantum cryptography: plubic key distribution and coin tossing", Proceedings of IEEE International Conference on Computers, Systems and Signal Processing, IEEE Press., pp. 175-179, 1984.

[2] C. H. Bennett, "Quantum Cryptography Using any Two Nonorthogonal States", Phys. Rev. Lett., Vol. 68, No. 21, pp. 3121, 1992.

[3] V. Scarani, A. Acín, G. Ribordy and N. Gisin, "Quantum Cryptography Protocols Robust against Photon Number Splitting Attacks for Weak Laser Pulse Implementations", Phys. Rev. Lett, Vol. 92, 2004.

[4] V. Scarani, H. Bechmann-Pasquinucci, N.J. Cerf, N. Lütkenhaus and M. Peev, "The security of practical quantum key distribution", Reviews of modern physics 81, 1301-1310, 2009.

[5] A. Ruiz-Alba, D. Calvo, V. Garcia-Muñoz, A. Martinez, W. Amaya, J.G. Rozo, J. Mora, J. Capmany, "Practical Quantum Key Distribution based on the BB84 protocol", Waves, 2011

[6] G. Brassard and L. Salvail, "Secret-Key Reconciliation by Public Discussion", Lecture Notes in Computer Science, Vol. 765, pp. 411-423, 1994.

[7] W. Buttler, J. Torgerson, G. Nickel, C. Donahue, and C. Peterson. "Fast, efficient error reconciliation for quantum cryptography", Physical Review A, 67(5), 2003.

[8] C. Elliot, A. Colvin, D. Pearson, O. Pikalo, J. Schlafer and H. Yeh, "Current status of the DARPA Quantum Network", BBN Technologies, 2005.

[9] D. Pearson, "Building a QKD Network out of Theories and Devices", Building the DARPA Quantum Network, BBN Technologies, 2005.

[10] W. Stallings, "Cryptography and Network Security: Principles and Practice", 5th Edition, Pearson, 2011.

[11] S. Turner, L. Chen, "RFC 6151: Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms", IETF, 2011.

[12] S. Gueron, S. Johnson, and J. Walker, "SHA-512/256", ;presented at IACR Cryptology ePrint Archive, 2010, pp.548-548. A

Amplificación de la tasa de bit para distribución de clave cuántica basada en el protocolo BB84

Amplification of the bit rate for quantum key distribution based on BB84 protocol

Juan PRADILLA, José MORA, José CAPMANY

Grupo de Comunicaciones Ópticas y Cuánticas (GCOC), Instituto de Telecomunicaciones y Aplicaciones Multimedia (iTEAM), Universitat Politècnica de València, C/ Camino de Vera s/n 46022, Valencia, Spain.

Persona de contacto: José Mora (jmalmer@ upv.es).

RESUMEN:

La seguridad es un campo de trabajo en continuo desarrollo y representa una de las líneas con mayor perspectiva de futuro dentro de las telecomunicaciones. Este trabajo se centra en las nuevas técnicas y procedimientos que emplean las propiedades de la física cuántica que se están abriendo camino al brindar la posibilidad de tener una seguridad incondicional. Actualmente, estos sistemas de distribución de clave carecen de una tasa de clave equiparables con las tasas de transmisión actuales. Para lograr mejorar esta desigualdad entre tasa de transmisión de datos y la tasa de transmisión de clave cuántica se propone en este artículo el uso de las funciones hash criptográficas usadas en la capa de amplificación de la privacidad para lograr ampliar la tasa binaria apoyándose en las propiedades de estas funciones.

Palabras clave: Distribución de clave cuántica, funciones hash, protocolo BB84, amplificación de privacidad

ABSTRACT:

The security is a research area a continuous evolving field and represents one of the most important perspective future lines in telecommunications. This work is focused on new techniques and methods employing the properties of quantum physics to provide the ability to obtain an unconditional security. Currently, these techniques and methods lack a key rate equivalent to the current transmission rate. In order to improve the quantum key transmission rate, this article proposes the use of cryptographic hash functions used in the amplification layer of privacy to expand the bit rate relying on the properties of these functions.

Key words: Quantum key distribution, hash functions, BB84 protocol, privacy amplification

1.- Introducción

En la actualidad, la seguridad en las comunicaciones se perfila como uno de los temas de mayor desarrollo y relevancia en el mundo. Aplicaciones como las comunicaciones en centrales nucleares, las operaciones interbancarias y las redes de organismos de inteligencia militar requieren que se garantice la máxima seguridad para su operación diaria.

Para dar respuesta a estos requerimientos crecientes de seguridad han comenzado a emerger protocolos para generación y distribución de claves (conocidos como *Quantum Key Distribution*, QKD) que utilizan los principios de la física cuántica

con el objetivo de garantizar una seguridad incondicional. Ejemplos de estos protocolos son: el BB84, que fue propuesto por Bennett y Brassard en la International Conference on Computers, Systems and Signal celebrada en Los Álamos, California, en el año 1984 [1]; el B92, que es una modificación al protocolo BB84 propuesta en 1992 por Bennett [2], la cual no brinda grandes ventajas sobre su predecesor, por lo que su interés no va más allá del académico; y el SARG04, propuesto en el año 2004 por Scarani, Acín, Ribordy y Gisin [3], como una alternativa al protocolo BB84, para evitar el ataque por división del número de fotones, PNS.

Todos estos protocolos utilizan los principios de la física cuántica. Principalmente el principio de incertidumbre de Heisenberg que asegura que es imposible determinar, con precisión absoluta y de forma simultánea, el valor de dos magnitudes conjugadas de un sistema elemental. Por otro lado, el teorema de la no clonación propuesto por Dieks, Wootters y Zurek, asegura que no se puede clonar de forma exacta un estado cuántico desconocido manteniendo el original sin modificaciones, y finalmente que las correlaciones cuánticas obtenidas de medidas separadas de estados entrelazados violan la desigualdad de Bell y por tanto impiden crear un acuerdo antes de la medida [4]. El hecho de que la seguridad esté basada en los principios de la física sugiere la posibilidad de seguridad incondicional, la cual, implica la posibilidad de demostrar que el sistema es seguro sin ninguna restricción en las capacidades que tenga un atacante, salvo los límites fijados por la física [1].

Sin embargo, todos estos protocolos presentan una carencia importante: sus tasas binarias (del orden de kbps a distancias de 100 km) son bajas comparadas con las tasas de transmisión (del orden de Gbps o Tbps a distancias de 100 km). Esto ha llevado a centrar la atención de varios investigadores con el fin aumentar las tasas binarias de los protocolos de distribución de clave cuántica (QKD) de forma que se equipare la generación y distribución de claves con la generación y distribución de información, logrando así, sistemas de comunicaciones más rápido y seguros.

El presente artículo explora el funcionamiento del protocolo BB84 y propone una técnica, para aumentar la tasa binaria de los sistemas de distribución de clave cuántica. Inicialmente presenta el funcionamiento del protocolo BB84. Posteriormente, muestra una técnica que usa las propiedades de las funciones hash criptográficas para el aumento de la tasa binaria. Finalmente, muestra las conclusiones del trabajo realizado utilizando esta técnica.

2.- Funcionamiento del protocolo BB84

El protocolo BB84 para la distribución de clave cuántica se modela como la interacción de tres actores: un emisor (Alice), un receptor (Bob) y un atacante (Eve). Para comunicarse Alice y Bob emplean dos canales de comunicación: uno cuántico, que les permitirá compartir señales cuánticas, y un canal clásico, en el cual pueden enviar mensajes de forma clásica [4].

El canal clásico necesita ser autentificado, lo que implica que Alice y Bob deben identificarse entre ellos directamente o a través de un tercero (entidad certificadora). Eve, por su parte, puede escuchar la conversación clásica, pero no participar en ella. Sin embargo, el canal cuántico está abierto a cualquier manipulación por parte de Eve. De forma que, la tarea de Alice y Bob es garantizar la seguridad, teniendo en cuenta el libre acceso de Eve para manipular el canal cuántico y escuchar la transmisión del canal clásico. La figura 1 muestra un esquema general de esta técnica.

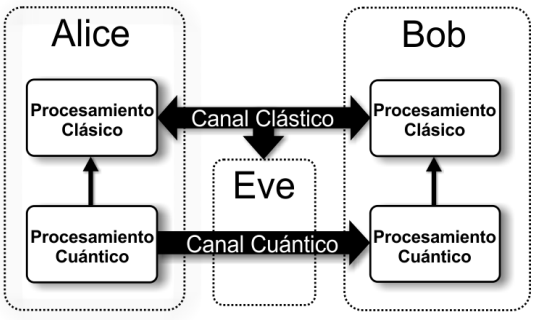

Fig. 1: Modelo de un protocolo de QKD.

Sobre este modelo, el protocolo BB84 se puede dividir en dos partes, según el canal que este empleando: cuántica y clásica. Estas

dos partes de presentan en profundidad a continuación.

2.1.- Intercambio de la clave a través del canal cuántico

En la primera parte del protocolo BB84 se emplea el canal cuántico, a través del cual, Alice envía a Bob un conjunto aleatorio de qbits (bits cuánticos) codificados según cuatro estados. Estos cuatro estados se agrupan formando dos bases con estados ortogonales: los dos primeros estados de la expresión (1) forman una primera base y los otras dos forman una segunda, logrando que las condiciones $\langle\psi_0 | \psi_1 \rangle=0$ y $\langle\psi_+ | \psi_x\rangle=0$, correspondientes al producto escalar entre estados, sean satisfechas. Al mismo tiempo, los estados de diferentes bases de la expresión (1) no son ortogonales, ya que $\langle\psi_{0,1} | \psi_{+,x}\rangle\neq 0$. Así, un estado queda absolutamente determinado al proyectarlo sobre su correspondiente base, mientras que el resultado será totalmente aleatorio si se proyecta en la otra base.

$$Base_1 = \begin{cases} |\psi_0\rangle = |+, x\rangle = |0\rangle \\ |\psi_1\rangle = |-, x\rangle = |1\rangle \end{cases}$$

$$Base_2 = \begin{cases} |\psi_+\rangle = |+, y\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle + |1\rangle\right] \\ |\psi_-\rangle = |-, y\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle - |1\rangle\right] \end{cases} \quad (1)$$

Por su parte, Bob determina el estado que Alice le ha enviado escogiendo aleatoriamente, para cada uno de los estados recibidos, una de las dos posibles bases de $\langle\psi_{0,1} | \psi_{+,x}\rangle$, midiendo y almacenando el resultado. Es de esperar que Bob escoja la misma base que Alice en el 50% de los casos.

Mientras Alice trasmite a Bob sus estados, Eve puede interactuar con ellos e interferir en la comunicación. Eve no sabe cuál de los cuatro estados de la expresión (1) ha seleccionado Alice. Por tanto, debe escoger aleatoriamente una de las dos bases para realizar sus medidas. En el caso que acierte y escoja la misma base de Alice, Eve podrá trasmitir el estado correcto a Bob. En el caso que seleccione una base distinta, Eve trasmitirá a Bob un estado incorrecto y este podrá detectar la presencia del espía. Es importante remarcar que Eve no sabrá si ha elegido la base correcta hasta que se lleva a cabo la comunicación por el canal público. Por tanto, existe el 50 % de probabilidad de que Eve haya usado la base incorrecta en sus medidas. La figura 2 esquematiza el proceso de intercambio de clave a través del canal cuántico.

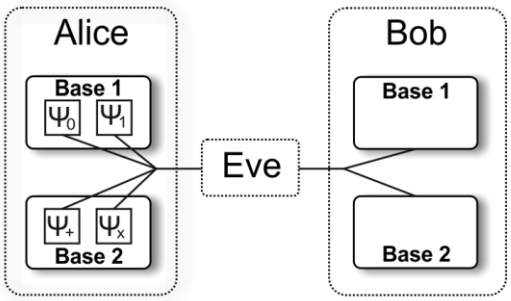

Fig. 2: Esquema del procedimiento para implementar el protocolo BB84.

Alice y Bob tienen una forma sencilla de detectar la presencia de Eve, observando si al realizar la fase inicial del proceso de reconciliación de claves encuentran que las bases escogidas en los dos extremos coinciden en un porcentaje cercano o inferior al 25%. Esta estimación es posible porque en un canal cuántico, una fuga de información es cuantificable por medio de la degradación de la comunicación [4].

Desafortunadamente, la transmisión de claves cuánticas sigue siendo una disciplina novel dentro de las telecomunicaciones, lo que conlleva a que los componentes utilizados sean aun imprecisos y por tanto las tasas de transmisión de clave sean bajas. Por ejemplo, los emisores de pulsos atenuados y los detectores de fotones suponen porcentajes de detecciones máximas inferiores al 10%.

2.2.- Destilación de clave cuántica por canal clásico

Una vez que el envío de la clave finaliza por parte de Alice, comienza el proceso de destilación de clave que se lleva a cabo a través del canal clásico en los dos extremos. Este proceso está compuesto por cuatro capas sucesivas representadas en la figura 3: Reconciliación de bases (S*ifted*), detección de errores (*Error Detection*), corrección de

errores (*Error Correction*) y amplificación de la privacidad (*Privacy Amplification*).

En la capa de reconciliación de bases, Alice envía a Bob las bases que ha seleccionado para codificar cada uno de los qbits, pero sin revelar en ningún momento el estado seleccionado. De igual forma Bob comparte que bases ha empleado para realizar las medias en cada uno de los qbits que ha recibido, sin indicar el resultado de la medida. Alice y Bob retienen los qbits donde la base elegida por ambos coincide y descartan el resto. Como resultado obtienen una cadena de bits de longitud aproximada la mitad de la original, hecho que inmediatamente reduce a la mitad la tasa de bit y degrada enormemente la comunicación. En el caso de que un espía intercepte la comunicación, la longitud de la clave será aproximadamente del 25% con lo cual se descarta la clave completa.

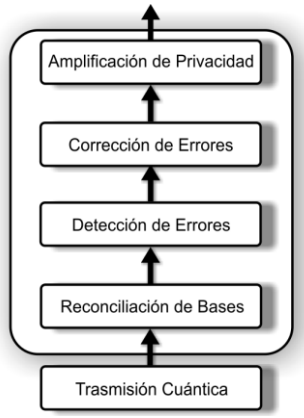

Fig.3: Capas del proceso de destilación de claves.

La relación entre los qbits descartados y los enviados es conocida como la tasa de errores cuánticos (QBER, Quantum Bit Error Rate), magnitud que resultará clave para decidir si se puede transmitir una clave con seguridad incondicional; en realidad, con un número acotado de bits conocidos por un hipotético espía. Esta cota se debe ajustar a los requerimientos de la aplicación que usara la clave y las características del canal.

Por su parte, la capa de detección de errores comparte una pequeña parte de la clave en crudo entre Alice y Bob para tratar de determinar la cantidad de errores que el canal cuántico ha introducido en la comunicación. Los bits compartidos son eliminados de la clave, de forma que la clave resultante junto con la estimación realizada se pasa la capa de corrección de errores para intentar mitigar los errores introducidos por el canal.

En la capa de corrección de errores se pueden emplear diversos algoritmos que corrijan el mayor número de errores y a su vez minimicen la cantidad de información que se revela mediante el canal clásico. Los algoritmos más utilizados incluyen el *Cascade* [5], el *Winnow* [6] y el uso de *LDPC* [7,8].

Para el caso del algoritmo *Cascade*, que es el más empleado en conjunción con el BB84, la clave se divide en bloques de igual tamaño. Se calcula la paridad a cada uno de los bloques y se intercambia la misma entre Alice y Bob. Si la paridad es la misma se asume que ese bloque no contiene errores, de lo contrario se realiza una búsqueda binaria del error en el bloque. La búsqueda binaria consiste en dividir el bloque en dos y para cada uno de los sub-bloques calcular la paridad y compartirla con el otro extremo de la comunicación. Si las paridades son iguales el bloque no se subdivide. De no ser así, el proceso se repite hasta encontrar el bit erróneo y se corrige.

Siguiendo con el algoritmo *Cascade*, después de corregir los errores en los bloques iníciales mediante la búsqueda binaria, se reorganizan de forma aleatoria los bits de la clave, y se divide la clave nuevamente para repetir el proceso completo. Se hacen máximo cuatro iteraciones aumentando al doble, en cada ocasión, el tamaño de los bloques iníciales. Se considera que el algoritmo ha corregido todos los errores si llega a las cuatro iteraciones o si el tamaño inicial de bloque supera al tamaño total de la clave.

Contando con una clave idéntica en Alice y Bob se pasa la clave a la capa de amplificación de la privacidad, en esta se emplea comúnmente una función hash para codificar los bits de la clave de forma que la información que Eve pueda tener de la clave se reduzca a cero.

Las funciones hash utilizadas en la amplificación de la privacidad son conocidas

como hash criptográficas y deben cumplir los siguientes criterios [9]:

• Unidireccionalidad (Resistencia pre-imagen): dado un hash **H**, debe ser computacionalmente inviable encontrar un mensaje **M** tal que **H = hash(M)**

• Resistencia a colisiones: dado **M1** debe ser computacionalmente inviable encontrar **M2** tal que **H = hash(M1) = hash(M2)**

• Uniformidad: dado **H1 = hash(M1)** y **H2 = hash(M2)**; **H1** y **H2** deben diferir en aproximadamente el 50% de sus bits, cuando M2 es igual a M1 con un bit modificado

• Facilidad de cálculo: Debe ser fácil calcular **hash(M)** a partir de un mensaje M

• Resultado constante: para un mensaje **M** la función **hash(M)** debe siempre entregar la misma cantidad de bits (n), sin depender del tamaño de **M**

3.- Amplificación la privacidad y la tasa binaria

Como se mencionó anteriormente uno de los grandes problemas de los sistemas de distribución de clave cuántica es la baja tasa de bits que alcanzan. Para buscar ampliar el tamaño de la clave sin comprometer la seguridad que aporta el protocolo BB84 se propone la hipótesis de emplear las funciones hash criptográficas pasándoles como entrada una clave de tamaño inferior al de la salida de las mismas, aprovechando así las propiedades de uniformidad y resultado constante.

Para comprobar el funcionamiento de la hipótesis se creó un sistema en el cual Eve tiene entre el 86 % y el 99 % de la clave que ha obtenido Bob, escenificando así un escenario de peor caso posible. Además, se ha definido el QBER inicial entre la clave de Eve y la de Bob como la diferencia entre las sus claves (número de bits diferentes) sobre el tamaño total de la clave que se ha obtenido mediante el protocolo BB84 (número de bits de la clave de Bob). Este QBER inicial varia entonces entre 1 % y 14 % en pasos de 0.1%.

Así, en Bob y en Eve se generan cuatro claves de 128 bits (512 bits en total) para cada valor de QBER inicial. Cada una de las claves es pasada como parámetro a una función hash que entrega 128 bits a su salida (obteniendo así 512 bits en total). Las cuatro claves resultado de la función hash se concatenan una después de la otra obteniendo una clave final de 512 bits en Bob y en Eve. Al comparar los bits que difieren entre la clave final de Bob y Eve y dividirlo por el tamaño total de la clave final se obtiene un QBER medido. En la búsqueda de encontrar un comportamiento promedio se realizan 10 veces el proceso para cada uno de los QBER iníciales y se promedian. Los resultados del experimento se muestran en la figura 4.

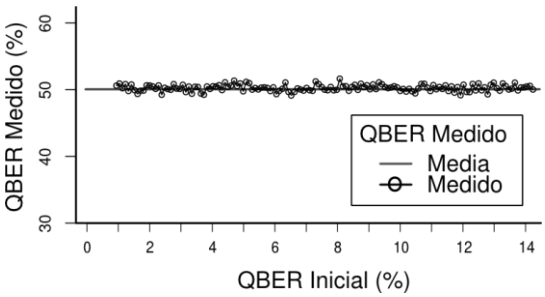

Fig. 4: Clave inicial de 128 bits x 4, con hash de 128 bits. Clave final de 512 bits.

Este resultado, que servirá como línea base de comparación, muestra que para un QBER inicial de entre el 1% y el 14%, es decir en los casos en que Eve dispone entre el 86% y el 99% de la clave, la diferencia entre las claves finales de Bob y Eve difieren en un 50%, hecho atribuible a las propiedades de las funciones hash criptográficas (en este caso MD5 [10]).

Para observar el impacto de amplificar el tamaño de la clave final, se plantea un procedimiento análogo al anterior, generando cuatro claves de 128, pero empleando funciones hash criptográficas que entregan 512 bits a la salida (función SHA-512), de forma que las cuatro claves resultado de la función hash se concatenan obteniendo así una clave final de 2048 bits en Bob y en Eve. Calculando el QBER entre las dos claves finales se obtienen los resultados de la figura 5.

Este resultado indica que las claves finales difieren en un 50% para QBER iníciales de entre 1% y 14%; de forma que la seguridad no se ve comprometida al ampliar el tamaño de la clave. Este resultado es relevante si se

tiene en cuenta que con este simple procedimiento, que consume escasos recursos computacionales, se obtiene una mejora cuatro veces superior en las tasas de bit. Evidentemente, la seguridad de la clave final con este procedimiento depende íntegramente a que el protocolo BB84 asegure hasta este punto una seguridad incondicional de forma inequívoca.

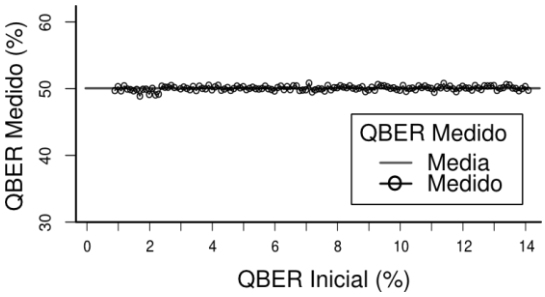

Fig. 5: Clave inicial de 128 bits x 4, con hash de 512 bits. Clave final de 2048 bits.

4.- Conclusión

La seguridad representa un campo en continuo desarrollo, donde nuevas técnicas y procedimientos que emplean las propiedades de la física cuántica se están abriendo camino al brindar la posibilidad de tener una seguridad incondicional. Sin embargo, estas carecen de una tasa de bit equiparable con las tasas de transmisión actuales.

Para lograr mejorar aumentar la tasa de transmisión de claves cuánticas se ha propuesto el uso de las funciones hash criptográficas usadas en la capa de amplificación de la privacidad para lograr ampliar la tasa binaria apoyándose en las propiedades de estas funciones. Así, experimentalmente se han obtenido mejoras en las tasas de bit para la transmisión de claves cuánticas de hasta cuatro veces.

Este trabajo presenta una guía para próximas experimentaciones que usen el proceso de forma iterativa para obtener rendimientos exponenciales de este efecto. Finalmente, en cualquiera de los casos para lograr esta mejora en la tasa de bit sin comprometer la seguridad de la clave se hace necesario que la seguridad incondicional sea asegurada por el protocolo cuántico.

Agradecimientos: Este trabajo ha sido financiado por la ayuda de la Universitat Politècnica de Valencia en la Prueba de Concepto SP20120588 y por Colciencias mediante la beca Francisco José de Caldas.

Referencias

[1] C. H. BENNETT and G. BRASSARD, "Quantum cryptography: plubic key distribution and coin tossing", Proceedings of IEEE International Conference on Computers, Systems and Signal Processing, IEEE Press., pp. 175-179, 1984

[2] C. H. BENNETT, "Quantum Cryptography Using any Two Nonorthogonal States", Phys. Rev. Lett., Vol. 68, No. 21, pp. 3121, 1992.

[3] V. SCARANI, A. ACÍN, G. RIBORDY and N. GISIN, "Quantum Cryptography Protocols Robust against Photon Number Splitting Attacks for Weak Laser Pulse Implementations", Phys. Rev. Lett, Vol. 92, 2004.

[4] V. SCARANI, H. BECHMANN-PASQUINUCCI, N.J. Cerf, N. Lütkenhaus, M. Peev, "The security of practical quantum key distribution", Reviews of modern physics 81, 1301-1310, 2009

[5] G. BRASSARD and L. SALVAIL, "Secret-Key Reconciliation by Public Discussion", Lecture Notes in Computer Science, Vol. 765, pp. 411-423, 1994.

[6] W. BUTTLER, J. TORGERSON, G. NICKEL, C. DONAHUE, & C. PETERSON. Fast, efficient error reconciliation for quantum cryptography. Physical Review A, 67(5), 2003.

[7] C. ELLIOT, A. COLVIN, D. PEARSON, O. PIKALO, J. SCHLAFER and H. YEH, "Current status of the DARPA Quantum Network", BBN Technologies, 2005.

[8] D. PEARSON, "Building a QKD Network out of Theories and Devices", Building the DARPA Quantum Network, BBN Technologies, 2005.

[9] W. STALLINGS, Cryptography and Network Security: Principles and Practice 5th Edition, Pearson, 2011

[10] S. TURNER, L. CHEN, "RFC 6151: Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms", IETF, 2011.