UNIVERSIDAD DE GRANADA

TRABAJO FIN DE GRADO

DOBLE GRADO EN INGENIERÍA INFORMÁTICA Y MATEMÁTICAS

Implementación de una blockchain resistente a ataques criptográficos cuánticos

Subtitulo del Proyecto

Autor
María Victoria Granados Pozo

Director
Gabriel Maciá Fernández
Francisco Javier Lobillo Borrero

Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Facultad de Ciencias

—

Granada, septiembre de 2020

Implementación de una blockchain resistente a ataques criptográficos cuánticos

Subtítulo del proyecto.

Autor
María Victoria Granados Pozo

Director
Gabriel Maciá Fernández
Francisco Javier Lobillo Borrero

Granada, septiembre de 2020

Implementación de una blockchain resistente a ataques criptográficos cuánticos: Subtítulo del proyecto

María Victoria Granados Pozo

Palabras clave: palabra_clave1, palabra_clave2, palabra_clave3, ...

Resumen

Poner aquí el resumen.

Implementation of a blockchain resistant to quantum cryptographic attacks: Project Subtitle

María Victoria Granados Pozo

Keywords: Keyword1, Keyword2, Keyword3, ...

Abstract

Write here the abstract in English.

Yo, **María Victoria Granados Pozo**, alumno de la titulación Doble Grado de Ingeniería Informática y Matemáticas de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación y Facultad de Ciencias de la Universidad de Granada**, con DNI 77137043, autorizo la ubicación de la siguiente copia de mi Trabajo Fin de Grado en la biblioteca del centro para que pueda ser consultada por las personas que lo deseen.

Fdo: María Victoria Granados Pozo

Granada a 7 de septiembre de 2020 .

D. **Gabriel Maciá Fernández**, Profesor del Área de Ingeniería Telemática del Departamento de Teoría de la Señal, Telemática y Comunicaciones de la Universidad de Granada.

D. **Francisco Javier Lobillo Borrero**, Profesor del Área de Matemáticas del Departamento Álgebra de la Universidad de Granada.

Informa:

Que el presente trabajo, titulado ***Implementación de una blockchain resistente a ataques criptográficos cuánticos, Subtítulo del proyecto***, ha sido realizado bajo su supervisión por **María Victoria Granados Pozo**, y autoriza la defensa de dicho trabajo ante el tribunal que corresponda.

Y para que conste, expide y firma el presente informe en Granada a 7 de septiembre de 2020 .

El director:

Gabriel Maciá Fernández **Francisco Javier Lobillo Borrero**

Agradecimientos

Poner aquí agradecimientos...

Índice general

Índice de figuras

Índice de tablas

Listados de código

Capítulo 1

Introducción

El objetivo de este proyecto es evitar que un sistema *blockchain* sea vulnerable a futuros ataques cuánticos. Para ello se ha implementado un algoritmo criptográfico resistente a ordenadores cuánticos, denominado UOV, para la firma de documentos, y posteriormente integrarlo en la *blockchain* ARK.

1.1. Motivación y contexto del proyecto

A lo largo del desarrollo de este trabajo tendremos presentes dos tecnologías, la computación cuántica y las cadenas de bloques.

La computación cuántica constituye un nuevo paradigma de la informática basado en los principios de la teoría cuántica. La computación clásica funciona con bits cuyos valores pueden ser 0 o 1, mientras que la computación cuántica funciona con bit cuánticos o cúbits, donde son una combinación de 0 y 1, pudiendo tomar ambos valores a la vez, esto se denomina la superposición cuántica de los estados [1].

La superposición cuántica aporta gran capacidad de procesamiento, lo que hace posible resolver de manera eficiente problemas de mayor complejidad como la factorización de enteros, el algoritmo discreto y la simulación cuántica, que a día de hoy con los ordenadores clásicos son difíciles de romper.

Otro aspecto importante de la física cuántica relacionado con la superposición es el entrelazamiento de las partículas[2]. Esto es, si dos partículas en algún instante han interactuado retienen un tipo de conexión y pueden entrelazarse formando pares. Esto permite que aunque los cúbits estén separados interactúen entre sí. Con estos dos aspectos la capacidad de procesamiento aumenta considerablemente, cuántos más cúbits la capacidad de procesamiento aumenta considerablemente.

La evolución de la tecnología se ha basado principalmente en la reducción de los transistores para aumentar la velocidad, llegando a escalas de tan

solo algunas decenas de nanómetros. Esto tiene un límite y es la eficiencia, puesto que al seguir disminuyendo el tamaño podrían dejar de funcionar correctamente. De ahí surge la necesidad de descubrir nuevas tecnologías, la computación cuántica [3].

El estudio de las tecnologías cuánticas se inició en 1980, donde surgieron teorías con la posibilidad de realizar cálculos cuánticos. En la década de los 90 se empezó a poner en práctica algunas teoría, apareciendo los primeros algoritmos cuánticos, primeras aplicaciones cuánticas y las primera máquinas diseñadas para realizar cálculos cuánticos.

Por otro lado tenemos las cadenas de bloques o *blockchain*. Esta tecnología permite verificar, validar, rastrear todo tipo de información, ya sean contratos inteligentes, transacciones financieras, certificados digitales o firmas, que será el centro de este proyecto.

Los datos que almacenan cada bloque son transacciones válidas, información referente a ese bloque y la relación con el bloque anterior mediante el *hash*, por tanto el bloque tiene un lugar específico dentro de la cadena. De esta forma si hay una alteración en un determinado bloque se verá reflejado en su *hash* y en el de los bloques posteriores, haciendo que la cadena que la información de la cadena no se pueda perder, modificar o eliminar.

Las aplicaciones de las cadenas de bloques son diversas entre ellas se encuentra la salud o la firma de documentos en las notarías. En el primer caso, cada centro de salud podría tener el historial médico de cualquier paciente, de una forma segura y evitando falsificaciones, estos historiales se encontrarían en nodos distribuidos de forma descentralizada de forma que tuvieran un acceso rápido y seguro. En el segundo caso será en el que nos centraremos a lo largo de este proyecto. Hoy día la firma de documentos o transacciones por parte de un usuario es un problema puesto que se pueden copiar con facilidad, pero con *blockchain* no podrían ser falsificadas debido a la propiedad de validación y rastreo de los datos.

Las *blockchain* por si mismas serían vulnerables a ataques cuánticos, pues su única línea de defensa sería el algoritmo de firma de los bloques. Las cadenas de bloques actualmente son seguras, puesto que un ordenador clásico no tiene la capacidad de cómputo necesaria para descifrar cada bloque, obtener la información y volver a firmar todos los bloques sin dejar huella. Por eso para hacer una *blockchain* resistente es necesario tener un criptosistema que no se pueda romper con computación cuántica, como por ejemplo el algoritmo UOV, ver sección 5.

1.2. Objetivos del proyecto y logros conseguidos

El objetivo de este proyecto es modificar el algoritmo de firma y verificación de las transacciones de la *blockchain* ARK, para hacerla resistente a ataques cuánticos.

- Implementación del algoritmo UOV: Se ha implementado las funciones de generación de claves tanto públicas como privadas, la función de firma a partir de la clave privada y la función de verificación de la misma con la clave pública. Además ha sido necesario implementar la aritmética de cuerpo finito de 2^7 elementos.

- Integrar el algoritmo UOV en la *blockchain* ARK para comprobar su funcionamiento: Se ha modificado el algoritmo de firma dado en la *blockchain* por el algoritmo UOV para aumentar la seguridad.

1.3. Estructura de la memoria

A continuación se muestran los capítulos que presenta la memoria junto con una breve descripción de lo que contiene cada uno.

1. Introducción: Presenta la motivación del proyecto, con una introducción a la computación cuántica y la *blockchain*. También encontramos los objetivos que se persiguen con este trabajos.

2. Planificación y costes: Definición de las entregas y seguimiento del proyecto. Además incluye el presupuesto del proyecto.

3. Análisis del problema: Descripción de las funcionalidades y requisitos, y análisis de los objetivos que se muestran en la sección 1.2.

4. Diseño: En esta sección podemos encontrar el diseño de la implementación del algoritmo UOV y el diseño del ecosistema ARK, donde integraremos el algoritmo.

5. Implementación: Se explicará el código de la implementación del algoritmo UOV y la aritemética implementada para el cuerpo finito de 2^7 elementos.

6. Evaluación y pruebas: Ejemplo de la firma de una transacción en el sistema ARK.

7. Conclusiones

1.4. Contenidos teóricos para la comprensión del proyecto

A continuación se muestran algunos contenidos claves para la mejor comprensión del proyecto.

- Algoritmo cuántico: Son los algoritmos que pueden ser resueltos por un computador cuántico en tiempo polinómico.
- Firma electrónica: Sirve para controlar la integridad de los datos y asegurar que la información procede de quien dice ser su remitente, garantiza que la información que se almacena o se envía no ha sido modificada. Controla la auditoría del documento [4].
- Criptosistemas asimétricos: Cada usuario posee una clave pública y otra privada. El usuario cifrará con la clave privada del usuario y descifrará con la clave pública del usuario que haya mandado el mensaje, que previamente se la habrá mandado a dicho usuario. En algunos criptosistemas se utilizan claves compartidas que se calculan a partir de la clave privada de un usuario A y la clave pública de un usuario B. El algoritmo UOV es de este tipo, es decir, que necesitaremos claves privadas y públicas para firmar y verificar las firmas.
- Resumen o *Hash*: Es el resultado de aplicar una función que transforma un mensaje que longitud variable en uno de longitud fija, denominada función hash. Es el resultado de calcular el resto módulo n con n la longitud fija. Al aplicar la función hash a un fichero, si se modifica algún dato del mismo cambiará su hash y por tanto se sabrá si ha sido manipulado desde que se envió. Así podemos conseguir la integridad del mensaje
- *Blockchain*: Sistemas de almacenamiento de información que se divide en bloque de datos enlazados mediante los hash. A cada bloque se le asocia un hash a partir del bloque anterior, creando una lista enlazada, la búsqueda de información no es muy óptima si hay un número elevado de bloques.

  Blockchain por si mismo no soluciona los problemas de los sistemas de la información y la comunicación, pero permiten impulsar modificaciones orientadas a crear soluciones más robustas, implicando conocer donde hay que usar las *blockchain* y cual es la infraestructura [5].

Capítulo 2

Planificación y costes

Definir claramente de acuerdo con el tutor los "paquetes de trabajo" (PTs), identificando claramente los entregables resultantes de cada uno de ellos. Esto definirá claramente los resultados del proyecto. Pueden usarse Diagramas de Gantt o cualquier herramienta o metodología siempre que facilite la visualización secuencial y dependencias entre los PT. En este mismo capítulo se incluir un presupuesto –ajustado en lo posible a la realidad- que incluya recursos humanos y materiales, así como cualquier dato que determine la viabilidad del proyecto.

Capítulo 3

Análisis del problema

3.1. Especificación de requisitos

Debe incluir una clara descripción de las funcionalidades que se esperan alcanzar, así como las restricciones o condicionantes que puedan determinar el diseño o solución adoptada. Tras eso deben especificarse claramente los requisitos.

Los requisitos pueden ser funcionales (e.g. La herramienta debe mostrar las medidas de la red en tiempo real), o no funcionales (e.g. se debe garantizar el acceso seguro a la herramienta; el rendimiento debe ser alto; el consumo de memoria debe ser bajo; etc.)

3.2. Análisis

El objetivo de este apartado es mostrar en diferentes subapartados los diferentes subproblemas que han aparecido al realizar el proyecto, describiendo las alternativas que se han considerado, y justificando las decisiones que se han adoptado. A veces, especialmente cuando los conceptos utilizados en este apartado son extensos, es necesario clarificarlos previamente en el Capítulo de *Introducción* (sección *Contenidos teóricos para la comprensión del proyecto*).

Capítulo 4

Diseño

Es uno de los capítulos más importantes. Debe explicar claramente la solución propuesta justificando la aproximación adoptada. Este capítulo, según el caso, es aconsejable que defina claramente la arquitectura del sistema propuesto, identificando los roles o partes o actores del sistema. Pueden emplearse metodologías basadas en diagramas de clases, paquetes, diagramas secuenciales, diagramas de relación, etc.

Si se ha diseñado una interfaz gráfica debe también describirse su estructura, dónde se mostrará la información, etc.

Capítulo 5

Implementación

5.1. Cuerpos finitos

Se trabajará con el cuerpo finito de 128 elementos, GF(2^7), que es un cuerpo extendido de GF(2) que corresponde con el cociente

$$\mathrm{GF}(128) = \frac{\mathrm{GF}(2)[x]}{\langle x^7 + x + 1 \rangle} \tag{5.1}$$

Además el orden del cuerpo de las unidades es 127, que es primo entonces todo elemento del cuerpo distinto de 1 es un elemento primitivo, es decir, un generador.

La tabla 5.1 muestra una representación de los elementos no nulos del cuerpo. En la implementación se ha utilizado la representación como cadena de bits, puesto que a la hora de trabajar es más fácil con una cadeba de bits que con los polinomios.

La implementación del cuerpo finito de 2^7 elementos no se ha realizado de forma genérica sino para que sea específica para el algoritmo UOV, de esta forma es mucho más sencillo implementar la aritmética del cuerpo. Para la suma en Z_2 sólo tenemos que fijarnos que es lo mismo que el operador lógico *XOR*, mientras que para el producto, al encontrarmos en un cuerpo como un orden pequeño, se usarán unas tablas que contienen las correspondencias entre los elementos no nulos del cuerpo y sus logaritmos en base a, por lo que el producto se convierte en una suma módulo 127.

5.2. Parámetros y fórmula

Para empezar indicamos los parámetros que serán de utilidad para entender el algoritmo.

- r: Grado del cuerpo extendido, $F_2 \subset F_{2^r}$.

Tabla 5.1: Representación de los elementos no nulos del cuerpo finito de 2^7 elementos

Polinomio	**Bits**	$\log_a$
1	[0, 0, 0, 0, 0, 0, 1]	0
a	[0, 0, 0, 0, 0, 1, 0]	1
a^2	[0, 0, 0, 0, 1, 0, 0]	2
⋮	⋮	⋮
$a^6+a^5+a^4+1$	[1, 1, 1, 0, 0, 0, 1]	124
a^6+a^5+1	[1, 1, 0, 0, 0, 0, 1]	125
a^6+1	[1, 0, 0, 0, 0, 0, 1]	126

- x: Vector de n componentes, denominando a las primeras v componentes $x_1, \cdots, x_v$ vinagre y al resto aceites.
- m: Tamaño de la clave pública, además del número de variables de aceite.
- v: Número de variables vinagre.

$\mathcal{P} : F_{2^r}^n \to F_{2^r}^m$, esta función se puede descomponer como $\mathcal{P} = \mathcal{F} \circ \mathcal{T}$, donde $\mathcal{T} : F_{2^r}^n \to F_{2^r}^n$ es invertible, y $\mathcal{F} : F_{2^r}^n \to F_{2^r}^m$ siendo sus m componentes de la forma:

$$f_k(x) = \sum_{i=1}^{v} \sum_{j=i}^{n} \alpha_{i,j,k} x_i x_j + \sum_{i=1}^{n} \beta_{i,k} x_i \tag{5.2}$$

donde $\alpha_{i,j,k}$ y $\beta_{i,k}$ se toman aleatoriamente en F_2 siendo α una matriz triangular superior. De esta manera será más eficiente y no afectará a la seguridad del algoritmo.

5.3. Generación de la clave privada

La clave privada está formada por $\alpha_{i,j,k}$ y $\beta_{i,k}$ que son valores del cuerpo F_2, tomados de forma aleatoria.

5.4. Generación de la clave pública

Generaremos una clave pública partiendo de la clave privada $\alpha_{i,j,k}$ y $\beta_{i,k}$.

Para entenderlo mejor ponemos las m ecuaciones (5.2) en forma matricial,

$$f_k(x) = x^v \ [\alpha_{i,j,k}] \ (x^v, x^m)' + [\beta_{i,k}] \ (x^v, x^m)' \tag{5.3}$$

siendo $[\alpha_{i,j,k}]$ y $[\beta_{i,k}]$ son las representaciones matriciales de $\alpha_{i,j,k}$ y $\beta_{i,k}$, x^v los vinagres y x^m los aceites, así x se puede expresar como $(x^v, x^m)'$.

Conociendo los valores de la clave privada α y β, tomando de forma aleatoria los del vinagre x^v, los cuales pasaremos a denominarlos como a^v, y tomando los m primeros bits del hash del mensaje h_k podemos generar la clave pública.

Hacemos el cambio de notación $A_k = a^v \ [\alpha_{i,j,k}] = (A_k^v, A_k^m)$, lo sustituimos en la ecuación (5.3) y despejamos los aceites.

$$h_k = A_k^v \ (a^v)' + A_k^m \ (x^m)' + \beta_k^v \ (a^v)' + \beta_k^m \ (x^m)' + \gamma_k \tag{5.4}$$

$$(A_k^m + \beta_k^m)(x^m)' = h_k - (A_k^v + \beta_k^v)(a^v)' - \gamma_k \tag{5.5}$$

$$(x^m)' = (A_k^m + \beta_k^m)^{-1}(h_k - (A_k^v + \beta_k^v)(a^v)' - \gamma_k) \tag{5.6}$$

Si $(A_k^m + \beta_k^m)$ fuese una matriz singular, entonces se tomarían otros valores de vinagres.

Para generar la clave pública necesitamos incluir una nueva matriz T, donde $T \cdot s' = x'$. Incluimos esta matriz T para aumentar la seguridad del algoritmo y así sea más complejo calcular la función inversa $\mathcal{P}$

$$T = \left[\begin{array}{c|c} I_v & T_{vxm} \\ \hline 0 & I_m \end{array}\right] \tag{5.7}$$

Despejando x, obtenemos:

$$x = s \cdot T' = s \left[\begin{array}{c|c} I_v & 0 \\ \hline T'_{vxm} & I_m \end{array}\right] = [s^v, s^m] \left[\begin{array}{c|c} I_v & 0 \\ \hline T'_{vxm} & I_m \end{array}\right] = (s^v + s^m T'_{vxm}, s^m) \tag{5.8}$$

Sustituimos en (5.3):

$$f_k(x) = s \left[\frac{I_v}{T'_{vxm}}\right] [\alpha_{i,j,k}]_{\substack{1<i<v \\ i<j<n}} \ T \ s' + [\beta_{j,k}]_{1<j<n} \ T \ s' \tag{5.9}$$

donde $k \in \{1, ..., m\}$

Así obtenemos las claves públicas definidas para cada k

- $\alpha_{pub_k} = \left[\frac{I_v}{T'_{vxm}}\right] [\alpha_{i,j,k}]_{\substack{1<i<v \\ i<j<n}} \ T$
- $\beta_{pub_k} = [\beta_{j,k}]_{1<j<n} \ T$

5.5. Algoritmo de firma

Por la definición de x obtenemos la firma s como

$$s = x \cdot T'^{-1} \tag{5.10}$$

donde $x = (x^v, x^m)$ con x^v son los vinagres aleatorios y x^m los aceites que hemos calculado en la ecuación (5.6).

5.6. Algoritmo de verificación

Para comprobar que el mensaje es correcto y que no ha sufrido ninguna transformación durante el envío del mismo, se tiene que cumplir la igualdad (5.11).

$$h_k = \alpha_{pub_k}\ s' + \beta_{pub_k}\ s' \tag{5.11}$$

Capítulo 6

Evaluación y pruebas

En este capítulo se debe proporcionar una medida objetiva de las bondades y beneficios de la solución propuesta, tanto en términos absolutos, como –en la medida de lo posible- comparándola con otras soluciones. Dependiendo del tipo de proyecto, debe incluir los resultados experimentales obtenidos al probar la solución; también puede incluir una tabla o diagrama de los costes reales del desarrollo, para así establecer conclusiones respecto a la planificación y costes estimados a priori. Finalmente, cuando se trata del desarrollo de una aplicación software, se pueden definir baterías de pruebas a realizar, de modo que en este capítulo se especificarán qué pruebas se han realizado, los resultados esperados y los resultados obtenidos.

Capítulo 7

Conclusiones

Capítulo en el que deben resumirse las principales aportaciones del trabajo realizado.

7.1. Valoración personal

Se puede incluir una valoración personal del proyecto (opcionalmente)

Bibliografía

[1] A. M. y Jose Ignacio Escribano, "La computación cuántica y el "futuro de la criptografía": la criptografía post-cuántica," 2020, https://www.bbvanexttechnologies.com/la-computacion-cuantica-y-el-futuro-de-la-criptografia-la-criptografia-post-cuantica/.

[2] A. Banafa, "Computación cuántica y blockchain mitos y realidades," 2019, https://www.bbvaopenmind.com/tecnologia/mundo-digital/computacion-cuantica-y-blockchain-mitos-y-realidades/.

[3] "Computación cuántica wikipedia," https://es.wikipedia.org/wiki/Computaci%C3%B3n_cu%C3%A1ntica.

[4] "Firma digital," https://www.cert.fnmt.es/content/pages_std/html/tutoriales/tuto7.htm.

[5] C. Pastorino, "Blockchain: qué es, cómo funciona y cómo se está usando en el mercado," 2018, https://www.welivesecurity.com/la-es/2018/09/04/blockchain-que-es-como-funciona-y-como-se-esta-usando-en-el-mercado/.

Siglas

GCD Greatest Common Divisor.

GMF Gabriel Maciá Fernández.

LCM Least Common Multiple.

SVM support vector machine.

Apéndice A

Manual de usuario