
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

**1. Introducción** **1**
1.1. Motivación y contexto del proyecto . . . . . 1
1.2. Objetivos del proyecto y logros conseguidos . . . . . 5
1.3. Estructura de la memoria . . . . . 5
1.4. Contenidos teóricos para la comprensión del proyecto . . . . . 6
1.4.1. Computación cuántica . . . . . 6
1.4.2. Blockchain . . . . . 7
1.4.3. Algoritmo UOV . . . . . 10

**2. Planificación y costes** **15**

**3. Análisis del problema** **17**
3.1. Especificación de requisitos . . . . . 17
3.2. Análisis . . . . . 17

**4. Diseño** **19**

**5. Implementación** **21**

**6. Evaluación y pruebas** **23**

**7. Conclusiones** **25**
7.1. Valoración personal . . . . . 25

**Bibliografía** **28**

**Glosario de siglas** **29**

**A. Manual de usuario** **31**

# Índice de figuras

1.1. Comparativa de la capacidad de cómputo de un ordenador clásico con un ordenador cuántico[1] . . . . . 3
1.2. Comparativa de la capacidad de cómputo de un ordenador clásico con un ordenador cuántico[1] . . . . . 6

# Índice de tablas

1.1. Niveles de seguridad de ordenadores clásicos y cuánticos [2] . . . . 4
1.2. Representación de los elementos no nulos del cuerpo finito de $2^7$ elementos . . . . . 11

# 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

La tecnología ha transformado nuestra sociedad en una sociedad digitalizada, donde actualmente, los dispositivos digitales comportan la mayor parte de nuestras actividades diarias en distintos ámbitos, como económicas, organizativas o sociales. En el proceso de digitalización de la sociedad podemos distinguir las siguientes cinco fases [3].

La primera fase o era del Internet, corresponde a mediados de los 90. En esta fase se comenzaron a crear páginas web para que los medios de comunicación y las empresas puedieran publicar y compartir información.

La segunda fase o era de las redes sociales, tuvo mayor auge a partir de 2005. Plataformas de bajo o ningún coste, se utilizaban en las empresas para poder llegar mejor a los clientes.

La tercera fase o era de la economía colaborativa, nació con la crisis de 2008 cuando las empresas tenían pocos recursos. Surgieron plataformas para conectar a las personas, y poder obtener lo que necesitasen unas de otras. Por ejemplo pagos online, ver recomendaciones y reseñas de un alojamiento o pedir un taxi. Además se da un gran paso ya que estas aplicaciones pasan de estar alojadas en ordenadores a teléfonos inteligentes.

La cuarta fase o era del mundo autónomo, se ha desarrollado durante décadas. Se desarrollan tecnología con inteligencia artificial, es decir, que simulan la inteligencia de los humanos para poder resolver problemas más complejos.

Quinta fase o era del bienestar moderno, comienza con las pulseras inteligentes como *Fitbit* o *Fuelband* de Nike. Estas pulseras son el impulso de la tec-

nología para facilitar la vida de los clientes y poder integrar la tecnología en la vida de los mismos.

La digitalización debe de venir acompañada de mecanismos que aporten seguridad a los datos. Los pilares de la seguridad de la información son los conocidos como la tríada CIA (confidencialidad, integridad y disponibilidad)[4].

La **confidencialidad** es la propiedad que impide que la información pueda ser accesible por entidades no autorizadas. Un sistema garantiza la confidencialidad cuando un tercero entra en posesión de la información intercambiada entre el remitente y el destinatario, no es capaz de extraer ningún contenido legible. Para asegurar la confidencialidad se utilizan mecanismos de cifrado y ocultación de la comunicación.

La **integridad** busca mantener la exactitud de los datos, es decir, que no hayan sido modificados durante su envío. La integridad se obtiene adjuntando al mensaje otro conjunto de datos de comprobación de la integridad, un ejemplo es la firma digital.

La **disponibilidad** es la cualidad de la información de encontrarse a disposición de quienes deben acceder a ella, ya sean personas, procesos o aplicaciones, en el momentos que así lo quieran. Los mecanismos para asegurar la disponibilidad se implementan con la infraestructura tecnológica.

Además de estos tres pilares hay otro principio, la **autentificación**, que es la propiedad que permite identificar al generador de la información. Trata de comprobar si un mensaje enviado por un usuario, ha sido verdaderamente firmado por él mismo. Esto se consigue con el uso de cuentas de usuario y contraseñas de acceso.

Para garantizar estos servicios de seguridad se hace uso de protocolos de seguridad de la información entre los que se encuentra la criptografía, la lógica y la autenticación.

La criptografía se ocupa de cifrar ciertos mensaje con el fin de hacerlos ilegibles a receptores no autorizados, una vez que llega a su destino y sea descifrado, el receptor obtendrá el mensaje original [5]. Además dota de seguridad a las comunicaciones, a la información y a las entidades que se comunican.

Podemos diferenciar dos tipos de criptografía, la criptografía simétrica y la asimétrica. La criptografía simétrica utiliza la misma clave para cifrar y descrifrar un mensaje, esta clave la ha de conocer tanto el emisor como el receptor. Mientras que la asimétrica utiliza dos claves la pública y la privada.

En la criptografía asimétrica podemos diferenciar dos ramas, el cifrado de clave pública y las firmas digitales [6]. En el cifrado de clave pública, el emisor cifra el mensaje con la clave pública del destinatario y el receptor lo descifra con su propia clave privada. En las firmas digitales, el emisor firma el mensaje con su

clave privada y el receptor puede verificar el mensaje con su propia clave pública, además cualquier manipulación del mensaje se refleja en su resumen o *hash*.

Este tipo de criptografía basa su seguridad en la hipótesis de que no se pueden encontrar las claves por fuerza bruta con la tecnología existente en la actualidad. Los ataque de fuerza bruta tratan de recuperar las claves probando todas las posibles combinaciones hasta encontrar la que permite el acceso, a partir del algoritmo de cifrado y del texto cifrado con su original [7]. Para que la búsqueda tenga éxito se deberán de realizar $10^n - 1$ operaciones donde $n$ es la longitud de la clave.

Otro factor importante, en la seguridad, es si en la clave aparencen números, caracteres o la combinación de ambos, aumentando así el coste de encontrar las claves, llegando a alcanzar tiempos de cálculo logarítmicos, es decir, que podrían tardar siglos en encontrar una contraseña compleja pero también depende de la capacidad de operación del ordenador.

En este contexto, la aparición de la futura computación cuántica permitirá el cálculo de operaciones a una velocidad mucho mayor. En la gráfica 1.1 podemos observar la capacidad de cómputo del peor ordenador cuántico, con la línea continua, que sigue la gráfica de una función exponencial, frente a la capacidad del mejor ordenador clásico, la línea discontinua, que sigue una función lineal.

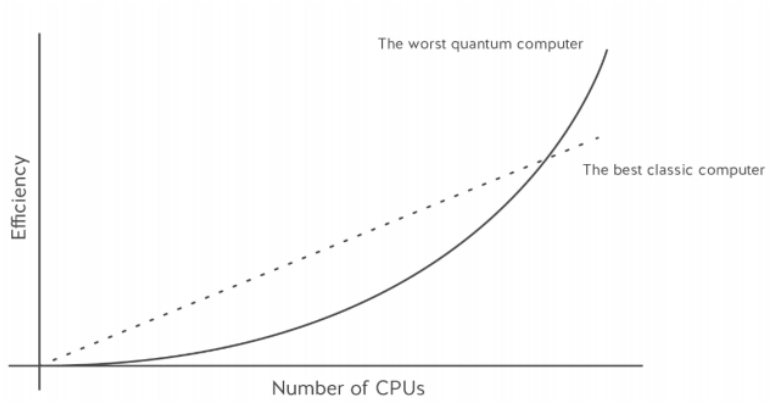


Figura 1.1: Comparativa de la capacidad de cómputo de un ordenador clásico con un ordenador cuántico[1]

La comparativa también nos muestra que para operaciones pequeñas como, por ejemplo, editar un documento de texto un ordenador cuántico sería probablemente ineficienciente. Por tanto lo mejor sería un ordenador híbrido, que mezclase computación clásica, para cálculos pequeños, y computación cuántica, para operaciones de mayor tamaño.

Cuando esté desarrollado el ordenador cuántico no serán válidos los ac-

tuales algoritmos criptográficos de clave pública, como RSA, Diffie-Hellman y ECDSA, ya que se basan en los problemas del logaritmo discreto y factorización de enteros, resolubles fácilmente por un ordenador cuántico. Las primeras ideas de la criptografía cuántica se tiene en los años 70, destacando los algoritmos de Shor y Grover.

Veamos la tabla comparativa 1.1, esta nos indica el tipo de algoritmo criptográfico, el algoritmo con la longitud de la clave y a continuación el nivel de seguridad tanto en un ordenador clásico como en uno cuántico. El nivel de seguridad de un algoritmo nos indica el número de operaciones necesarias para romper dicho algoritmo, por ejemplo, si tiene un nivel de seguridad $n$ entonces se requieren $2^n$ operaciones para romper el algoritmo [8]. Observamos que hay una diferencia considerable en los niveles de seguridad de los algoritmo asimétricos, puesto que con un ordenador clásico al menos se necesitan $2^{112}$ operaciones mientras que con computación cuántica solo una.

| Tipo | Algoritmo-Longitud clave | Nivel seguridad (ordenador clásico) | Nivel seguridad (ordenador cuántico) | Ataque cuántico |
|---|---|---|---|---|
| Asimétrico | RSA-2048 | 112 | 0 | Algoritmo de Shor |
| | RSA-3072 | 128 | 0 | Algoritmo de Shor |
| | ECC-521 | 128 | 0 | Algoritmo de Shor |
| | ECC-521 | 256 | 0 | Algoritmo de Shor |
| Simétrico | AES-128 | 128 | 64 | Algoritmo de Grover |
| | AES-256 | 256 | 128 | Algoritmo de Grover |

Tabla 1.1: Niveles de seguridad de ordenadores clásicos y cuánticos [2]

En la actualidad, se están desarrollando muchos algoritmos para que sean resistentes a ataques de tipo cuántico, denominados algoritmos de criptografía postcuántica [9]. Estos ataques afectan principalmente a los algoritmos de clave pública o asimétrica, puesto que para la criptografía simétrica duplicar el tamaño de clave empleada es suficiente para hacerlos seguros y hacer inservible el algoritmo de Grover.

Por otro lado, también en la actualidad está siendo muy relevante la adopción de las *blockchain* como tecnología para ofrecer diversos servicios. Las *blockchain* o cadenas de bloques son listas de transacciones, denominadas bloques, firmadas y unidas con algoritmos criptográficos. Además cada bloque contiene el hash del bloque anterior, se explicará con más detalle en la sección 1.4.2.

Esta tecnología se ha integrado en diferentes áreas, donde resalta el uso en los servicios financieros o criptomonedas, que aumenta la eficiencia y disminuye los costes. Otro uso de las *blockchain* es en las cadenas de suministro, algunos restaurantes, como Fogo de Chão [10], están empezando a utilizar las *blockchain* para poder rastrear el origen de sus alimentos hasta llegar al propio restaurante,

una gran ventaja para encontrar fácilmente si hay algún producto contaminado o en mal estado.

Un ejemplo del uso de las *blockchain* queda reflejado en este proyecto en el que se ha implementado un algoritmo resistente a ataques cuánticos, UOV [11] y se ha adaptado a la *blockchain* ARK [12] para que se utilice dicho algoritmo de firma.

## 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 y el contexto en el que surge, además incluye una breve reseña introduciendo las dos tecnologías que se han utilizado computación cuántica y la *blockchain*, aparte de la explicación matemática del algoritmo utilizado. En este capitulo también se encuentran los objetivos que se persiguen con este trabajo.
2. Planificación y costes: Contiene el diagrama de Gantt con la defición de las entregas y seguimiento del proyecto, así como 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: Se encuentra el diseño de la implementación del algoritmo UOV y el diseño del ecosistema ARK, donde se integrará el algoritmo de firma.

5. Implementación: Contiene la explicación de la implementación del algoritmo UOV y la aritemética del 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

En los siguientes apartados se explica los contenidos claves de este proyecto, que son la computación cuántica 1.4.1, la tecnología *blockchain* 1.4.2 y el algoritmo UOV 1.4.3.

### 1.4.1. Computación cuántica

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 [13]. La figura 1.2 muestra

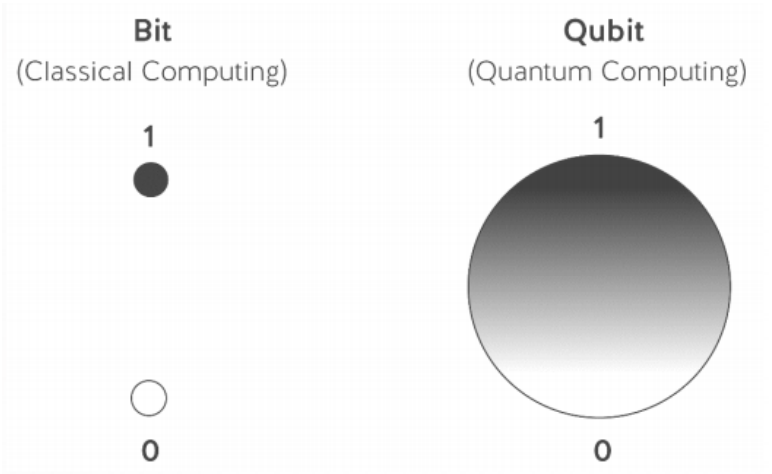


Figura 1.2: Comparativa de la capacidad de cómputo de un ordenador clásico con un ordenador cuántico[1]

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[14]. 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 [15].

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.

### 1.4.2. Blockchain

*Blockchain* es un 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. Para la búsqueda eficiente en *blockchain* se usan los árboles merkle.

Los datos que almacena 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 información de la cadena no se pueda perder, modificar o eliminar.

Los árboles merkle [16] son una estructura de datos en árbol en el que cada nodo que no es hoja está etiquetado con el *hash* que surge de la combinación de los valores o etiquetas de sus nodos hijo. Esta estructura permite que aunque los datos estén separados puedan ser ligados a un único valor de *hash*, el *hash* del nodo raíz del árbol. El *hash* de este nodo va firmado para asegurar la integridad y hacer que la verificación sea fiable.

De esta forma se asegura que los datos son recibidos sin daños y sin ser alterados, además permite que los datos puedan ser entregados por partes, ya que un nodo puede obtener solo la cabecera de un bloque desde una fuente y otra

pequeña parte del árbol desde otra fuente, y poder asegurar que los datos son correctos. Esto funciona porque si un usuario intenta hacer un cambio en una transacción falsa en la parte inferior del árbol en seguida se verá reflejado en la parte superior del árbol, es decir, en el nodo raíz.

La idea de la tecnología *blockchain* surge a comienzos de 1991 cuando los científicos Stuart Haber y W. Scott Stornetta introducen una solución computacional para la firma de documentos digitales y que no puedieran ser modificados con el tiempo. Usaron cadenas de bloque para almecenar los documentos con sello de tiempo y en 1992 se incorporaron los árboles Merkle, que podían recopilar varios documentos en un bloque haciendo el diseño más eficiente. Sin embargo, esta tecnología no se utilizó y la patente caducó en 2004 [17].

En 1998, Nick Szabo trabaja en una moneda digital descentralizada, "bit gold". Dos años después Stefan Konst publica su teoría sobre la seguridad criptográfica en las cadenas de bloques junto con algunas ideas de implementación [18].

En 2004, el informático y criptógrafo Harold Thomas Finney introdujo el sistema RPoW (prueba de trabajo reutilizable). El sistema se basa en *HashCash* pero los token de prueba no están ligados a una aplicación sino que pueden ser gastados libremente como una moneda. Los clientes pueden crear tokens e intercambiarlos sin necesidad de regenerarlos [19]. RPoW resolvió el problema del doble gasto registrando los tokens en un servidor fiable diseñado para permitir a los usuarios verificar su exactitud e integridad en tiempo real. Este sistema puede considerarse como un prototipo de las criptomonedas.

A finales de 2008, un grupo de desarrolladores bajo el nombre de Satoshi Nakamoto publican un documento técnico en que se establece un modelo para *blockchain*. Está basado en el algoritmo RPoW pero en lugar de usar dicho hardware, se utiliza un protocolo descentralizado peer-to-peer para verificar y restrear las transacciones. En otras palablas los "mineros" extraen bitcoins para obtener una recompensa mediantes pruebas de trabajo y posteriormente los nodos los verifican. Bitcoin nació el 3 de enero de 2009 cuando Satoshi Nakamoto extrajo el primer bloque de bitcoin con una recompensa de 50 bitcoins. Y el 12 de enero de 2009 tuvo lugar la primera transacción entre Satoshi Nakamoto y Hal Finney que obtuvo 10 bitcoins.

A partir de 2014, se comienzan a explorar el potencial de las cadenas de bloque y a buscar otras aplicaciones fuera de su uso en las transacciones financieras. Ethereum introduce programas informáticos que se ejecutan en la *blockchain*, se pueden utilizar para realizar una transacción cumpliendo ciertas condiciones como los contratos inteligentes.

Un contrato inteligenete se tratan de contratos que tienen la capacidad de cumplirse de forma automática. Un contrato inteligente está constituido por un protocolo de códigos que permiten a un dispositivo ejecutar de forma automatizada las sentencias previamente programadas, prescindiendo de la intervención humana [20].

Además de los contratos inteligentes, Ethereum tiene su propia criptomoneda llamada Ether, se puede transferir entre cuentas y se utiliza para pagar las tarifas por la ejecución de los contratos inteligentes.

Actualmente las *blockchain* tiene otros usos más allá de las criptomonedas.

Las cadenas de bloques o *blockchain* permiten verificar, validar, rastrear todo tipo de información, ya sean contratos inteligentes, transacciones financieras, certificados digitales o firmas [21], siendo estas últimas el centro de este trabajo. También permiten impulsar modificaciones orientadas a crear soluciones más robustas, por ejemplo en centros de salud o notarías que se explicarán más adelante.

Las *blockchain* son vulnerables a futuros ataques cuánticos ya que su única línea de defensa es el algoritmo de firma de los bloques. Aunque, actualmente las cadenas de bloques 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 apartado 1.4.3.

Los tres pilares de la tecnología *blockchain* son la descentralización, transparencia e inmutabilidad [22].

Un sistema centralizado almacena todos los datos en una misma entidad y habría que interactuar con la misma para obtener la información necesaria. Un ejemplo de un sistema centralizado son los bancos que almacenan todo el dinero y la única forma de pagar a alguien es a través de un banco. Es similar a la arquitectura cliente-servidor donde los clientes se comunican entre ellos mediante el servidor. Pero tener un único sitio para almacenar todos los datos es vulnerable a los ataques, informáticos, por otra parte si el nodo central se corrompe o tiene una actualización, los datos será incorrectos o no se podrán acceder a ellos. De los contras de los sistemas centralizados surge la idea de los sistemas **descentralizados**, la información no la tiene un único nodo sino que todos los usuarios son dueños de la información. La principal ideología de las *blockchain* es poder interactuar usuario con usuario sin tener que pasar por un tercero.

El concepto **transparencia** se refiere a la transparencia de los datos no de las identidades. Esto es la identidad de la persona se oculta a través de la criptografía y lo único que se ve es su dirección pública, pero podemos ver todas las transacciones que se han realizado en su dirección pública. En el historial de transacciones no vemos “Antonio envió 1BTC” sino que aparece “1MF1bhsFLkBzzz9vpFYEmvwT2TbyCt7NZJ envió un 1BTC”. Este nivel de transparencia nunca antes había existido en el sistema financiero, lo que exige más responsabilidad a las grandes empresas. De la misma forma podemos

trasladar este concepto fuera del sistema financiero por ejemplo a las cadenas de suministro, y saber exactamente de donde provienen los alimentos de un restaurante.

La **inmutabilidad** en el contexto de las cadenas de bloques significa que una vez introducida una transacción en la *blockchain* ya no se puede alterar. De esta forma aplicando esta tecnología a los bancos se evitarían casos de malversación de fondos. Esta propiedad se obtiene gracias a la función criptográfica *hash*.

La función *hash* es el resultado de aplicar una función que transforma un mensaje de longitud variable en uno de longitud fija. Esto es 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, consiguiendo la integridad del mensaje.

De la misma forma si hay un cambio en una de las transacciones de un bloque se reflejará en el *hash* del bloque, afectado a todos los bloques anteriores. Así si el atacante quiere preservar la integridad deberá de modificar todos los bloques siendo una tarea imposible. De esta forma se obtiene la inmutabilidad de los datos.

Hoy en día, la tecnología *blockchain* está ganando mucha atención, no limitandose solo al uso en las criptomonedas. Así las cadenas de bloques tienen diversas aplicaciones 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 así se obtendría un acceso rápido y seguro. 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.

### 1.4.3. Algoritmo UOV

#### Cuerpos finitos

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

$$\mathrm{GF}(128) = \frac{\mathrm{GF}(2)[x]}{\langle x^7 + x + 1 \rangle} \tag{1.1}$$

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 1.2 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.

Tabla 1.2: 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 |

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 $\mathbb{Z}_2$ 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.

### Parámetros y fórmula

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

- r: Grado del cuerpo extendido, $\mathbb{F}_2 \subset \mathbb{F}_{2^r}$.
- $x$: 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.

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

forma:

$$f_k(x) = \sum_{i=1}^{v} \sum_{j=i}^{n} \alpha_{i,j,k} x_i x_j + \sum_{i=1}^{n} \beta_{i,k} x_i \tag{1.2}$$

donde $\alpha_{i,j,k}$ y $\beta_{i,k}$ se toman aleatoriamente en $\mathbb{F}_2$ siendo $\alpha$ una matriz triangular superior. De esta manera será más eficiente y no afectará a la seguridad del algoritmo.

**Generación de la clave privada**

La clave privada está formada por $\alpha_{i,j,k}$ y $\beta_{i,k}$ que son valores del cuerpo $\mathbb{F}_2$, elegidos de forma aleatoria.

**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 (1.2) en forma matricial,

$$f_k(x) = x^v \, [\alpha_{i,j,k}] \, (x^v, x^m)' + [\beta_{i,k}] \, (x^v, x^m)' \tag{1.3}$$

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 $\alpha$ y $\beta$, tomando de forma aleatoria los del vinagre $x^v$, los cuales pasaremos a denominarlos como $a^v$, y cogiendo 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 (1.3) y despejamos los aceites.

$$h_k = A_k^v \, (a^v)' + A_k^m \, (x^m)' + \beta_k^v \, (a^v)' + \beta_k^m \, (x^m)' + \gamma_k \tag{1.4}$$

$$(A_k^m + \beta_k^m)(x^m)' = h_k - (A_k^v + \beta_k^v)(a^v)' - \gamma_k \tag{1.5}$$

$$(x^m)' = (A_k^m + \beta_k^m)^{-1}(h_k - (A_k^v + \beta_k^v)(a^v)' - \gamma_k) \tag{1.6}$$

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 $\mathscr{P}$

$$T = \left[\begin{array}{c|c} I_v & T_{vxm} \\ \hline 0 & I_m \end{array}\right] \tag{1.7}$$

Despejando x, obtenemos:

$$x = s \cdot T' = s \left[\begin{array}{c|c} I_v & 0 \\ \hline T'_{vxm} & I_m \end{array}\right] = [s^v, s^m] \left[\begin{array}{c|c} I_v & 0 \\ \hline T'_{vxm} & I_m \end{array}\right] = (s^v + s^m T'_{vxm}, s^m) \tag{1.8}$$

Sustituimos en (1.3):

$$f_k(x) = s\left[\frac{I_v}{T'_{vxm}}\right][\alpha_{i,j,k}]_{\substack{1<i<v \\ i<j<n}} T\ s' + [\beta_{j,k}]_{1<j<n}\ T\ s' \tag{1.9}$$

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

Así obtenemos las claves públicas definidas para cada $k$

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

**Algoritmo de firma**

Por la definición de $x$ obtenemos la firma $s$ como

$$s = x \cdot T'^{-1} \tag{1.10}$$

donde $x = (x^v, x^m)$ con $x^v$ son los vinagres aleatorios y $x^m$ los aceites que hemos calculado en la ecuación (1.6).

**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 (1.11).

$$h_k = \alpha_{pub_k}\ s' + \beta_{pub_k}\ s' \tag{1.11}$$

# 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

# 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] "Qilimanjaro: Next computing generation, quantum computation at your fingertips," 2018, https://neironix.io/documents/whitepaper/4514/whitepaper.pdf.

[2] L. Wilson, "The Rise of Quantum Computers – The Current State of Cryptographic Affairs," 2016, https://www.activecyber.net/rise-quantum-computers-current-state-cryptographic-affairs/.

[3] J. V. of Jeremiah Owyang, "Roadmap: Five Phases of Digital Eras," 2019, https://web-strategist.com/blog/2019/01/04/roadmap-five-phases-of-digital-eras/.

[4] Wikipedia, "Seguridad de la información," https://es.wikipedia.org/wiki/Seguridad_de_la_informaci%C3%B3n.

[5] Wikipedia, "Criptografía," https://es.wikipedia.org/wiki/Criptograf%C3%ADa.

[6] Wikipedia, "Criptografía Asimétrica," https://es.wikipedia.org/wiki/Criptograf%C3%ADa_asim%C3%A9trica.

[7] Wikipedia, "Atáque de fuerza bruta," https://es.wikipedia.org/wiki/Ataque_de_fuerza_bruta.

[8] Wikipedia, "Security Level," https://en.wikipedia.org/wiki/Security_level.

[9] Wikipedia, "Criptografía postcuántica," https://es.wikipedia.org/wiki/Criptograf%C3%ADa_postcu%C3%A1ntica.

[10] Wikipedia, "Restaurante Fogo de Chão," https://en.wikipedia.org/wiki/Fogo_de_Ch%C3%A3o.

[11] A. Szepieniec, B. Preneel, F. Vercaunteren, and W. Beullens, "LUOV: Signature Scheme proposal for NIST PQC Project (Round 2 version)," 2018, https://github.com/WardBeullens/LUOV/blob/master/Supporting_Documentation/luov.pdf.

[12] "Ark," https://ark.io/.

[13] A. Muñoz and J. I. 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/.

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

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

[16] Wikipedia, "Árboles Merkle," https://es.wikipedia.org/wiki/%C3%81rbol_de_Merkle.

[17] "History of Blockchain," https://academy.binance.com/blockchain/history-of-blockchain.

[18] "A brief history of Blockchain," https://www.icaew.com/technical/technology/blockchain/blockchain-articles/what-is-blockchain/history.

[19] Wikipedia, "Reusable Proof of Work," https://es.wikipedia.org/wiki/Reusable_Proof_Of_Work.

[20] "Contrato inteligente," https://elderecho.com/los-contratos-inteligentes-smart-contracts-contratos-inteligentes.

[21] 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/.

[22] A. Rosic, "What is blockchain Technology? A Step-by-Step Guide For Beginners," https://blockgeeks.com/guides/what-is-blockchain-technology/#The_Three_Pillars_of_Blockchain_Technology.

# Siglas

**ECDSA** Elliptic Curve Digital Signature Algorithm.

**RSA** Rivest, Shamir y Adleman.

## Apéndice A

# Manual de usuario