UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO

FACULTAD DE INGENIERÍA
DOCTORADO EN CIENCIAS DE LA INGENIERÍA

"UN MODELO MULTICAPAS BASADO EN BLOCKCHAIN QUE FORTALECE LA INTEGRIDAD Y SEGURIDAD DE INFORMACIÓN PÚBLICA"

TESIS

PARA OBTENER EL GRADO DE:
DOCTOR EN CIENCIAS DE LA INGENIERÍA

PRESENTA:
M.C.I. FERNANDO REBOLLAR CASTELAN

DIRECTOR DE TESIS:
DR. MARCO ANTONIO RAMOS CORCHADO

CO-DIRECTORA:
DRA. ROCÍO ALEJANDRA ALDECO PÉREZ

MÉXICO 2021

Siempre recordare este fantástico viaje Doctoral lleno de emociones y experiencias que aportaron en mi manera de ver la vida, en este tiempo me case, viví una pandemia y me convertí en padre entre muchas otras cosas. Admito que muchas veces me desviaba de mi tema para leer avances de temas de la física que me emocionaban y de tantas cosas que aún no podemos explicar, un fenómeno cuántico que me exalta es el hecho de que un simple observador puede influir en los comportamientos y partiendo de ese hecho, si cualquier observador hubiere podido influir en la realización de este trabajo, ni que decir de mis tutores, asesores y profesores académicos, de mi familia, amigos y conocidos, aunque sea difícil de creer, todos me han aportado más de un elemento y solo puedo decirles una palabra, gracias.

Es gracias a todos ustedes que es posible el presente trabajo.

Y quien sabe, tal vez, incluso tu, que estas leyendo esto también influiste...

Resumen

Los servicios digitales son utilizados por la población en general, el acceso al Internet y eventos como las pandemias han acelerado la aparición y uso de nuevos servicios digitales. Los gobiernos también han incrementado sus servicios digitales, sin embargo, estos servicios digitales aun carecen de suficiente seguridad e integridad en la información, los datos que recolectan no siempre son bien protegidos y en el caso de los servicios de gobiernos estos carecen de transparencia. Blockchain contiene un conjunto de técnicas criptográficas que permiten descentralizar y aumentar la integridad en la información que maneja, pero aun contiene diversas desventajas en cuanto a la eficiencia que lo hacen incapaz de ser implementado en algunos servicios digitales donde se requiere una alta tasa de transacciones por segundo. En esta tesis se propone un modelo multicapa basado en blockchain, con el objetivo de que dicho modelo permita adecuar su funcionamiento para almacenar la información de los servicios digitales, proporcionando una mayor transparencia e integridad en la información que los servicios actuales y que al mismo tiempo se obtenga una eficiencia razonable para que se viable una futura implementación. Para lograr esas metas el modelo divide la información en 4 tipos y la separa en el mismo numero de capas, cada capa se especializa en procesar su tipo de información y tiene propiedades únicas que le permiten mantener e incluso incrementar la integridad de la información contenida conservando una alta eficiencia en las transacciones. La propuesta también permite que nodos voluntarios puedan participar para contribuir a descentralizar la información y hacerla más segura. Finalmente se concluye que el modelo multicapa basado en blockchain presenta un marco para los servicios gubernamentales que permitirán ofrecer servicios digitales seguros, auditables, transparentes y en consecuencia confiables. Una vez realizada la propuesta se le realizó un análisis estadístico a la capas y se formalizaron algunas de sus propiedades como lo es la velocidad de generación de bloques, los niveles de descentralización variable que tiene y los tamaños de bloques, obteniendo un modelo flexible y capaz adaptarse a diferentes tipos de servicios digitales, garantizando la transparencia y la integridad.

Abstract

Digital services are used by the general population, Internet access and events such as pandemics have accelerated the emergence and use of new digital services. Governments have also increased their digital services, however, these digital services still lack sufficient security and integrity of information, the data they collect are not always well protected and in the case of government services they lack transparency. Blockchain contains a set of cryptographic techniques that allow decentralizing and increasing the integrity of the information it handles, but it still contains several disadvantages in terms of efficiency that make it unable to be implemented in some digital services where a high rate of transactions per second is required. In this thesis a multilayer model based on blockchain is proposed, with the objective that this model allows to adapt its operation to store the information of digital services, providing greater transparency and integrity in the information than the current services and at the same time obtaining a reasonable efficiency so that a future implementation is viable. To achieve these goals the model divides the information into 4 types and separates it into the same number of layers, each layer specializes in processing its type of information and has unique properties that allow it to maintain and even increase the integrity of the information contained while maintaining a high efficiency in transactions. The proposal also allows voluntary nodes to participate to contribute to decentralize the information and make it more secure. Finally, it is concluded that the multilayer model based on blockchain presents a framework for government services that will allow to offer secure, auditable, transparent and consequently reliable digital services. Once the proposal was made, a statistical analysis of the layers was performed and some of its properties were formalized, such as the speed of block generation, the variable decentralization levels and block sizes, obtaining a flexible model capable of adapting to different types of digital services, ensuring transparency and integrity.

Índice general

Pág.

1. Introducción **1**
- 1.1. Hipótesis . . . 5
- 1.2. Justificación . . . 5
- 1.3. Objetivos de la tesis . . . 5
- 1.4. Estructura de la tesis . . . 6

2. Marco teórico **7**
- 2.1. Sistemas centralizados, distribuidos y descentralizados . . . 7
  - 2.1.1. Sistemas centralizados . . . 7
  - 2.1.2. Sistemas distribuidos . . . 9
  - 2.1.3. Sistemas descentralizados . . . 10
- 2.2. Criptografía . . . 11
  - 2.2.1. Criptografía simétrica . . . 12
  - 2.2.2. Criptografía asimétrica o de clave pública . . . 13
- 2.3. Criptomonedas . . . 14
  - 2.3.1. Orígenes de las criptomonedas . . . 15
  - 2.3.2. Principales criptomonedas . . . 16
- 2.4. Blockchain . . . 19
  - 2.4.1. Funcionamiento del blockchain . . . 20
  - 2.4.2. Tipos de blockchain . . . 22
  - 2.4.3. Retos del blockchain . . . 24
- 2.5. Modelos de consenso . . . 28
  - 2.5.1. El dilema de los generales bizantinos . . . 29
  - 2.5.2. Proof of work (Pow) . . . 30
  - 2.5.3. Proof of stake (PoS) . . . 31
  - 2.5.4. Delegate Proof of stake (DPoS) . . . 32
  - 2.5.5. Practical Byzantine Fault Tolerance (PBFT) . . . 32

3. Estado del Arte **35**
3.1. Trabajos Relacionados 35
3.2. Comparativa 39

4. Propuesta: Un blockchain multicapas **41**
4.1. Un modelo de 4 capas basado en blockchain 42
4.1.1. Capa 1: Index-Keys 43
4.1.2. Capa 2: Transactions 44
4.1.3. Capa 3: SmartContracts 44
4.1.4. Capa 4: Files 45
4.2. Estructura de los bloques 47
4.2.1. Flujos de información 49

5. Resultados **55**
5.1. Validación de integridad de los bloques 55
5.2. Modelado de la propuesta 60
5.3. Arquitectura general de flujos de información 66
5.4. Caso de estudio: Un sistema de infracciones en un blockchain multicapa 71
5.4.1. Posible forma de implementar el caso de estudio 75
5.4.2. Consideraciones al implementar el caso de estudio 76

6. Conclusiones y trabajo futuro **79**
6.1. Conclusiones 79
6.2. Trabajo futuro 80

7. Artículos, ponencias y estancia de investigación **83**
7.1. Artículos 83
7.2. Ponencias 84
7.3. Estancia de investigación 84

Referencias **85**

Índice de tablas

Pág.

2.1. Top 10 de criptomonedas con mejor capitalización de mercado junio 2019 (valores en USD) obtenido de coinmarketcap.com 16
2.2. Comparación entre blockchain público y blockchain privado 24

3.1. Comparación de los trabajos relacionados 40

4.1. Propiedades de la capa 1 43
4.2. Propiedades de la capa 2 44
4.3. Propiedades de la capa 3 45
4.4. Propiedades de la capa 4 46

5.1. Probabilidad de éxito de un atacante disminuyendo conforme se van generando más bloques en la cadena 57

Índice de figuras

Pág.

2.1. Esquema lógico de los sistemas centralizados. 8
2.2. Esquema lógico de los sistemas distribuidos. 9
2.3. Esquema lógico de un sistemas completamente descentralizado. . . 10
2.4. Bloques con los datos generales del blockchain [1], se resalta que cada bloque contiene el hash valido del bloque previo. 22
2.5. Atributos de los blockchain [2] 26

3.1. Capas propuestas por MOAC [3] 38

4.1. Esquema general del modelo 42
4.2. Descripción del algoritmo de consenso por cada capa 46
4.3. Pirámide que representa el nivel de descentralización de cada capa de la propuesta 47
4.4. Pirámide que representa el nivel de tamaño de los bloques de cada capa de la propuesta 47
4.5. Bloque conector que conecta su bloque anterior y agrega la conexión a una capa distinta a la suya 48
4.6. Esquema general de una posible generación de bloques en las distintas capas 49
4.7. Información que procesa cada capa, donde la capa 1 es la superior y la capa 4 es la inferior 50
4.8. Flujo 1: se utilizan las primeras dos capas 50
4.9. Flujo 2: se utilizan las capas 1, 2 y 3 51
4.10. Flujo 3: se utilizan las capas 1, 2 y 4 52
4.11. Flujo 4: se utilizan las 4 capas 52
4.12. Esquema de contenido general de los bloques 53

5.1. Tendencia exponencial a la baja conforme se generan más bloques 58
5.2. Caso 1: Llaves e índices y transacciones 67
5.3. Caso 2: Llaves e índices, transacciones y smartcontracts 68

5.4. Caso 3: Llaves e índices, transacciones y archivos 69
5.5. Caso 4: Llaves e índices, transacciones, smartcontracts y archivos 70
5.6. Propuesta de distribución de nodos en las 4 capas 72
5.7. Interacciones entre capas 73
5.8. Ejemplo de flujo de la información utilizando las 4 capas del caso de estudio en un sistema de infracciones 74
5.9. Esquema de lenguajes, algoritmos de consenso y frameworks para una posible implementación 75

Capítulo 1

Introducción

El aumento del acceso a Internet lo ha convertido en un servicio cotidiano y de fácil uso para la mayoría de la población, permitiendo que la cantidad de dispositivos electrónicos que se conectan a la red vaya en aumento constante así como el incremento en su funcionalidad [4]. Un ejemplo claro de lo anterior son los celulares que se convirtieron en celulares inteligentes, después las televisiones pasaron a ser televisiones inteligentes, lo mismo está sucediendo con los autos y la tendencia va en aumento con muchos otros aparatos electrónicos y mecánicos.

El termino Internet de las cosas (IoT por *Internet of Things* en inglés) hace referencia a este tipo de dispositivos [5]. La combinación de estos han permitido generar conceptos como casas, edificios e incluso ya se habla de ciudades inteligentes [6]. Estos estarán dotados de una diversidad de sensores repartidos por todas partes que estarán generando una gran cantidad de información, misma que necesitará ser transferida por la red de manera segura para garantizar que sea completa, objetiva, confiable y confidencial.

En el IoT se generan grandes cantidades de datos en tiempo real (no todos confiables), y con frecuencia tienen implicaciones de privacidad y seguridad [7], es por ello que es necesario contar con tecnologías que garanticen la integridad de los datos que van a circular por dicha red.

El aumento de dispositivos conectados a la red también ha permitido un aumento en la cantidad de servicios que se ofrecen en línea. Estos servicios digitales se han ido popularizando y distintas organizaciones las han impulsado. Por ejemplo, google y facebook han atraído a los anunciantes que anteriormente se anunciaban por periódicos, revistas y libros amarillos, aumentando las ganancias de los primeros y disminuyendo las ganancias de los segundos en pocos años. Uber ha conectado a personas con la necesidad de transportarse con conductores particulares afectando a una gran cantidad de taxistas. Netflix a perjudicado a los videoclubs hasta casi extinguirlos. Airbnb conecta a huéspedes y anfitriones perjudicando a los hoteles de las zonas turísticas, entre muchos otros ejemplos

[8].

Una gran cantidad de estos servicios digitales únicamente fungen de intermediarios lo cual resulta muy lucrativo puesto que cobran comisiones por únicamente conectar proveedores de productos o servicios con clientes, sin mencionar que también lucran con los datos de todas las personas que interactúan en sus sistemas. Durante este tiempo se han modificado políticas y reglamentos al paralelo del crecimiento de los servicios digitales, debido a la enorme cantidad de información que pasa por los administradores de los servicios digitales. Las modificaciones han llevado a regular y proteger los datos personales de los usuarios y tipificar diversos crímenes cibernéticos [9]. Estas eventualidades generaron desconfianza entre los usuarios de los servicios digitales, y poco a poco se ha buscado que estos servicios tengan suficiente seguridad y generen confianza entre los usuarios.

Los gobiernos de múltiples países han actuado en consecuencia al aumento de servicios digitales regulando algunos e implementado sus propios servicios digitales que ofrecen a sus contribuyentes, con el propósito de agilizar tramites y disminuir costos [10]. A su vez los ciudadanos exigen a sus gobiernos cada vez más transparencia en el uso de sus recursos [11]. Por todo ello se requiere que la información que es manejada por los diversos servicios digitales sea segura para volverse confiable. Por lo que estos servicios también requieren de un aumento de confianza ante la población.

Las ventajas de las cadenas de bloques ya han sido reconocidas por los gobiernos de diferentes países, algunos de ellos ya han puesto en marcha proyectos para incorporar las cadenas de bloques en los servicios digitales, sin embargo, también han encontrado dificultades para funcionar de la mejor manera [12].

Blockchain surge de la necesidad de generar confianza entre los usuarios que utilizan los servicios digitales. Debido a que blockchain permite funcionar sin una entidad centralizada y a que permite a múltiples verificadores voluntarios aportar a que la información se segura y verificable. Esta propuesta tiene su origen un 31 de octubre del 2008, donde Satoshi Nakamoto [1] propuso un sistema distribuido y políticamente descentralizado para un sistema de envió de dinero mediante tokens digitales, que utilizando un conjunto de técnicas criptográficas es capaz de validar las transacciones entre los usuarios a través de los mismos nodos de la red, dichos nodos se conectan voluntariamente. Pero no fue hasta el 3 de enero del 2009 que se generó el primer token, naciendo con este el Bitcoin (nombre que se le dio al token) y a la tecnología distribuida posteriormente se le nombró blockchain, debido a que cada determinado tiempo generaba un bloque sellado criptográficamente que era unido a un bloque anterior [13].

Estas transacciones con tokens digitales no dependen de ninguna institución, hacen el envió del tokens directo entre cliente y vendedor sin necesitar entidades bancarias. A partir de su popularización, el Bitcoin fue obteniendo la atención de distintos grupos de personas con conocimientos en tecnologías que permi-

tió mejorar su implementación y funcionalidad. Posteriormente, surgieron nuevas criptomonedas con distintas versiones de mejoras a la tecnología blockchain, por ejemplo, Ripple (XRP) que redujo la cantidad de tiempo necesario para que se confirmara una transacción quedando en el orden de los segundos, gracias a su algoritmo de consenso con la solidez frente a las fallas bizantinas [14].

Otra criptomoneda que surgió después es Ethereum (ETH), que implementa una mejora de blockchain la cual permite utilizar contratos inteligentes. Los cuales son programas que ejecutan acuerdos registrados entre dos o más intermediarios, que éstos se comprometen a cubrir una serie de especificaciones condicionadas por los mismos, y la tecnología de blockchain comprueba automáticamente con un alto grado de confiabilidad [15].

La tecnología blockchain no es exclusiva de las criptomonedas puede emplearse en otros casos de uso ajenos al del dinero, por ejemplo en 2015 Zyskind et. al. en [16] presentaron una propuesta donde utilizaban el blockchain para la protección de datos personales y confidenciales. Otro uso es el presentado en [17], [18], [19], [20] donde proponen sistemas de voto electrónico basando en blockchain donde una vez emitido el voto es imposible de cambiar y permite realizar el conteo de votantes y votados con total seguridad de que no hubo fraude. Si alguien trata de cambiar uno o varios votos ya emitidos el bloque se convierte en un bloque inválido. Utilizando este mismo enfoque se puede utilizar para registrar certificados, cédulas y demás documentos que suelen ser susceptibles a falsificaciones. Los bloques válidos son registrados en una blockchain y los falsos de ninguna manera se pueden agregar. En el libro "Blockchain: Blueprint for a new economy"[13] se describen una gran variedad de las aplicaciones que puede tener el blockchain.

En [21] J. Leon et. al. afirman que "Blockchain está a punto de convertirse en la invención más emocionante después de Internet", sin embargo, destacan que la tecnología blockchain todavía se encuentra en una etapa de desarrollo y se necesita más investigación para mejorar su eficiencia y seguridad. A demás, advierten que los investigadores se enfrentan a muchas oportunidades y desafíos para hacer que blockchain sea exitoso, debido a sus distintas desventajas que se mencionan a continuación.

En caso de que se quiera distribuir sin restricciones, los nodos que comprueban las transacciones necesitan ser recompensados por el mismo trabajo criptográfico que realizan (los llamados mineros), lo que genera un coste por transacción el cual aumenta cuando incrementa la demanda de transacciones. Dado que blockchain requiere un proceso de verificación muy estricto para crear un nuevo registro de transacciones, esto también hace que cuando se supera la capacidad de comprobación de transacciones surja una cola de transacciones a confirmarse, lo que vuelve lento el proceso de confirmaciones y se presenta un desperdicio de recursos computacionales porque distintos equipos trabajan para generar el siguiente bloque en la cadena pero solo uno (el que termina primero) lo agrega, es decir

presenta problemas de escalabilidad [22].

Otro problema surge cuando no existe la suficiente distribución en el blockchain. Si un nodo concentra el 51 % (es decir más de la mitad de poder de procesamiento) dicho nodo puede dominar todos los demás nodos, manipulando los registros en una cadena de bloques [23]. Además, la privacidad y la confidencialidad siguen siendo problemas con blockchain, porque todos los nodos de la cadena de bloques tienen acceso a todos los datos, aunque no puedan modificarlos [24], esto también depende de la aplicación que se le quiera dar. Debido a estos problemas surgen nichos de oportunidad de mejora como bien lo comentaba J. Leon en [21] la tecnología necesita mejorar ya que sus problemas dificultan su utilidad en el Internet de las cosas y en diversos servicios digitales.

Para mejorar la escalabilidad en blockchain han surgido diversas propuestas de distintos grados de modificación al blockchain original. Los primeros buscaban resolver el problema sin modificar el funcionamiento del blockchain, uno de esos fue aumentar el tamaño del bloque un determinado porcentaje [25] pero el problema se solucionaba momentáneamente, después se propuso eliminar el limite del tamaño de bloque [26], pero eso generaría bloques de gran tamaño y solo empresas con grandes cantidades de almacenamiento podrían ser mineros y eso haría que se centralizara la red blockchain.

Por otra parte surgieron modificaciones mayores, una de ellas es la implementada en [15] donde un nodo no distribuido en la red no necesariamente tiene que almacenar todo el blockchain completo sino que podría almacenar solo una parte y trabajar con esa. Para esto, se modificó el algoritmo de consenso para que pudiese funcionar con este cambio, lo que permitió aumentar el número de transacciones a unas 15 transacciones por segundo. Otra modificación fue por parte de [14] que optimizó el algoritmo de consenso de los datos distribuidos, dando prioridad o peso a los nodos de confianza. En esta implementación se superan las transacciones por segundo a 1500, pero termina centralizando en los nodos de confianza, las validaciones de las transacciones y el problema del que el 51 % de nodos maliciosos controlen la red se facilita para los nodos de confianza donde si quisieran estos nodos podrían modificar bloques de la cadena.

Existen otras modificaciones similares a las mencionadas, en general se observan entre una y otra que se pueden obtener dos de los siguientes tres atributos: seguridad, descentralización y escalabilidad. Se puede configurar el blockchain para que sea escalable y seguro pero esto termina centralizando la red, o se puede configurar para que sea escalable y descentralizado pero se tendrá que bajar la seguridad para permitir la escalabilidad y el último caso que fue la primera versión del blockchain una red segura y descentralizada pero con serios problemas de escalabilidad.

1.1. Hipótesis

Generar una nueva propuesta basada en blockchain que funcione mediante múltiples capas con diferentes algoritmos de consenso, permitirá adecuar su funcionamiento para almacenar información pública de gobiernos y que pueda ser verificada por los ciudadanos para asegurar su integridad y seguridad aumentando su confiabilidad.

1.2. Justificación

Los nuevos servicios digitales paulatinamente van disminuyendo a intermediarios que aumentan el costo de los bienes y servicios que requieren usuarios, empresas y organizaciones, sin que una tercera parte esté involucrada en los procesos. Con los avances tecnológicos y la llegada del blockchain ha surgido una revolución de cambios en los servicios digitales. Primero inició con la aparición de las criptomonedas que tienen características del dinero físico, pero siendo completamente digital. Después aparecieron los contratos inteligentes que aumentan la funcionalidad y hacen realidad automatizaciones que anteriormente no eran posibles.

Nuevas propuestas basadas en blockchain aumentan la funcionalidad de los procesos que requieren diversos servicios digitales y hacen posible aumentar la confianza en la información que se maneja en línea y como consecuencia la transparencia en la información y la fiabilidad de la misma. Sin embargo, en otras ocasiones se requiere eficiencia y confidencialidad por lo que se requiere de un modelo que contemple ambos aspectos que por si solos son incompatibles. Esto en respuesta a la necesidad del continuo incremento en la información y su uso transaccional en la web.

1.3. Objetivos de la tesis

El principal objetivo de la tesis es generar un modelo multicapas basado en blockchain que permita adecuar su funcionamiento para almacenar información de servicios digitales, donde la transparencia de la información se una cualidad, como lo es la información pública de gobiernos y permita que los ciudadanos participen para asegurar su integridad y seguridad, aumentando su confiabilidad. De la cual se derivan los siguientes objetivos específicos:

- Realizar un análisis de los algoritmos criptográficos que utiliza el blockchain así como sus protocolos para revisar el funcionamiento identificando las fortalezas y debilidades de cada uno de ellos.
- Proponer un modelo multicapas basado en blockchain con adecuaciones necesarias para que tenga la funcionalidad de que los ciudadanos puedan contribuir a verificar la información que sea publicada por gobiernos, contribuyendo a hacer la segura y confiable. Sin sacrificar la eficiencia y confidencialidad de la misma.
- Validar experimentalmente la propuesta para verificar su funcionamiento, considerando costes por transacción y grado de seguridad.

1.4. Estructura de la tesis

La tesis cuenta con 6 capítulos, este primer capítulo es una breve introducción al trabajo realizado. El segundo capítulo contiene el marco teórico con conceptos relacionados con la investigación. El tercer capítulo contiene los trabajos relacionados. El cuarto capítulo describe la propuesta de investigación. El quinto capítulo muestra los resultados obtenidos al realizar las pruebas a la propuesta. El sexto capítulo contiene las conclusiones de la presente investigación así como el trabajo a futuro.

Capítulo 2

Marco teórico

En este capítulo se describen los conceptos necesarios que fundamentan la investigación.

2.1. Sistemas centralizados, distribuidos y descentralizados

Los sistemas son parte de nuestra vida cotidiana y todo el tiempo interactuamos con diversos tipos de sistemas aunque en ocasiones no lo parezca, es por ello que es importante recordar la definición de sistema, "Es un conjunto de cosas que relacionadas entre sí ordenadamente contribuyen a determinado objeto"[27]. Si se analiza un poco la definición, existen múltiples sistemas y también se puede notar que muchos de estos sistemas funcionan mediante la unión de sistemas mas pequeños. Por ejemplo, en una escuela se pueden identificar diversos sistemas, como el aula de clases, los profesores, los padres de familia, etc.

En el área de las tecnologías de la información y comunicación (TICs) sucede lo mismo muchos sistemas están compuestos por sistemas más pequeños. Dentro de estos sistemas se pueden identificar tres tipos: sistemas centralizados, sistemas descentralizados y sistemas distribuidos.

2.1.1. Sistemas centralizados

Los sistemas centralizados fueron los primeros en existir, siguen la arquitectura cliente-servidor, son los más simples y consisten en un dispositivo central al que

se conectan todos los demás dispositivos para interactuar entre sí. En la Figura 2.1 se puede observar como lógicamente se comunican los dispositivos donde el único requisito es utilizar el mismo o mismos protocolos de comunicación para concretar los intercambios de información [28].

Figura 2.1: Esquema lógico de los sistemas centralizados.

Los sistemas centralizados tienen las siguientes características principales [28]:

- Son menos costosos.
- Requieren de un mantenimiento fácil y rápido.
- Implementación simple al ser pocos complejos.
- Utilizan protocolos de respaldo de información simples.
- Son vulnerables a más tipos de ataques.
- Tienen mayor riesgo de perdida de servicio.
- Tienen poca tolerancia a fallas.
- Cuentan con seguridad arquitectónica mínima.

2.1.2. Sistemas distribuidos

Los sistemas distribuidos surgen después de los sistemas centralizados como una mejora a la seguridad y el aumento de disponibilidad del servicio. A continuación un par de definiciones de sistema distribuido: "Un sistema distribuido es una colección de computadoras independientes que dan la apariencia al usuario de ser una computadora única"[28]. "Un sistema en el cual componentes conectados a través de una red de computadoras se comunican y coordinan sus acciones mediante el intercambio de mensajes"[29]. En la Figura 2.2 se observa un ejemplo de como se configura un red distribuida en tres nodos, donde uno o dos de ellos pueden fallar y el tercero mantener el servicio.

Figura 2.2: Esquema lógico de los sistemas distribuidos.

Los sistemas distribuidos tienen las siguientes características principales [29]:

- Tienen concurrencia de componentes.
- Requieren de un mantenimiento más complejo en comparación a los sistemas centralizados.
- Su implementación puede ser escalada.

- Tiene independencia de fallos en sus componentes.
- Cuentan con mayor disponibilidad del servicio.
- Son capaces de tolerar un mayor número de fallas.
- Tiene seguridad arquitectónica.

2.1.3. Sistemas descentralizados

Los sistemas descentralizados buscan eliminar cualquier nodo central. A diferencia de los sistemas distribuidos que mantienen una configuración similar a los sistemas centralizados pero con distintos nodos centrales. Los sistemas descentralizados completamente, permiten que cualquier dispositivo conectado a la red pueda ofrecer el servicio continuamente o momentáneamente en caso de una o varias fallas. En la Figura 2.3 se observa un ejemplo de una red completamente descentralizada.

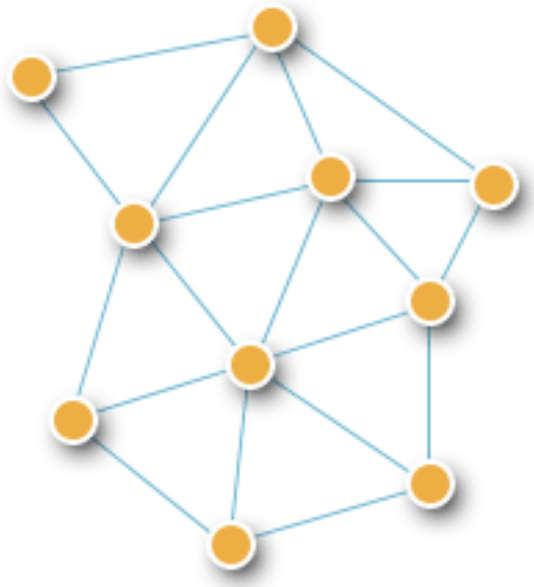

Figura 2.3: Esquema lógico de un sistemas completamente descentralizado.

Con respecto a la diferenciación de los sistemas distribuidos y descentralizados existen definiciones que solapan ambos conceptos, para esta tesis es importante diferenciar entre uno y el otro. En este sentido, un sistema distribuido no es los mismo que un sistema descentralizado. Según Vitalik Buterin en [30], "distribuido

significa que no todo el procesamiento de las transacciones se hace en el mismo lugar, mientras que descentralizado significa que no hay una única entidad que tenga control sobre todo el procesamiento". Para entender esto es importante definir algunos de los tipos de sistemas descentralizados de la misma manera que en [30]:

- Descentralización arquitectónica: Se presenta cuando los equipos físicos que dan un determinado servicio manejan protocolos que en caso de fallas controlan cuantas computadoras se puede permitir el sistema perder, sin que el mismo sistema se colapse.
- Descentralización política: Se refiere al número de individuos y/o organizaciones que controlan los equipos de los que se conforma el sistema. La descentralización política es parte clave para distinguir quien es el dueño o dueños del sistema. También puede que no pertenezca a nadie, tal como sucede con el Internet.
- Descentralización lógica: Se refiere a un sistema que si fuese cortado a la mitad (incluidos proveedores y usuarios) él mismo podría funcionar como dos partes completamente funcionales.

2.2. Criptografía

La criptografía es un área de estudio que se encarga de profundizar en el diseño de protocolos, algoritmos, y sistemas que una vez probados y validados forman parte de la seguridad en diferentes medios tecnológicos como las comunicaciones digitales [31]. La criptografía tiene como tareas principales diseñar, implementar y hacer uso de algoritmos matemáticos que pudieran servir para mejorar la seguridad digital. Por tanto el tipo de propiedades de las que se ocupa la criptografía son [32]:

- **Confidencialidad**: Garantiza que la información sea accesible únicamente a personal autorizado. Para conseguirlo utiliza códigos y técnicas de cifrado.
- **Autenticación**: Es cuando se buscan mecanismos que permiten verificar a los implicados en la comunicación, para lograrlo se debe conocer la identidad del comunicador. Algunos de los algoritmos que permiten hacerlo son la función hash criptográfica MAC o protocolo de conocimiento cero.
- **Integridad**: Garantiza que la información contiene el mensaje exacto cuando fue emitida y cuando fue recibida o consultada. Para conseguirlo puede

usar funciones hash criptográficas como MD5, SHA256 entre otras como protocolos de compromiso de bit, o protocolos de notarización electrónica.

- **Vinculación**: Consiste en vincular un archivo o transacción a una entidad como lo puede ser una organización o una persona. En el caso de ambos, el objetivo es asegurar su conformidad respecto a un archivo o transacción de tal manera que pueda entenderse que la vinculación adiciona el entendimiento de sus implicaciones, obligaciones y/o derechos. También la vinculación hace referencia el el término "No repudio", aunque el termino esta dejándose de utilizar, ya que implica conceptos jurídicos que la tecnología por sí sola no puede resolver, sin embargo, el no repudio pretende proporcionar protección frente a que alguna de las entidades implicadas en la comunicación no pudiera negar haber participado en toda o parte de la comunicación. La firma digital en las transacciones en los archivos es utilizada para mitigarlo. En algunos casos la firma digital también es utilizada para negar que se ha intervenido en la comunicación, por ejemplo cuando se hace uso de un servicio de mensajería instantánea y se pretende que no se pueda demostrar esa comunicación, para estos caso se pueden utilizar técnicas como el cifrado negable.

La criptografía agrupa conjuntos de funcionalidades, para las cuales se utilizan dos tipos de criptografías: la criptografía simétrica y la criptografía asimétrica.

2.2.1. Criptografía simétrica

La criptografía simétrica, también conocida como criptografía de mensaje secreto, es un método criptográfico que en su funcionamiento utiliza una misma clave para cifrar y descifrar los mensajes del emisor y del receptor. Ambas partes que se comunican y deben ponerse de acuerdo de antemano sobre el mensaje secreto o la clave a usar. Cuando ambas partes tienen acceso al mensaje secreto o clave, el emisor cifra un mensaje usando la clave, lo envía a su destino, y éste lo descifra con la misma clave [33].

La seguridad de la criptografía simétrica recae en la clave y no en el algoritmo. Es decir, de nada sirve que el atacante conozca el algoritmo usado para generar los mensajes con la clave, únicamente si el atacante encontrara la clave usada, le serviría conocer el algoritmo. Los algoritmos de cifrado ampliamente utilizados tienen estas propiedades por ejemplo: AES (Advanced Encryption Standard) [34].

2.2.2. Criptografía asimétrica o de clave pública

La criptografía de clave pública es un método que permite aumentar la seguridad y agrega funcionalidad cuando se realiza un intercambio de mensajes, la forma en que funciona es utilizado un par de claves para el envío de mensajes. Ambas claves pertenecen a una misma persona que recibirá el mensaje. Una clave es privada y el dueño de la misma debe guardarla de modo que nadie tenga acceso a ella, la otra clave es pública y se debe entregar a toda persona o entidad con la que se desee entablar una comunicación segura. Además, por su naturaleza de generación de claves se garantiza que ese par de claves sólo se puede generar una vez, de esta manera se puede asumir que no es posible que dos entidades puedan obtener la misma pareja de claves [33].

Cuando una persona o entidad que posee su par de claves, utiliza su clave privada para cifrar un mensaje, cualquiera que tenga su clave pública podrá descifrar el mensaje. De esta manera se consigue la identificación y autenticación del remitente, porque el único que pudo haber cifrado el mensaje fue el poseedor de la clave privada. La vulnerabilidad surge cuando un tercero obtiene la llave privada de otra persona, porque podría hacerse pasar dicha persona. La criptografía simétrica es el motor de la firma electrónica, y se asume que el firmante es efectivamente el dueño de la clave privada es por ello que debe guardar su clave privada y no compartirla con nadie.

La criptografía asimétrica también se encarga eliminar el problema del intercambio de claves en los sistemas de cifrado simétricos. En el caso de la criptografía asimétrica no es necesario que el emisor y el receptor tengan que ponerse de acuerdo en la clave a usar. Lo único que deben tener antes de iniciar el intercambio de mensajes seguros, es que cada uno debe obtener clave pública del otro y evitar que su llave privada termine en posesión de otra entidad. También es posible enviar la llave publica a más entidades y intercambiar mensajes seguros con varias entidades usando las mismas claves.

Las dos principales ramas de la criptografía asimétrica son:

- **Firmas digitales**: Se utiliza para garantizar la autenticidad del mensaje y es un mensaje firmado con la clave privada del remitente y puede ser verificado por todas las entidades que tengan acceso a la clave pública del remitente, lo que permite comprobar que este remitente tenía acceso a la clave privada y en consecuencia asumir que la entidad esta asociada con la clave pública utilizada. De esta manera se asegura que el mensaje no ha sido modificado, ya que cualquier modificación del mensaje arrojaría un distinto resultado del algoritmo de resumen del mensaje (encoded message digest).
- **Cifrado de clave pública**: Su función es garantizar la confidencialidad

del mensaje y consiste en un mensaje cifrado con una clave pública de una entidad que no puede ser descifrado por ninguna entidad (incluyendo a la entidad que cifro el mensaje), por lo tanto solo el poseedor de la clave privada correspondiente puede descifrar el mensaje. presumiblemente su propietario y la persona asociada con la clave pública utilizada.

Una analogía con el cifrado de clave pública es la de un buzón con una ranura de correo. La ranura de correo está expuesta y accesible al público; su ubicación (la dirección de la calle) es, en esencia, la clave pública. Alguien que conozca la dirección de la calle puede ir a la puerta y colocar un mensaje escrito a través de la ranura; sin embargo, sólo la persona que posee la llave (clave privada) puede abrir el buzón de correo y leer el mensaje.

Una analogía para las firmas digitales es el marcado de un mensaje con una marca única como la de un sello personal. El mensaje puede ser visto por cualquier entidad, pero el sello identifica al remitente. Por ejemplo el algoritmo criptográfico asimétrico RSA (Rivest, Shamir y Adleman) [35].

2.3. Criptomonedas

Una criptomoneda, criptodivisa o criptoactivo (cryptocurrencys en inglés) es un medio digital que permite ser intercambiado y que es imposible de falsificar, duplicar o copiar. Utiliza criptografía fuerte para asegurar las transacciones, administrar la creación de unidades adicionales y verificar el actual poseedor de la misma [36].

Bitcoin fue la primer criptomoneda en existir con estas características, sin embargo, al ser de código abierto, en poco tiempo surgieron más criptomonedas con diferentes modificaciones. Algunas con pocos cambios con respecto a la original otras con cambios más drásticos.

Las criptomonedas en general buscan transferir digitalmente valor a un bajo costo, es decir permitir transacciones por medios digitales sin importar la zona geográfica y sin intermediarios pero garantizando la integridad y seguridad de la transacción. La seguridad de una criptomoneda puede ser vulnerada y es matemáticamente posible, pero el costo para lograrlo sería tan alto que se vuelve completamente inviable. Por ejemplo, un atacante que intentase vulnerar el sistema de bitcoin necesitaría una potencia computacional mayor que el de toda la red en su conjunto, y aun así, solo tendría una probabilidad de éxito del 50 %.

Las principales ventajas de las criptomonedas son [13]:

1. Reducir o eliminar el coste de la transacción.

2. Eliminar intermediarios (punto clave para que ocurra el punto anterior).
3. Reducir el tiempo de confirmación de la transacción.
4. Operar sin la necesidad de agentes financieros.
5. Agregar funcionalidades que permitan la automatización de transacciones.

Según Jan Lansky en [37] un sistema de criptomonedas debe cumplir las siguientes seis condiciones:

1. "El sistema no necesita una autoridad central. Su estado es mantenido a través de un consenso distribuido".
2. "El sistema mantiene todas las unidades y su propietario".
3. "El sistema define si se pueden crear nuevas unidades. En este caso, el sistema debe definir las circunstancias de su origen y cómo determinar el propietario de las nuevas unidades".
4. "Sólo se puede asegurar la propiedad de una unidad a un usuario de manera criptográfica".
5. "El sistema permite las transacciones de unidades, en las cuales se cambia el propietario de dichas unidades. Una transacción solo puede ser efectuada si se puede probar el actual propietario de estas unidades".
6. "Si se efectúan dos transacciones sobre las mismas unidades, el sistema solo ejecuta una de ellas".

2.3.1. Orígenes de las criptomonedas

En 1981 en [38] David L. Chaum propuso una técnica que mediante algoritmos criptográficos de claves publicas era capaz de mantener el anonimato en correos electrónicos que impedían rastrear los correos desde su origen y mantenían oculta la identidad de los emisores. Posteriormente en 1983 en [39] proponía un sistema de pagos electrónicos mediante firmas ciegas para que los pagos fueran no rastreables. En 1983 en [40] agrega avances en su investigación y propone un modelo de seguridad sin la necesidad de identificación de los emisores en sistemas de transacciones digitales. En 1988 en [41] propone un sistema de dinero electrónico imposible de rastrear. Estos primeras aportaciones permitieron sembrar las bases para el funcionamiento de las tarjetas de crédito actuales.

En 1996 la NSA (National Security Agency de los EUA) en [42] describió un modelo de dinero electrónico que permitía enviar y recibir pagos manteniendo el anonimato mediante el uso de algoritmos criptográficos, con la peculiaridad de mantener dinero electrónico fuera de linea.

Hasta este momento las propuestas eran todas centralizadas y atacaban pero no resolvían completamente el problema del doble gasto [43]. Que consiste en evitar gastarse dos veces el dinero electrónico. Este problema no pasa con el dinero físico porque se entrega la moneda o billete al prestador del servicio o producto pero sin embargo en el dinero digital es un problema que actualmente manejan los bancos.

En 2002 en [44] Adam Black propuso un sistema para evitar el spam en correos electrónicos mediante una "proof of work", que se utilizo como base para lograr un consenso exitoso posteriormente en bitcoin.

En 2008 se propone el bitcoin en [1], siendo considerada la primer criptomoneda que retoma el pago electrónico anónimo pero a diferencia de las anteriores esta es completamente descentralizada.

En orden cronológico surgen varias criptomonedas pero eran copias del bitcoin y no tenían diferencias y no se popularizaron ni mejoraron. Actualmente existen cientos de criptomonedas, sin embargo las que ofrecen mejoras se popularizaron y en la Tabla 2.1 se describen las 10 más populares de acuerdo a su capitalización del mercado al día que se realiza el actual análisis (10/06/2019).

#	Nombre	Capitalización del mercado	Precio	Volumen (24h)	Tokens en circulación
1	Bitcoin	$141,164,625,821	$7,953.30	$18,920,178,049	17,751,787 BTC
2	Ethereum	$26,116,547,040	$244.72	$8,247,022,195	106,435,188 ETH
3	XRP	$16,824,832,779	$0.3975	$1,599,940,608	42,238,947,941 XRP
4	Litecoin	$7,883,446,732	$126.87	$5,399,120,871	62,166,626 LTC
5	Bitcoin Cash	$6,995,770,895	$393.14	$1,568,041,384	17,830,538 BCH
6	EOS	$5,866,483,308	$6.40	$2,319,704,794	918,566,268 EOS
7	Binance coin	$4,499,940,737	$31.86	$407,334,038	141,175,490 BNB
8	Bitcoin SV	$3,354,837,274	$188.20	$520,176,083	17,828,336 BSV
9	Tether	$3,299,753,465	$1.01	$19,351,556,865	3,276,289,280 USDT
10	Stellar	$2,383,225,500	$0.1232	$370,093,460	19,331,690,111 XLM

Tabla 2.1: Top 10 de criptomonedas con mejor capitalización de mercado junio 2019 (valores en USD) obtenido de coinmarketcap.com

2.3.2. Principales criptomonedas

Descripción del top 10 de criptomonedas más populares:

1. **Bitcoin**: se abrevia (BTC) y es una red de consenso que permite un nuevo sistema de pago y una moneda completamente digital. Desarrollado por sus usuarios, es una red de pago de igual a igual que no requiere una autoridad central para operar. El 31 de octubre de 2008, un individuo o grupo de personas que operan bajo el seudónimo "Satoshi Nakamoto"publicaron el artículo de Bitcoin y lo describieron como: "una versión de dinero electrónico exclusivamente de igual a igual, que permitiría el envío de pagos en línea directamente de una parte a otra sin pasar por una institución financiera"[1].

2. **Ethereum**: se abrevia (ETH) y es una red similar a la de BTC pero introduce el termino de contrato inteligente que permite a los desarrolladores crear aplicaciones descentralizadas (dapps) conceptualizadas por Vitalik Buterin en 2013. ETH es la moneda nativa de la plataforma Ethereum y también funciona como las tarifas de transacción para los mineros en la red Ethereum. Ethereum es el pionero de los contratos inteligentes basados en blockchain. Cuando se ejecuta en la cadena de bloques, un contrato inteligente se convierte en un programa informático que funciona automáticamente y se ejecuta autónomamente cuando se cumplen condiciones específicas. En la cadena de bloques, los contratos inteligentes permiten que el código se ejecute exactamente como se programó sin ninguna posibilidad de tiempo de inactividad, censura, fraude o interferencia de terceros. Además de permitir el intercambio de dinero también puede realizar intercambios de contenido, propiedad, acciones o cualquier cosa que se le de un valor. La red Ethereum se puso en funcionamiento el 30 de julio de 2015 con 72 millones de premiados por Ethereum [15].

3. **Ripple**: se abrevia (XRP) y es un activo digital independiente que es nativo del libro mayor de consenso de Ripple. Con un gobierno probado y la confirmación de transacción más rápida de su tipo, se dice que XRP es la opción de liquidación más eficiente para las instituciones financieras y los proveedores de liquidez que buscan un alcance global, accesibilidad y una rápida liquidación para los flujos interbancarios [14].

4. **Litecoin**: se abrevia (LTC) y es una criptomoneda de igual a igual creada por Charlie Lee. Fue creado en base al protocolo Bitcoin pero difiere en términos del algoritmo de hash utilizado. Litecoin utiliza el algoritmo de extracción de trabajo de prueba de escritura de script intensivo en memoria, el mismo le permite que el hardware de grado de consumo, como GPU, extraiga esas monedas [45].

5. **Bitcoin Cash**: Bitcoin Cash (BCH) es una bifurcación (una actualización activada por la comunidad del protocolo o código) de Bitcoin que entró

en vigencia el 1 de agosto de 2017 y que aumentó el tamaño del bloque a 8MB, para ayudar a escalar la tecnología subyacente de Bitcoin. El 16 de noviembre 2018: BCH se bifurcó nuevamente y se dividió en Bitcoin SV y Bitcoin ABC. Bitcoin ABC se convirtió en la cadena dominante y se apoderó del ticker BCH, ya que tenía más poder de calculo de hash y la mayoría de los nodos de la red [46].

6. **EOS**: EOS es un token que tiene sus orígenes en el blockchain de ETH pero posteriormente migro a su propio blockchain. Su protocolo emula la mayoría de los atributos de una computadora real, el hardware CPUs y GPUs para procesamiento, memoria en disco duro y memoria RAM, con los recursos computacionales distribuidos equitativamente entre los titulares de criptomonedas. EOS funciona como una plataforma de contrato inteligente y un sistema operativo descentralizado destinado al despliegue de aplicaciones descentralizadas a escala industrial a través de un modelo de empresa autónoma descentralizada. La plataforma de contrato inteligente pretende eliminar las tarifas de transacción y también realizar millones de transacciones por segundo. EOS presenta una arquitectura de cadena de bloques diseñada para permitir el escalamiento vertical y horizontal de aplicaciones descentralizadas [47].

7. **Binance coin**: Binance Coin (BNB) es la criptomoneda de la plataforma de Binance. El nombre "Binance"se refiere a una combinación de binario y finanzas. A partir de 2019, muchas empresas aceptan el BNB como forma de pago.

8. **Bitcoin SV**: Bitcoin SV significa Satoshi Vision. Derivado de Bitcoin Cash, BSV es una bifurcación fija (actualización activada por la comunidad del protocolo o código) establecida como distinta de BCH después de la actualización de la red programada para el 15 de noviembre de 2018 que resultó en una guerra de trash que determinó que las cadenas se dividirían. Según su sitio web, el proyecto Bitcoin SV está respaldado principalmente por CoinGeek Mining con el trabajo de desarrollo realizado por nChain.

9. **Tether**: Tether (USDT) es una criptomoneda con un valor fijo que se ajusta al valor del dólar estadounidense. El objetivo es tener una criptomoneda estable que se pueda utilizar como dólares estadounidenses digitales. Existen más criptomonedas de este tipo que toman el valor del dólar estadounidense a estas monedas que sirven para este propósito se denominan "monedas estables". Según su sitio, Tether convierte el efectivo en moneda digital, y por cada token emitido dicen guardar un dolar real.

10. **Stellar**: La red Stellar es una red de código abierto, distribuida y de propiedad de la comunidad que se utiliza para facilitar las transferencias de valor entre activos. Stellar tiene como objetivo ayudar a facilitar la transferencia de valor de activos cruzados a una fracción de un centavo, mientras que pretende ser un sistema financiero abierto que ofrezca a personas de todos los niveles de ingresos acceso a servicios financieros de bajo costo. Stellar puede manejar intercambios entre monedas basadas en fiat y entre criptomonedas. Stellar.org, la organización que apoya a Stellar, está centralizada como XRP y está diseñada para manejar transacciones multiplataforma y micro transacciones como XRP. Sin embargo, a diferencia de Ripple, Stellar.org no tiene fines de lucro y su propia plataforma es de código abierto y descentralizada. Mediante el uso de su moneda intermedia Lumens (XLM), un usuario puede enviar cualquier moneda que posea a cualquier otra persona en una moneda diferente.

    Stellar fue fundada por Jed McCaleb en 2014. Jed McCaleb también es el fundador de Mt. Gox y co-fundador de Ripple, lanzaron el sistema de red Stellar con la abogada Joyce Kim. Stellar también es una tecnología de pago que tiene como objetivo conectar a las instituciones financieras y reducir drásticamente el costo y el tiempo requerido para las transferencias transfronterizas. De hecho, ambas redes de pago utilizaron inicialmente el mismo protocolo.

Las criptomonedas se han conformado y consolidado como un medio de valor alternativo. El uso de las mismas ha ido en aumento desde su aparición. Muchas han aumentado su valor a más del doble o triple y diversos países han comenzado a regular su uso. Si bien se prevé que ningún país adopte el uso de una criptomoneda que no pueda controlar, si existen propuestas de que surjan criptomonedas emitidas por los propios países como monedas digitales estables que contengan las ventajas de las mimas y puedan ser utilizadas para automatizar servicios. También se contempla una disminución del dinero en efectivo. Estas monedas digitales emitidas por los propios países serian compatibles con la propuesta de la investigación.

2.4. Blockchain

Blockchain es una base de datos descentralizada que esta formada por múltiples bloques interconectados por llaves criptográficas de bloques anteriores. Su origen se remota al 31 de octubre del 2008, cuando Satoshi Nakamoto publicó el artículo "Bitcoin: A Peer-to-Peer Electronic Cash System"[1] en un foro de criptografía.

En su artículo propuso un sistema distribuido y políticamente descentralizado para un sistema de envió de dinero mediante tokens digitales, que utilizado un conjunto de técnicas criptográficas es capaz de validar las transacciones entre los usuarios a través de los mismos nodos de la red que se conectan voluntariamente, eliminando el control de la red a una sola persona u organización.

El 3 de enero del 2009 fue generado el primer token y así fue como nació el Bitcoin (nombre que se le dio al token) y a la tecnología distribuida se le nombró "Blockchain"(la traducción popular al español es cadena de bloques). En el articulo de S. Nakamoto no se menciona la palabra blockchain, sin embargo, el nombre se popularizó debido a que cada determinado tiempo generaba un bloque sellado criptográficamente que era unido a un bloque anterior parecido a una cadena [13].

2.4.1. Funcionamiento del blockchain

Para explicar brevemente el funcionamiento general del blockchain es importante mencionar los siguientes conceptos básicos:

- Hash: Es una huella digital de un archivo que tiene una determinada estructura y de tamaño fijo. Sin importar el archivo de entrada siempre el resultado es del mismo tamaño, una de sus características es ser irreversible, es decir, a partir de un hash es imposible calcular el archivo del cual se obtuvo [48]. Otro atributo es que cualquier cambio en el archivo o texto, por mínimo que este sea, el hash de salida cambia completamente. Un ejemplo de algoritmo hash es SHA256, que es el que utiliza el bitcoin [49].

- Algoritmo de consenso: Es una técnica para lograr el consenso entre nodos distribuidos, que debe ser capaz de tomar una decisión en caso de que no todos los nodos presenten los mismo datos, (en la sección 2.5 se explica más a detalle). El primer algoritmo de consenso utilizado en un blockchain fue proof of work que es un técnica similar al hashcash, propuesta por Adam Back en [44], para combatir el spam en los correos electrónicos. Se tomará de ejemplo este algoritmo de consenso para realizar la explicación de funcionamiento.

- Nonce: Es un número que se agrega al contenido de un archivo, el *nonce* se utiliza para variar el contenido del archivo hasta encontrar un hash que satisfaga la proof of work del mismo archivo [1].

Una vez aclarados los conceptos básicos, es posible explicar el funcionamiento general del blockchain:

1. En un bloque como el que se presenta en la Figura 2.4, se añade la información que se va a guardar. En este bloque se añade el hash del bloque anterior (que debe cumplir las características de la prueba de trabajo). Este hash crea una unión con el bloque anterior de manera que los bloques resultantes no pueden ser modificados.

2. Se añade el *número nonce*, que se modifica cuando se calcula el hash del bloque (esto se hace cuando se extrae el bloque). Dado que el hash es necesario para cumplir con cierta estructura, es necesario variar el contenido del bloque y el *número nonce* hasta obtener el resultado esperado.

3. Mientras se valida el bloque, no es posible añadir más información a dicho bloque (es decir, se cierra). La validación se realiza siguiendo el mecanismo de prueba de trabajo que se utiliza en toda la red. Este proceso se conoce también como minería de bloques. En esta validación, el *número nonce* se modifica hasta que el hash del bloque cumple con ciertas especificaciones. La validación lleva cierto tiempo dependiendo de la dificultad del mecanismo de prueba de trabajo utilizado.

4. Cuando se completa el proceso de validación del bloque, el hash de dicho bloque se añade al siguiente bloque que, para entonces, contendrá nuevos datos.

Cualquier cambio realizado en un bloque validado será fácilmente detectado, ya que el hash del bloque cambiará completamente y, por lo tanto, se convertirá en un bloque inválido. Esto ocurrirá cuando el siguiente bloque, que contiene el hash del bloque actual, sea validado permitiendo detectar cualquier inconsistencia de la información. Del mismo modo, el bloque modificado también tendrá que ser validado de nuevo, así como todos los bloques que aparezcan después de él.

Este proceso de validación hace que la información almacenada en una cadena de bloques sea más fiable, dada la dificultad y el tiempo que se requiere para validar de nuevo los bloques que están delante del bloque modificado. En la Figura 2.4 se ilustra de manera general los datos que contiene el blockchain del bitcoin.

Para controlar la creación de nuevos bloques, en el caso del blockchain de Bitcoin la dificultad se reajusta cada 2016 bloques (aproximadamente catorce días), con tal de que la creación de nuevos bloques tenga una frecuencia aproximada de un bloque cada diez minutos [50]. La fórmula para dicho ajuste es la siguiente:

$$dn = \frac{dp * 2s}{tm}$$

Donde dn es la nueva dificultad, dp es la dificultad previa, s son semanas y tm es el tiempo en minar los anteriores 2016 bloques.

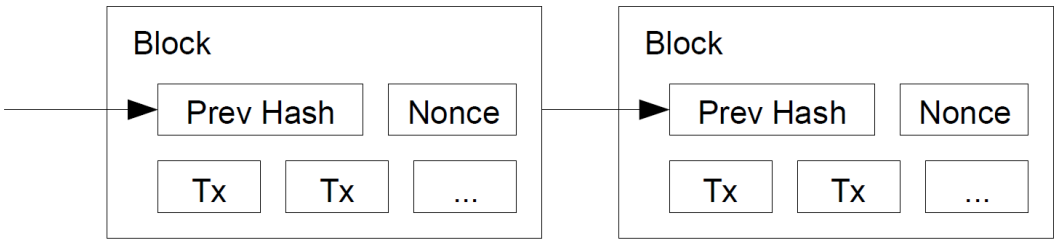

Figura 2.4: Bloques con los datos generales del blockchain [1], se resalta que cada bloque contiene el hash valido del bloque previo.

Cada nodo que se une al blockchain obtiene una copia de la cadena de bloques, siendo distribuida en diferentes computadoras, haciendo una descentralización física y que por medio de algoritmos de consenso se sincronizan y actualizan.

La proof of work que actúa como un sellado criptográfico se dificulta a posibles atacantes modificar la información de los bloques e impiden que estos se distribuyan en la red como bloques válidos, puesto que es relativamente sencillo identificar a bloques que han sido manipulados [13].

Si bien blockchain surgió como un sistema políticamente descentralizado, posteriormente surgieron implementaciones del blockchain que no eran políticamente descentralizados obteniendo algunas ventajas, generando así dos tipos de blockchain.

2.4.2. Tipos de blockchain

En general se identifican dos tipos de blockchain los cuales se diferencian porque en un blockchain publico cualquier computadora se puede unir a la red del blockchain, descargar la cadena completa de bloques e incluso en algunos casos agregar información propia a los bloques. En cambio si es un blockchain privado solo pueden participar determinadas computadoras autorizadas en la red.

1. Blockchain pública o blockchain no permisionados

Permite que cualquier computadora se pueda unir a la red, por lo que son políticamente descentralizadas puesto que el control pertenece a la misma red y en consecuencia se requiere de la máxima seguridad para impedir que alguna organización o individuo tome el control de la misma.

A mayor distribución de la red, mayor es su dificultad para realizar un ataque. Para lograrlo sacrifican velocidad en las transacciones que se pueden almacenar en el bloque. Algunos ejemplos de blockchain públicas son Bitcoin y Ethereum [15]. Ethereum, genera tokens llamados ETH e implementa una mejora para agregar funcionalidad que permite utilizar smart contracts (programas que ejecutan acuerdos registrados entre dos o más entes), capaces de ejecutarse cuando se cumplan las condiciones del contrato. Las reglas y políticas quedan grabados en el blockchain que es capaz de comprobar automáticamente el cumplimiento de dichas especificaciones con un alto grado de confiabilidad.

2. Blockchain privada o blockchain permisionados

Estas blockchain no permiten que cualquier computadora pueda realizar las verificaciones de su red, normalmente pertenecen a una empresa u organización que la controla, algunas permiten el acceso al contenido de sus bloques, otras no. Por ejemplo la empresa Ripple, realizó su implementación produciendo su token XRP que redujo la cantidad de tiempo necesario para que se confirmara una transacción (en el orden de segundos) [14]. Esto le impide ser políticamente descentralizada, la seguridad recae en la empresa, pero su principal ventaja es que al controlar los equipos que se unen a su blockchain puede optimizar las transacciones lo que se traduce en una validación de transacciones más rápida en comparación de una blockchain pública.

Existe también una clasificación llamada consorcio de blockchain [51] en la que este blockchain sería controlado por varias organizaciones. Mediante la realización de un consenso entre los múltiples dueños para determinar las decisiones. Sin embargo, no es diferente del blockchain privado en cuanto a sus políticas, puesto que la única diferencia entre estos, es que en el privado el blockchain pertenece a una organización, mientras que en el consorcio blockchain pertenece a un grupo de organizaciones, de tal manera que en ambas, cualquier equipo que quiera participar en la red no lo puede hacer.

Ventajas de los tipos de blockchain

Como ya se ha mencionado la principal ventaja de los blockchain privados es la alta velocidad de transacciones que puede obtener, por lo que también son más eficientes al controlar los equipos que componen la red, en consecuencia también requieren menos energía eléctrica. A pesar de que los blockchain privados pueden hacer publico el contenido de sus bloques, dicho contenido es vulnerable por el dueño o dueños del blockchain ya que en su poder están los equipos que controlar el blockchain y podrían realizar cambios en bloques pasados con la consecuencia de

que entre más antiguo sea el bloque más difícil sera modificarlo. En los blockchain público es casi imposible realizar una modificación a un bloque pasado, por lo tanto la información es altamente confiable y se tiene la certeza de que no va a cambiar por intereses de algunos (Ver Tabla 2.2).

Propiedad	Blockchain público	Blockchain privado
Transacciones	Bajo	Alto
Consumo Eléctrico	Alto	Bajo
Cambios de datos	Muy difícil	Vulnerable
Descentralización política	Si	No
Transparencia de los datos	Si	Depende del dueño

Tabla 2.2: Comparación entre blockchain público y blockchain privado

2.4.3. Retos del blockchain

En [52] J. Leon et. al. Afirman que el "Blockchain está a punto de convertirse en la invención más emocionante después de Internet", sin embargo destacan que la tecnología blockchain todavía se encuentra en una etapa de desarrollo y se necesita más investigación para mejorar su eficiencia y seguridad, así como advierten que los investigadores se enfrentan a muchas oportunidades y desafíos para hacer que blockchain sea exitoso, debido a sus distintas desventajas que se mencionan a continuación:

- En caso de que se quiera distribuir indiferentemente los nodos que comprueban las transacciones, necesitan ser recompensados por el mismo trabajo criptográfico que realizan (los llamados mineros), lo que genera un coste por transacción el cual aumenta conforme aumenta la demanda de transacciones.
- Dado que blockchain requiere un proceso de verificación muy estricto para crear un nuevo registro de transacciones, como consecuencia cuando se supera la capacidad de comprobación de transacciones surja una cola de transacciones a confirmarse, lo que vuelve lento el proceso de confirmaciones y se presenta un desperdicio de recursos computacionales porque distintos equipos trabajan para generar el siguiente bloque en la cadena pero solo uno (el que termina primero) lo agrega. Es decir presenta problemas de escalabilidad.
- Un problema en los blockchain públicos surge cuando no existe la suficiente distribución en el blockchain ya que si un nodo concentra el 51 % (más

de la mitad) de poder de procesamiento puede dominar a todos los demás nodos y manipular los registros en una cadena de bloques. Pero solo podrían manipular los últimos bloques de la cadena ya que entre más antiguo es el bloque mayor cantidad de procesamiento se requiere [23].

- La privacidad y la confidencialidad siguen siendo un problema con blockchain, debido a que todos los nodos de la cadena de bloques tienen acceso a todos los datos, aunque no puedan modificarlos [53], esto también depende de la aplicación que se le quiera dar.

Debido a estos problemas surgen nichos de oportunidad de mejora como bien lo menciona J. Leon en [52]. Para mejorar la escalabilidad en blockchain han surgido diversas propuestas de distintos grados de modificación al blockchain original. Los primeros buscaban resolver el problema sin modificar el funcionamiento del blockchain, uno de esos fue aumentar el tamaño del bloque un determinado porcentaje [25]. No obstante el problema se solucionaba momentáneamente, después se propuso eliminar el límite del tamaño de bloque [26] pero eso generaría bloques de gran tamaño y solo empresas con grandes cantidades de almacenamiento podrían ser mineros, lo que haría que se centralizara la red blockchain. Existen otras modificaciones similares a las mencionadas y en general se observan entre una y otra que se pueden obtener dos de los siguientes tres atributos [52]: seguridad, descentralización, velocidad de transacciones, como se puede ver en la Figura 2.5.

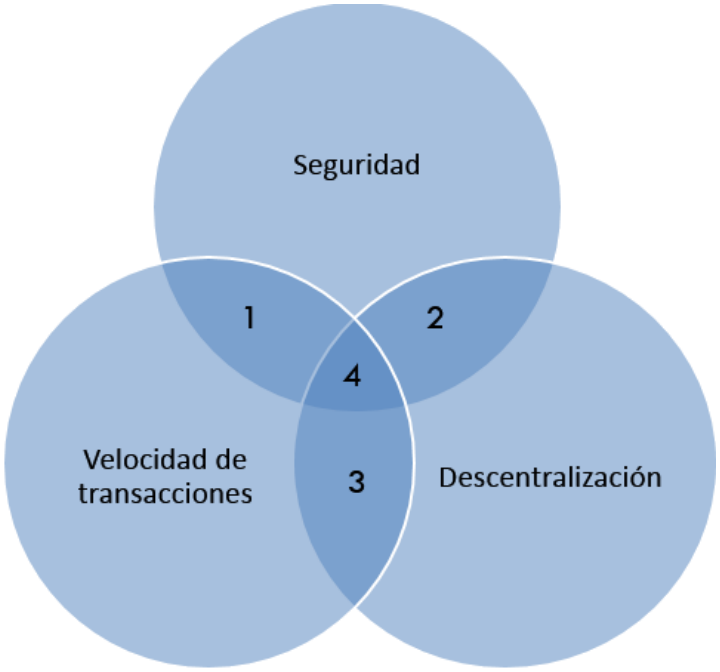

Figura 2.5: Atributos de los blockchain [2]

De acuerdo a la Figura 2.5 se tienen las siguientes opciones:

1. Se configura el blockchain para aumentar la velocidad de transacciones y mantenerse seguro. La desventaja es que se termina centralizando la red, hasta ahora son controladas por organizaciones, es común en los blockchain privados.

2. Aumentar la velocidad de transacciones y estar descentralizada. Esta opción implica bajar la seguridad, siendo la opción menos recomendada puesto que la seguridad es prioritaria.

3. Se configuran para estar descentralizadas y mantener la seguridad en la red, sacrifican la velocidad en las transacciones, es común en los blockchain públicos.

4. Es el estado ideal donde se obtienen los tres atributos deseables, seguridad, velocidad de transacciones y descentralización.

Existe un proyecto similar al blockchain pero de distinto funcionamiento que busca tener los tres atributos, seguridad, escalabilidad de transacciones y descentralización, la tecnología IOTA [54]. IOTA está pensado para el Internet de las cosas (IoT) y sus principales innovaciones radican en que, en vez de la tradicional cadena de bloques, utiliza una arquitectura basada en un concepto matemático llamado Gráfico Acíclico Dirigido (DAG), conocido como "*The tangle*"(enredadera)

debido a su aspecto de red enmarañada. Dicha arquitectura hace posible que no necesite de mineros y que la red aumente su capacidad conforme se incremente el número de usuarios, es decir que a mayor demanda de transacciones mayor será su velocidad. Para que un usuario de IOTA pueda realizar una transacción primero tiene que validar otras dos transacciones seleccionadas aleatoriamente [54]. Una transacción tiene que acumular un nivel suficiente de verificaciones (tiene que ser validada un determinado número de veces por otros usuarios) para ser aceptada por el receptor, eso permite que los propios usuarios con sus transacciones verifican las transacciones de otros participantes en la red. Esto busca que no existan comisiones por realizar transacciones y que la red sea escalable, segura y descentralizada.

El problema surge cuando la red no es utilizada constantemente, ya que se vuelve lenta. Si se realiza una transacción y después ya no se genera otra, la primera se queda sin confirmarse y así quedara indefinidamente hasta que existan suficientes nuevas transacciones que confirmen las anteriores. Una primera solución fue agregar uno o varios nodos confiables que estuviesen comprobando las transacciones, pero esto claramente centraliza la red al depender de esos nodos y disminuye la seguridad de la red.

El blockchain ha demostrado tener la suficiente capacidad criptográfica para ser confiable, su utilización sigue en constante aumento, las aplicaciones para las que puede ser utilizado están aumentando conforme distintos grupos de usuarios lo conocen. En algunos años gobiernos, organizaciones públicas y privadas utilizarán directa o indirectamente sistemas basados en blockchain por lo que será importante contar con personas calificadas para implementar el uso de estos sistemas, gobiernos y universidades deberán invertir en la generación de personal capacitado. En cuanto sea viable van a desaparecer algunos intermediarios de servicios que puedan ser sustituidos por los *smart contracts* que funcionan en distintos blockchain por lo mismo se requiere más investigación para ambos.

Los blockchain públicos son difíciles de regular por los gobiernos, por lo que corren el riesgo de ser prohibidos o limitados en varios países, sin embargo los propios gobiernos seguramente implementaran sus propios blockchain para montar uno o varios de sus servicios actuales, también es probable que los países emitan su propias criptomonedas en un futuro, como es el caso de Venezuela que ya ha emitido la suya. Habrá que estar pendientes de los cambios y novedades de estas tecnologías para intentar prever lo que se aproxima en un futuro no muy lejano.

2.5. Modelos de consenso

Los modelos de consenso fueron creados para el apoyo de toma de decisiones para un grupo, donde cada individuo dentro del grupo participa y apoya una decisión que funcionar mejor entre los integrantes del grupo. Es una forma de resolución democrática en la que los miembros deben apoyar la decisión mayoritaria. El consenso es fácil cuando todo va bien, sin embargo, cuando surgen fallos por diversas circunstancias como lo pueden ser canales imperfectos, caídas de participantes, violación de sincronizaciones o incluso cuando algunos de ellos pueden conspirar para que ese consenso no se produzca (comportamiento malicioso), es cuando surge la necesidad de los modelos de consenso y son requeridos para los sistemas distribuidos y los sistemas descentralizados donde suelen ocurrir las fallas antes mencionadas [55].

Un algoritmo de consenso en computación es un protocolo utilizado para lograr un acuerdo sobre un valor de datos único entre procesos o sistemas distribuidos. Estos algoritmos se utilizan principalmente para lograr la fiabilidad en una red en la que intervienen múltiples nodos distribuidos que contienen la misma información [56]. En el caso del blockchain, el algoritmo de consenso es el elemento que define la forma en que se crean, distribuyen y validan los bloques [57]. Para optimizar las implementaciones de las cadenas de bloques y en función de las propiedades que los blockchain deben mantener, han surgido diferentes propuestas de algoritmos de consenso. Más información sobre las características de estos algoritmos de consenso se puede encontrar en [58].

Los algoritmos de consenso tienen los siguientes objetivos particulares:

- **Actividad:** Cada miembro del grupo es igualmente activo. No hay miembros con más responsabilidades que otros en el grupo.
- **Llegar a un acuerdo:** El mecanismo reúne todos los acuerdos del grupo tanto como puede.
- **Cooperación:** Cada miembro trabajara en equipo y dejara de lado sus propios intereses.
- **Participación:** Todos los que están dentro de la red deben de participar en la votación. Nadie se quedar fuera o nadie puede quedarse fuera de la votación.
- **Colaboración:** Cada uno en el grupo apunta a un mejor acuerdo que resulte en los intereses colectivos del grupo.

- **Igualdad de derechos:** Cada uno de los participantes tiene el mismo valor en la votación. Esto significa que el voto de cada persona es importante.

Un sistema es fiable si el mismo se puede adaptar y soportar fallos ante diversas circunstancias. Para lograrlo es común resguardar un respaldo de información en varios nodos, replicando la información siempre se puede recuperar un nodo caído y después puede ser restablecido. El reto es mantener la consistencia de la información guardada en cada nodo, puesto que pudiera que algunos nodos se pongan de acuerdo y quieran hacer pasar información alterada como valida. El problema de los generales bizantinos ejemplifica los dilemas que pueden surgir, a continuación se describe el problema.

2.5.1. El dilema de los generales bizantinos

El dilema de los generales bizantinos fue concebido en 1982 [59] como un problema lógico que muestra cómo en un grupo de generales bizantinos, se pueden generar problemas con la información y las acciones a proseguir, cuando uno o varios generales se proponen a frustrar un ataque.

El problema llega cuando cada general tiene su propio ejército y cada grupo está ubicado en lugares geográficos distintos, rodeando una supuesta locación que se quiere atacar. Los generales antes de atacar tienen que sincronizarse para hacerlo al mismo tiempo o deben decidir si es mejor retirarse. Al final no importa si se retiran o si atacan, lo importante es que todos los generales lleguen a un acuerdo para tomar una decisión en común para ejecutarla en coordinación.

Entonces, se consideran los siguientes objetivos:

- Cada general debe decidir si atacar o no.
- Una vez que se evalúa la situación se hace una elección que no se puede cambiar;
- Todos los generales deben acordar una misma decisión y ejecutarla de manera coordinada.

Los generales bizantinos se comunican únicamente con mensajes de correo, y dichos mensajes son propensos a perderse, destruirse o tener retrasos.

También, es posible que si el mensaje se entrega integro, uno o varios generales bizantinos pueden traicionar por cualquiera que sea el motivo y mandar mensajes fraudulentos para corromper las comunicaciones o confundir a sus homólogos, lo que llevaría a una desorganización y en consecuencia un error total.

En el caso de los sistemas distribuidos se puede aplicar la analogía y equiparar a cada general bizantino con un nodo en la red. Los nodos deben alcanzar un consenso con los demás nodos incluso si hay nodos maliciosos. Entonces, los nodos de la red deben ponerse de acuerdo y ejecutar una misma acción para evitar una falla que ocasione la perdida del servicio total.

Al analizar el problema de los generales bizantinos se concluye que la única forma de lograr un consenso es tener al menos el 66 % o más de generales o nodos de red honestos y confiables. Esto significa que si menos de un tercio actúa de manera maliciosa, los otros dos tercios podrán tomar la mejor decisión, sin embargo si la mayoría de la red decide actuar maliciosamente, el sistema es susceptible de fallas y ataques.

Se dice que si sistema tiene **tolerancia a fallas bizantinas (BFT)** significa que dicho sistema es capaz de resistir al conjunto de fallas derivadas del problema del dilema de los generales bizantinos. Esto quiere decir que si un sistema es BFT puede continuar funcionando incluso si algunos de los nodos fallan o actúan maliciosamente [60].

Existen diferentes formas de alcanzar la consolación de decisiones en el dilema de los generales bizantinos, algunas más costosas que otras, por lo tanto, hay múltiples formas de construir un sistema que sea BFT. Estas maneras de llegar a un consenso fueron retomadas en los sistemas descentralizados de los blockchain y fueron adecuadas para su funcionamiento y han surgido diferentes soluciones que logran la tolerancia a faltas bizantinas y esto dio origen a los llamados algoritmos de consenso en los blockchain, algunos son:

2.5.2. Proof of work (Pow)

Proof of work o prueba de trabajo, es el primer algoritmo de blockchain introducido en la red blockchain. Muchas tecnologías de blockchain utilizan estos modelos de consenso para confirmar todas sus transacciones y producir bloques relevantes para la red.

El protocolo de prueba de trabajo (PoW) es una medida fiscal para desalentar los ataques de denegación de servicio (DDOS) y otras explotaciones de servicios de red como los spam que consumen el tiempo de procesamiento de la computadora [61].

En PoW, los nodos que calculan valores (hashes) se llaman mineros. Cada nodo en la red calcula el valor hash del bloque encabezado que contiene un nonce. Luego, los mineros cambian este valor con frecuencia a generar diferentes valores hash. Este protocolo implica que los valores calculados sean igual o menor que un valor especificado. Una vez que un nodo alcanza el valor objetivo, se difunde el bloque a otros nodos y ellos a su vez confirman la precisión del valor hash. Si se

autentica un bloque, otros nodos agregan este nuevo bloque validado a su propia blockchain. El proceso de calcular los valores hash se llama minería.

En un sistema descentralizado, los bloques válidos tienen probabilidad de que se produzcan simultáneamente, los nodos que encuentran el nonce aproximadamente al mismo tiempo genera que existan bifurcaciones en la cadena principal que existen momentáneamente. Las bifurcaciones se resuelven cuando se genera el siguiente bloque o siguiente par de bloques. La cadena más larga es la confiable y correcta porque significa que tiene una mayor prueba de trabajo en comparación con las más cortas. Para encontrar la cadena válida más larga, mucho del potencia computacional de la red se desperdicia. Algunos protocolos usan aplicaciones secundarias junto con PoW para mitigar la pérdida [62]. PoW es el algoritmo de consenso más popular en las blockchain públicas porque a pesar de propiciar un el desperdicio de recursos de los nodos de la red es altamente efectiva para lograr el consenso con la máxima distribución sin establecer un máximo de nodos que se pueden unir a la red.

2.5.3. Proof of stake (PoS)

Proof of Stake (PoS) es un protocolo que establece que un nodo o minero puede minar o validar transacciones en un bloque dependiendo de la cantidad de participación que dicho usuario tiene. Este protocolo confía en que si las personas tienen más valor de participación involucrado en el blockchain, son menos las probabilidades de que dicho nodo quiera atacar la red.

Los mineros en PoS necesitan probar la propiedad de la cantidad de tokens con los que participan. Sin embargo, este método de selección es injusto, basado en la investigación en [63] donde la persona más rica de la red puede actuar de forma egoísta, en la investigación se explica el protocolo en detalle y proporciona ejemplos para una mejor comprensión.

En este protocolo, la cadena de bloques rastrea un conjunto de mineros y cualquier minero debe contener la criptomoneda base para convertirse en un validador. El minero envía una forma específica de transacción que bloquea la criptomoneda como depósito. El proceso de creación y la validación de un nuevo bloque es realizada por todos los participantes validados.

En PoS el verificador es seleccionado pseudoaleatoriamente para un intervalo de tiempo y se le asigna la autoridad para crear un bloquear. El bloque creado debe tener el bloque anterior como precursor para lograr un sola cadena de bloques de alargamiento en continuo crecimiento. En PoS de estilo BFT, los mineros se les permite sugerir aleatoriamente la creación de un nuevo bloque. Sin embargo, la aprobación de un bloque se lleva a cabo mediante un proceso de votación de varias rondas, donde el verificador vota por un bloque preciso y, finalmente, todos

los verificadores acuerdan la validez del bloque que se va a agregar al blockchain

2.5.4. Delegate Proof of stake (DPoS)

La prueba de participación delegada (DPoS) es un algoritmo basado en el protocolo PoS. DPoS varía del algoritmo PoS en el aspecto que en DPoS, los titulares de las criptomonedas votan por los delegados para validar y procesar una transacción a cambio de tarifas de transacción, que es diferente en PoS donde una parte interesada valida y procesa una transacción para obtener recompensas y transacciones honorarios.

La diferencia entre PoS y DPoS se puede percibir como la diferencia entre un democrático directo y el democrático representativo. Partes interesadas en un sistema de criptomonedas eligen a sus delegados, quienes a su vez generan y validan bloques.

En DPoS el poder de aprobación de los interesados se utiliza para resolver los problemas de consenso de una manera justa y democrática. La selección determinista de los productores de bloques permite que las transacciones se confirmen en un promedio de solo unos segundos. Quizás lo más importante, el protocolo de consenso está diseñado para proteger a todos participantes contra una interferencia regulatoria no deseada [64].

Una transacción se confirma rápidamente si es necesario validar menos nodos, sin embargo, la forma de validación de bloque podría conducir a la manipulación de los parámetros de bloque tales como el tamaño y el intervalo de los delegados seleccionados. El proceso DPoS implica el uso de subredes de confianza dentro de una red más grande en la que los nodos se pueden dividir ya sea un servidor o un cliente. Un servidor contribuye al proceso de consenso y cada el servidor contiene una lista de nodos única, mientras que el cliente transferiría fondos. Para validar una transacción, el servidor consulta los nodos enumerados en la lista de nodos única. Si los acuerdos alcanzan al menos el 80 %, la transacción se valida y se agrega al libro mayor. Con el punto de vista de los nodos, el libro mayor o la transacción siguen siendo precisos y corrija hasta que el porcentaje de nodos defectuosos en la lista de nodos única permanezca por debajo del 20 %.

2.5.5. Practical Byzantine Fault Tolerance (PBFT)

Practical Byzantine Fault Tolerance (PBFT) es un algoritmo de imitación creado para soportar fallas bizantinas. En [65] se discute el protocolo en detalle mostrando que para tolerar las fallas bizantinas (se requiere entender el problema bizantino) es necesario contar con 2/3 de nodos confiables.

El PBFT es un algoritmo maneja hasta 1/3 réplicas bizantinas malévolas. Una vez que se resuelve un nuevo bloque, después de una ronda, se selecciona un primario basado en reglas predefinidas y es responsable para ordenar la transacción para cada ronda. Todo el proceso se divide en tres fases: pre-preparadas, preparadas y comprometidas. En todas las fases, el nodo avanza a la siguiente fase solo después de recibir 2/3 de los votos de todos los nodos en la red [66]. En PBFT, cada nodo es conocido por otros nodos en la red. Existe también la variante delegada a PBFT. La tolerancia de fallas bizantinas delegada (dPBFT) es un algoritmo como PBFT, sin embargo, en dPBFT se vota un grupo de nodos profesionales para registrar las transacciones en lugar de que los nodos sean aleatorios.

Existen más propuestas de modelos de consenso para los blockchain, sin embargo, son poco utilizados porque presentan mayores desventajas que ventajas y otros aun están en desarrollo. Los mencionados anteriormente son los más funcionales y utilizados.

A pesar de las diferentes propuestas de algoritmos de consenso, no se ha logrado resolver el problema de escalabilidad en la velocidad de transacciones sin sacrificar descentralización, funcionalidad o seguridad. Por ello han surgido propuestas que plantean la posibilidad de solución con la separación de funciones y/o información en capas. A continuación se describen estas propuestas.

Capítulo 3

Estado del Arte

En este capítulo se analiza el estado del arte, se identificaron 6 propuestas que son relacionadas a la temática de blockchain multicapas. Posteriormente se realiza una comparativa entre los 6 trabajos de investigación.

3.1. Trabajos Relacionados

En [67] Cheng y Zhang (2017) presentan una propuesta de un modelo de red basado en blockchain para mejorar la implementación de dispositivos IoT, manteniendo la seguridad de la red y combinando protocolos de almacenamiento en la nube. El modelo reduce la dificultad de implementación real de la tecnología blockchain, dividiendo el Internet de las cosas en un multinivel y la adopción de la tecnología blockchain en todos los niveles de la red, con la alta seguridad y credibilidad. Proporcionando una solución de red de área amplia de Internet de las cosas.

Su modelo consta de dos tipos de capas con subcapas, las capas externas y las capas de alto nivel. Donde las capas externas funcionan de manera centralizada como lo hacen los servidores en la nube comúnmente, mientras que las capas de alto nivel se conectan a las capas externas y se comportan como dispositivos conectados mientras que los dispositivos de la IoT se conectan a las capas de alto nivel. A diferencia de la capa de externa, en la capa de alto nivel, la red está descentralizada.

Algunas de las ventajas que se pueden obtener son:

- El equipo IoT se coordina localmente por un instrumento centralizado, con mayor capacidad y seguridad.

- La presencia de múltiples centros reduce la carga de la red computacional de la IoT, así como la reducción del riesgo de concentrado.
- En comparación con la red centralizada tradicional, las comunicaciones son peer-to-peer entre los centros, aumentando la disponibilidad del sistema.
- Contratos inteligentes en múltiples blockchains permiten asegurar la red IoT además de hacerla segura y fiable.

Posibles defectos pueden estar relacionadas con lo siguiente.

- Es costo realizar la mejora a la red total.
- Toma mucho tiempo y recursos mantener los contratos inteligentes.

Como mejoras a futuro los autores proponen construir aplicaciones prácticas de este modelo. Realizar el análisis de los problemas de incertidumbre y mejorar del modelo de red IoT.

En [68] Badr et al. (2018) describen un modelo multicapa para los datos clínicos de pacientes, retomando el modelo propuesto por Cheng y Zhang pero reconfigurando las capas en 3 niveles principales.

1. El primer nivel para los dispositivos y sensores del paciente.
2. El segundo nivel correspondiente a hospitales, laboratorios, organismos médicos, etc.
3. El tercer nivel para el almacenamiento en la nube de manera centralizada.

Proponen utilizar datos encriptados con el algoritmo PBE-DA, para mantener la confidencialidad de los pacientes. El algoritmo de encriptación permite utilizar seudónimos para reforzar la confidencialidad y evitar que terceros tracen una ruta para dar con los datos de un paciente.

En [69] Chang et al. (2018) proponen un modelo de dos capas basado en blockchain para preservar los datos clínicos de pacientes, asegurando su privacidad. Nombraron DeepLinQ al modelo al agregar algoritmos de aprendizaje profundo para asegurar la distribución de los datos sin sacrificar la privacidad de los mismos.

La distribución de los datos en las dos capas también mejora la flexibilidad y escalabilidad, con control de acceso granular y el adicionamiento de los contratos inteligentes.

El prototipo actual DeepLinQ emplea dos capas blockchain, donde la capa de base se encarga de validar los libros de contabilidad, mientras que la capa

de ramificación de implementos de capa de las características para apoyar las propiedades POET (una capa de base puede servir a múltiples capas de la rama con diferentes implementaciones).

DeepLinQ puede considerar que todas las bases de datos hospitalarias como el almacenamiento fuera de la cadena y el uso de contratos inteligentes para acceder al almacenamiento fuera de la cadena. Una vez que el contrato haya sido ejecutado y los datos han sido extraídas, los datos se pueden almacenar en una cartera asegurada.

En las capas de ramificación, DeepLinQ considera utilizar el acuerdo Federado bizantino (FBA) y hashgraph para lograr un alto rendimiento y baja latencia. Dejando pendiente un análisis para verificar el rendimiento con otros algoritmos de consenso.

En [3] Zhou et al. (2018) se propone una arquitectura multicapa de la cual se derivo una criptomoneda llamada MOAC. La propuesta contempla mantener seguridad, descentralización y velocidad de transacciones con funcionalidad de contratos inteligentes mediante la división de capas.

La propuesta contempla el flujo de la información por sus diferentes capas, siendo la primera capa las conexiones de los nodos distribuidos Peer-to-Peer. La red de nodos interconectados más la información en los bloques, que pertenecen a lo que ellos nombraron la MotherChain, conforman la segunda capa de la propuesta. La siguiente capa corresponde al despliegue de eventos y esta ligada a la cuarta capa que sigue manejando los contratos inteligentes. La ultima capa propone una API para montar los servicios que la quieran utilizar.

Las expectativas de la propuesta de MOAC fueron altas al prometer todos los atributos deseados en una red blockchain, velocidad de transacciones, seguridad y descentralización. Incluso contemplan su utilización para aplicaciones comerciales y el soporte para las mismas. Agregando funcionalidad de envíos de valor con el uso de su criptomoneda. Ver Figura 3.1.

La criptomoneda generada por MOAC no fue popular, esta clasificada en el lugar 762 con un valor por unidad de 0.07 USD a diciembre de 2020. Por lo que la propuesta no tuvo los resultados esperados en el modelo.

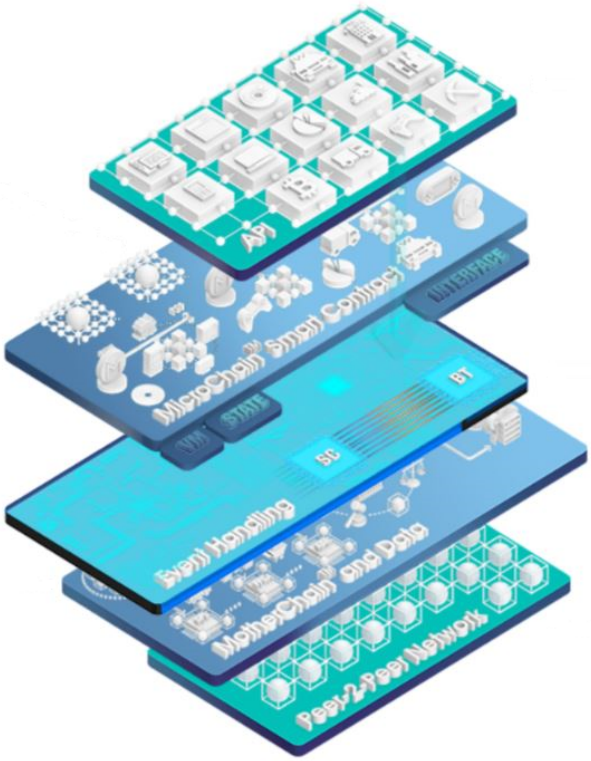

Figura 3.1: Capas propuestas por MOAC [3]

En [70] Chakraborty et al. (2018) proponen mecanismos de optimización para un modelo de red de IoT de múltiples capas, respaldado por la tecnología blockchain. Su objetivo es la optimización de la carga computacional para que el modelo pueda cumplir las condiciones factibles para su despliegue y maximizar el rendimiento de la red IoT sin comprometer la seguridad. Aunque la división de la red IoT en múltiples capas reduce la carga de cálculo en cada etapa, la división de la carga no es del todo proporcional a la cantidad de trabajo realizado en cada nivel. La propuesta consta de dos capas principales donde la segunda capa puede tener múltiples subcapas. El modelo se analiza en 2 aspectos: seguridad y rendimiento.

Los autores al final concluyen que su modelo tiene como objetivo proporcionar un marco más factible teniendo en cuenta una gran cantidad de factores en tiempo real. La elección de algoritmos de criptografía eficientes en blockchain juega un papel importante en el fortalecimiento de la red. Sin embargo, la falta de referencias industriales crea un obstáculo para calibrar el rendimiento de la red. El mayor enfoque de los negocios de TI tiene mucho margen para futuras investigaciones y estudios. Con nuevos desarrollos en este campo, no se puede negar que el lujo de la red IoT basada en blockchain se convertirá en una necesidad,

gestionará las transacciones diarias y, por lo tanto, conducirá a la evolución de una nueva era de estilo de vida inteligente.

En [71] López y Farooq (2020) presentan un marco blockchain de varias capas para el mercado de datos de movilidad inteligente, para abordar los desafíos asociados a la privacidad, la seguridad, la gestión y la escalabilidad. En la propuesta cada participante comparte sus datos cifrados con la red de blockchain y realizan transacciones con otros participantes siempre que ambas partes estén de acuerdo con las normas de transacción emitidas por el propietario de los datos. La propiedad de los datos, la transparencia, la auditabilidad y el control de acceso son los prioritarios para el mercado de datos de movilidad inteligente, por lo que son las bases para la propuesta.

Posteriormente presentan un estudio ejecutado en tiempo real que funciona con dispositivos heterogéneos y separados geográficamente que se comunican en una red física. También realizan una demostración donde garantizan la ciberseguridad y la privacidad de los individuos, protegiéndolos de los ataques de suplantación de identidad y de interceptación de mensajes.

La propuesta tiene un modelo general de 6 capas. La capa de identificación se compone de datos de movilidad y otra información que poseen los nodos. La capa de privacidad es el modelo de privacidad diferencial para acceder a los servicios basados en la ubicación. En la capa contract son el conjunto de contratos inteligentes y corredores que facilitan las transacciones de datos entre nodos. La capa de comunicación contiene los identificadores descentralizados de los nodos que sirven como puntos finales para establecer conexiones de igual a igual. La capa de consenso contiene los algoritmos de consenso en los que activo los nodos acuerdan escribir transacciones en el libro mayor. Finalmente, en la capa de incentivo están las recompensas que los nodos reciben por participar en el consenso y la recompensa que reciben los nodos por compartir (vender) su información.

3.2. Comparativa

En la Tabla 3.1 se pueden ver algunas características de las distintas propuestas multicapa de blockchain. En el caso de las propuestas 4 y 6 contemplan capas como parte del modelo de conexión que no utilizan cadenas de bloques como tal, por ejemplo la propuesta 4 agrega la capa api que ya no forma parte del blockchain principal. Mientras que la propuesta 6 tiene capas como parte de la identificación de los datos, una capa de incentivos para los que comparten los datos o los venden, donde ninguna de esas capas es parte del blockchain original. Por lo tanto en la ultima columna de la Tabla 3.1 aparece cuantas capas de diferentes blockchain realmente utilizan.

No	Año	Capas	Enfoque	Tipo	Capas multibloques
1	2017	2	Dispositivos IoT	Privado	1
2	2018	3	Expedientes clínicos con dispositivos IoT	Privado	1
3	2018	2	Expedientes clínicos	Privado	1
4	2018	5	Criptomoneda con smartcontracts	Privado	2
5	2018	2	Dispositivos IoT	Privado	1
6	2020	6	Mercados de datos de movilidad inteligente	Híbrido	2

Tabla 3.1: Comparación de los trabajos relacionados

Estas propuestas en su mayoría proponen un modelo de dos o más capas, donde cada propuesta considera un diferente tipo de capas a las que hacen referencia, algunas dividen en capas el modelo general y otras se enfocan solo a las capas del blockchain que utilizan. Todas buscan mejorar la eficiencia y la privacidad de los datos. Algunas propuestas contemplan los contratos inteligentes otras no. Todas las propuestas están ligadas con un caso de uso específico. Ninguna de las propuestas agrega la funcionalidad de almacenamiento de documentos. Por otra parte, la propuesta de esta investigación contempla que en su primera capa los nodos voluntarios que quieran puedan unirse para contribuir a hacer más segura la información. También la propuesta tiene 4 capas reales de cadenas de bloques a diferencia de los dos de las propuestas 4 y 6.

Capítulo 4

Propuesta: Un blockchain multicapas

En este capítulo se presentan la propuesta que surgió de la investigación doctoral.

Hay 3 atributos principales deseados en todas las redes blockchain, seguridad, velocidad de transacciones y descentralización [3]. La seguridad es parte esencial para hacer que la red sea confiable. La descentralización permite que la información siempre este disponible. La velocidad de transacciones es necesaria para determinar los tipos de usos que se le puede dar a la red blockchain, existen servicios digitales que exigen una alta velocidad de transacciones y con la llegada de IoT se prevé que esta demanda aumente [72, 73]. Una parte de la seguridad esta ligada a la descentralización por las mismas ventajas que se tienen al descentralizar la información. Sin embargo a mayor descentralización menor es la velocidad de transacciones. También a mayor tamaño en los bloques menor es la velocidad de transacciones, a menos, que esté poco descentralizado. La problemática aumenta cuando agregamos la funcionalidad de contratos inteligentes y estos deben almacenarse en los bloques puesto que ocupan más espacio de bloque que una transacción común y si a esto le agregamos que diversos servicios digitales necesitan de documentos, imágenes, etc. para funcionar, se afectaría drásticamente la velocidad de transacciones.

La finalidad del modelo es ser funcional y mantener seguridad, velocidad de transacciones y descentralización para los diversos servicios digitales gubernamentales. Así mismo también se busca obtener servicios audibles, transparentes y en consecuencia confiables (ver Figura 4.1).

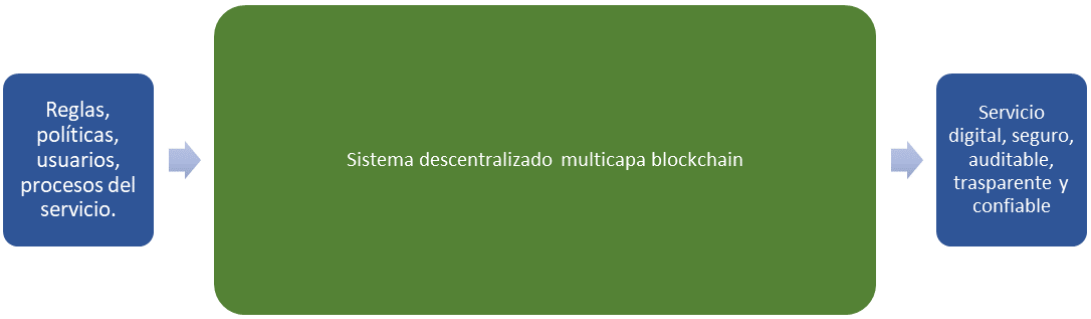

Figura 4.1: Esquema general del modelo

Analizando las necesidades de los nuevos servicios digitales se propone un modelo de 4 capas donde se dividen en los distintos tipos de información que se pueden almacenar en los bloques y que se explica a continuación.

4.1. Un modelo de 4 capas basado en blockchain

Los bloques que componen una cadena de bloques contienen diferentes tipos de información. Ejemplos de esta información son valores hash, tablas de índices, datos cifrados, información de transacciones, contratos inteligentes, entre otros. Hasta donde sabemos, los archivos no se almacenan en blockchain. Esto se debe a que el tamaño de los bloques con documentos se vuelven muy grandes y eso afecta la velocidad de translaciones [74].

Es por eso que nuestra propuesta separa la información generalmente contenida en un bloque, en diferentes. Al analizar el uso de cada pieza de información, es posible decidir qué propiedades tendrán mayor prioridad, cuales conviene descentralizar más y cuales no.

En trabajos relacionados se demostró que la separación de la información en capas permite la optimización del servicio basado en cadenas de bloques. Contemplando todas las funcionalidades que puede haber, se puede separar la información en distintas capas, según los distintos tipos de información que sean necesarios para el o los objetivos del servicio digital.

Por ejemplo, si se desea crear una capa dedicada para el almacenamiento de documentos y otra capa para contratos inteligentes, se puede optimizar el manejo de los bloques con distinta información, de tal manera que también se pueden automatizar diferentes transacciones, mientras se mantiene la conectividad, por ejemplo entre dispositivos del IoT [24].

Siguiendo este razonamiento, nuestro modelo separa la información en 4 tipos, por lo tanto se compone de las siguientes 4 capas:

- Capa 1: Index-Keys
- Capa 2: Transactions
- Capa 3: SmartContracts
- Capa 4: Files

4.1.1. Capa 1: Index-Keys

La capa 1 esta dedicada a almacenar las tablas de índices, las llaves criptograficas y los identificadores de transacciones, smartcontracts y de documentos. Los identificadores también van acompañados con los hashes de las distintas informaciones, para asegurar su integridad. La finalidad de esta capa es ser la capa con mayor descentralización, asegurando su disponibilidad e integridad. Los bloques de esta capa son los más ligeros para garantizar que sean de rápida distribución. Está capa tiene las propiedades de blockchain público, por lo que existe la posibilidad de permitir que las personas participen en la validación de los bloques, ayudando a la confirmación de bloque y a generar confianza en la información almacenada.

El algoritmo de consenso que utiliza está capa es PoW, donde se selecciona un grupo de nodos aleatoriamente para realizar el minado del bloque, que al combinarse con los bloques ligeros se puede garantizar una rápida distribución y validación de los bloques. Pow es la mejor elección de algoritmo de consenso para esta capa puesto que es capaz de alcanzar el consenso con múltiples nodos y donde no todos los nodos son de confianza.

Los bloques de esta capa se interconectan de manera escalonada a las otras tres capas. Las propiedades de la capa 1 están en la Tabla 4.1.

Propiedad	**Capa 1**
Transacciones	Alto
Consumo eléctrico	Alto
Cambio en datos	Muy difícil
Descentralización política	Si
Transparencia en la información	Si

Tabla 4.1: Propiedades de la capa 1

4.1.2. Capa 2: Transactions

La capa 2 esta dedicada a almacenar las transacciones. Almacena la información necesaria correspondiente a todas las transacciones que se realizan. Para mejorar la velocidad de transacciones esta capa es de menor descentralización que la capa 1, por lo tanto, esta capa implementará un cadena de bloques privada utilizado el algoritmo de consenso PBFT, el cual solo funciona con nodos definidos y de alta confiabilidad, los cuales necesitan estar autenticados para asegurar su legitimidad, por lo tanto, esta capa tiene las propiedades de los blockchain privados.

Los bloques de la capa 2 necesitan consultar información de los bloques de la capa 1, para validar que las transacciones que se realizan sean por usuarios legítimos y validos. Cando una transacción es concluida se retroalimenta a la capa 1 con su identificador y su hash. De esta manera se asegura que las transacciones de la capa 2 estén protegidas contra modificaciones posteriores por los nodos que se encargan de realizar las mismas transacciones.

Los bloques de la capa 2 también pueden realizar transacciones que desencadenen los contratos inteligentes que estén registrados en los bloques de la capa 3. Las propiedades de la capa 2 se pueden ver en la Tabla 4.2.

Propiedad	**Capa 2**
Transacciones	Alto
Consumo eléctrico	Bajo
Cambio en datos	Muy difícil
Descentralización política	No
Transparencia en la información	Si

Tabla 4.2: Propiedades de la capa 2

4.1.3. Capa 3: SmartContracts

En la capa 3 se almacena toda la información referente a los contratos inteligentes. Para esta capa también se utiliza el algoritmo de consenso PBFT, de la misma manera los nodos deben ser definidos y autenticados con la diferencia de configuración que para esta capa se requerirá una menor cantidad de nodos que en la capa 2, en consecuencia su comportamiento también es de tipo blockchain privada. El tamaño de bloque requerido para los smartcontracts es mayor que en la capa 2 y en compensación se requiere menos descentralización para mantener la eficiencia.

Cuando una contrato inteligente es definido se retroalimenta a la capa 1 con su identificador y su hash. De esta manera se asegura que los contratos inteligen-

tes estén protegidos contra modificaciones posteriores. También interactúa con la capa 2 puesto que los contratos inteligentes ejecutan una o varias transacciones cuando son activados. También se puede dar el caso en que sean necesarios archivos que se encuentren almacenados en la capa 4, para ello se liga el identificador del documento con su hash en el contrato inteligente. De esa forma no es necesario guardar el documento dentro del mismo bloque.

Las propiedades de la capa 3 se pueden ver en la Tabla 4.3.

Propiedad	**Capa 3**
Transacciones	Alto
Consumo eléctrico	Bajo
Cambio en datos	Muy difícil
Descentralización política	No
Transparencia en la información	Si

Tabla 4.3: Propiedades de la capa 3

4.1.4. Capa 4: Files

La capa 4 almacena los archivos en los formatos requeridos por la aplicación. Esta capa utiliza el sistema de archivos interplanetarios (IPFS), un protocolo y una red entre nodos que se utiliza para almacenar y compartir datos de forma distribuida [75].

IPFS permite a los nodos alojar y recibir contenido distribuido, de modo que cada uno de ellos posee una parte de los datos globales. Por lo tanto, cualquier nodo puede guardar un archivo y otros nodos de la red pueden encontrar y solicitar ese contenido a cualquier otro nodo. IPFS guarda los archivos por partes y mediante identificadores de contenido se pueden reestructurar los archivos. Este protocolo garantiza un alto grado de privacidad manteniendo la velocidad de transferencia.

Dependiendo de la aplicación, un nodo puede dar acceso a un archivo añadiendo el identificador de contenido en una transacción o en un contrato inteligente. Esta capa no utiliza un protocolo de consenso ya que éste no es necesario. Las propiedades de la capa 4 se pueden ver en la Tabla 4.4.

Propiedad	Capa 4
Transacciones	Medio
Consumo eléctrico	Bajo
Cambio en datos	Muy difícil
Descentralización política	No
Transparencia en la información	No

Tabla 4.4: Propiedades de la capa 4

En la Figura 4.2 se representan las cuatro capas con sus respectivos algoritmos de consenso, donde la capa 1 esta representada en color verde, la capa 2 en color amarillo, la capa 3 en color rojizo y la capa 4 en color anaranjado. Las figuras contenidas en cada capa representan nodos. Los ovalados correspondientes a la capa 1 denotando que son más en cantidad comparados con las demás capa, los círculos corresponden a nodos de la capa 2, los rectángulos a los de la capa 3 y los hexágonos a los nodos de la capa 4.

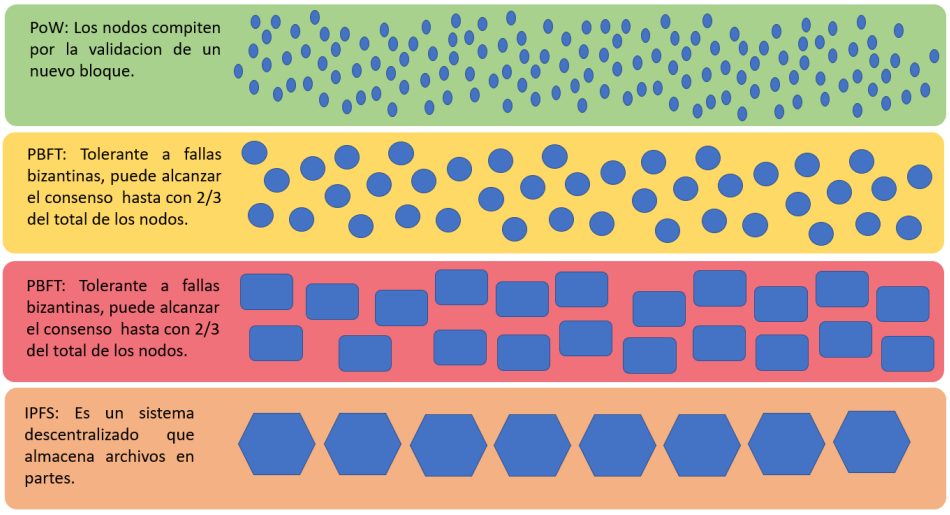

Figura 4.2: Descripción del algoritmo de consenso por cada capa

Donde la capa 1 es la capa más ligera y de mayor descentralización mientras la capa 4 apenas estaría descentralizada y es la de mayor tamaño de bloque. La capa 2 debe estar descentralizada en menor medida que la capa 1 pero en mayor medida que la capa 3 y 4, mientras que la capa 3 de estar descentralizada en mayor medida que la capa 4 pero en menor medida que en la capa 1 y 2. Esto será de gran importancia para mantener la velocidad de transacciones rápidas y la seguridad suficiente. El nivel de descentralización de la propuesta es variable puesto que dependerá de la capa o las capas que se requieran utilizar para determinados

servicios digitales.

En la Figura 4.3 se representa el nivel de descentralización de cada capa, donde la capa más ligera y de mayor generación de los bloques es la más descentralizada mientras que la capa de mayor tamaño tiene la menor descentralización posible.

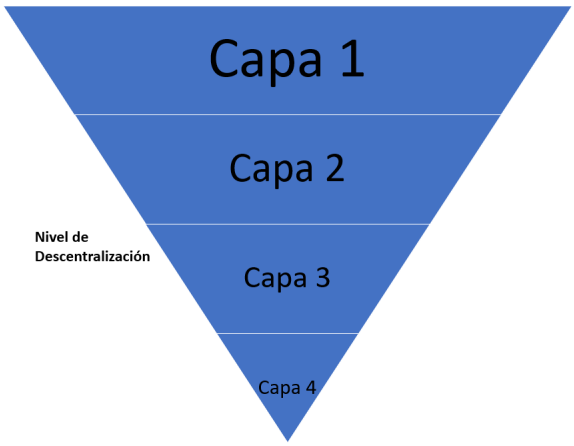

Figura 4.3: Pirámide que representa el nivel de descentralización de cada capa de la propuesta

En la Figura 4.4 se representa el tamaño de los bloques de cada capa, donde la capa más ligera es la primera mientras que la de mayor tamaño es la última.

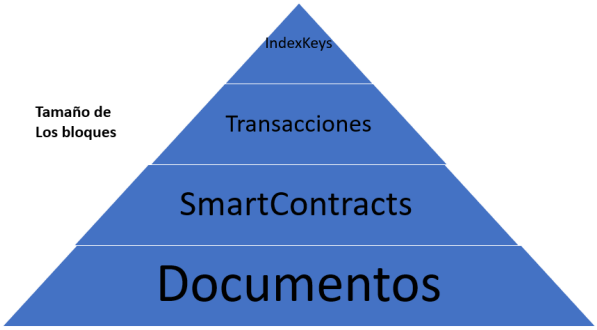

Figura 4.4: Pirámide que representa el nivel de tamaño de los bloques de cada capa de la propuesta

4.2. Estructura de los bloques

La generación de los bloques depende del caso de uso que se le quiera dar para adecuarlo a las necesidades que se quieran solventar, sin embargo, existen elemen-

tos que siempre deben estar presentes en las cadenas de bloques, con la agregación de los bloques conectores. Los bloques conectores además de los campos que normalmente tiene un bloque común se le agrega un campo extra que es un hash que lo interconecta a un bloque de otra capa diferente a la suya. En la Figura 4.5 se representa el esquema de los campos de un bloque conector y cada elemento se describe a continuación:

Block Es el número de bloque del blockchain.

Nonce Es el número entero que cambia al buscar el hash valido.

Prev Hash Es el hash valido encontrado del bloque anterior.

Hash Es la cadena hexadecimal que cumple con la dificultad y características establecidas para dar validez al bloque.

Data Son los datos que se quieren publicar en el bloque para transparentar la información.

SubPrev Hash Es el hash valido encontrado del bloque anterior pero de una capa arriba del bloque actual, que permite la conexión entre distintas capas.

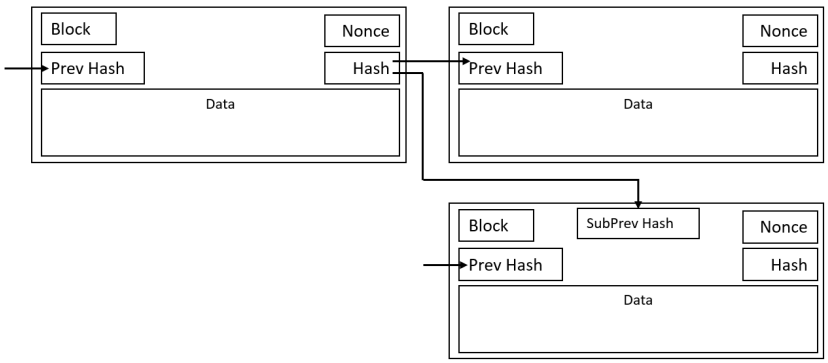

Figura 4.5: Bloque conector que conecta su bloque anterior y agrega la conexión a una capa distinta a la suya

Para optimizar el modelo multicapa se requieren diferentes tiempos de generación de los bloques de las distintas capas. En la Figura 4.6 se observa como en la primera capa se generan los bloques en mayor cantidad en comparación a las otras tres capas, la segunda con menor frecuencia que la primera, la tercera con menor frecuencia a las dos anteriores y la cuarta capa es la de menor frecuencia de todas. Lo mismo pasa con su nivel de descentralización como se mencionaba anteriormente.

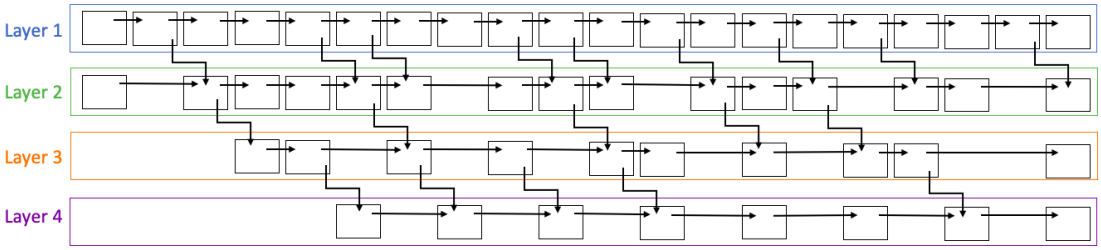

Figura 4.6: Esquema general de una posible generación de bloques en las distintas capas

Las conexiones entre los bloques se agregan dependiendo si se requiere o no almacenar información que corresponde a la siguiente capa, por ejemplo, si se agrega un usuario con sus correspondientes hashes solo se requiere de la capa 1 por lo que no se utilizan las demás capas. En cambio si se agrega una transacción entre dos usuarios se requiere de la capa 1 y 2 por lo que se enlazan los bloques participantes correspondientes. Si se requiere programar una o varias transacciones en un smart contract se enlazan las capas 1, 2 y 3 donde se agrega la información correspondiente. Si se requieren agregar documentos se enlazan las 4 capas.

Con este tipo de entrelazamiento la velocidad de transacciones depende del número de capas que utiliza, a mayor capas que utilice menor es la velocidad en confirmarse la validez de la información, mientras que a menor uso de capas que se utilicen mayor es la velocidad en que se da por buena la información.

No siempre se requerirá guardar datos en cada capa es por eso que pueden darse diferentes flujos de información dependiendo de las capas que se utilicen.

4.2.1. Flujos de información

La propuesta guarda los diferentes tipos de información en diferentes capas. Cada capa se especializa en un tipo en concreto de información relacionado a la funcionalidad del servicio. En la Figura 4.7 se indica el tipo de información que guarda cada capa.

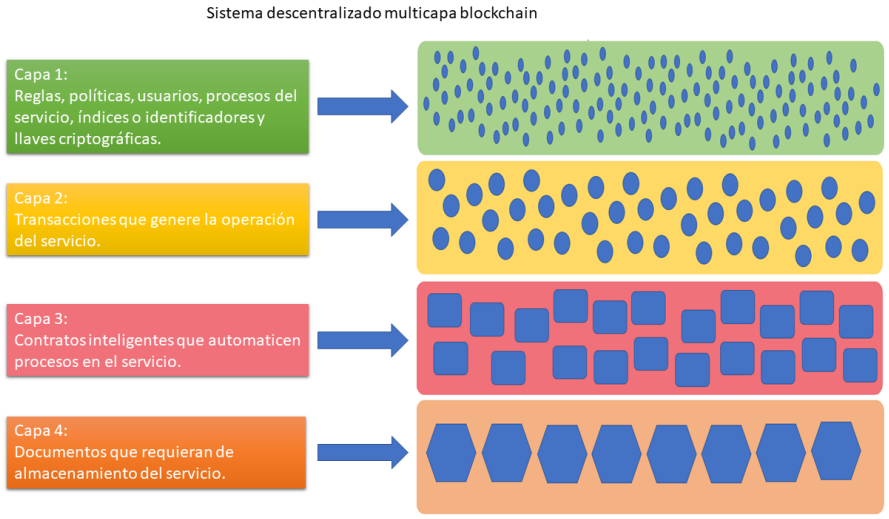

Figura 4.7: Información que procesa cada capa, donde la capa 1 es la superior y la capa 4 es la inferior

Dependiendo del servicio que se quiera montar en la arquitectura en blockchain multicapa pueden darse 4 tipos diferentes de flujos de información.

El primer caso es el más simple, ocurre cuando el servicio no requiere almacenar archivos ni va a utilizar contratos inteligentes (ver Figura 4.8). En este caso se utilizan las primeras dos capas, la primera para almacenar los datos operativos del servicio y la segunda para procesar y almacenar las transacciones que ocurran en el servicio. La capa 1 que tiene los datos operativos genera las llaves criptográficas que se requieran y detona las transacciones en la capa 2, las cuales una vez procesadas informan a la capa 1 para que guarde sus respectivos índices.

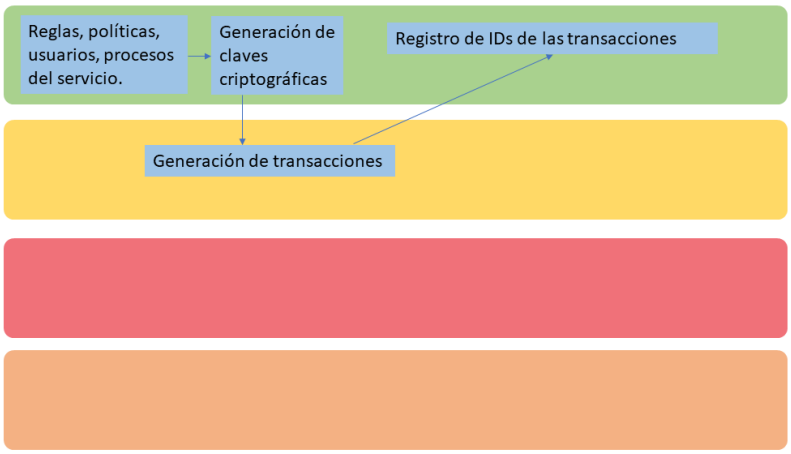

Figura 4.8: Flujo 1: se utilizan las primeras dos capas

El segundo caso ocurre cuando no es necesario el almacenamiento de archivos, por lo tanto la capa 4 no se utiliza. La capa 1 de la misma manera genera las llaves criptográficas que requiera y genera las transacciones que se requieran, que a su vez pueden generar contratos inteligentes en la capa 3. Dichos contratos tiene como respuestas una o varias transacciones que genera en la capa 2 y que dan retroalimentación a la capa 1 con sus índices para ser almacenados (ver Figura 4.9).

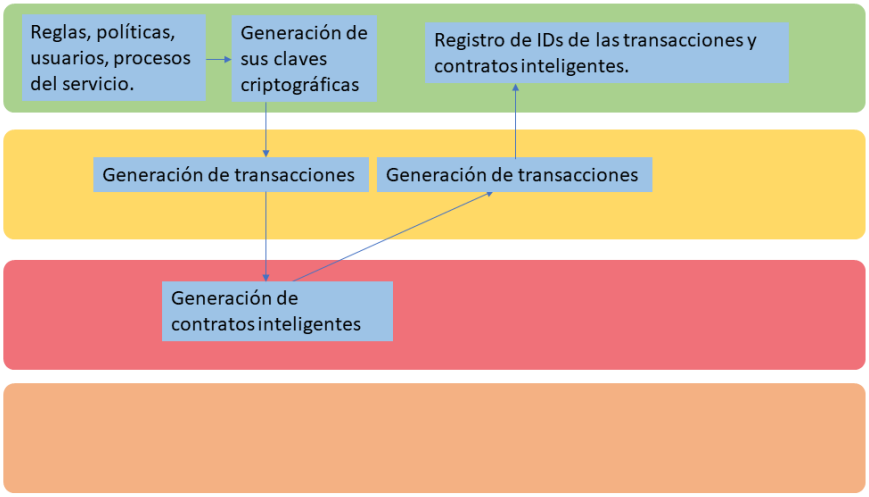

Figura 4.9: Flujo 2: se utilizan las capas 1, 2 y 3

El tercer caso ocurre cuando se requiere del almacenamiento de archivos pero no usaran de contratos inteligentes. Entonces la capa 1 que genera las llaves criptográficas para posteriormente generar transacciones en la capa 2, que una vez finalizadas guardan los archivos necesarios en la capa 4 y reporta los respectivos índices a la capa 1 para ser archivados (ver Figura 4.10).

Finalmente, el cuarto caso ocurre cuando son utilizadas todas las capas. Es cuando un servicio usara contratos inteligentes y requiere del almacenamiento de archivos. En este caso la capa 1 que contiene los datos operativos del servicio genera sus respectivas llaves criptográficas para posteriormente generar transacciones en la capa 2 que a su vez generan contratos inteligentes en la capa 3 donde se generan archivos para guardar en la capa 4 y generan una o varias transacciones más en la capa 2 y dichas transacciones guardan en la capa 1 los respectivos índices (ver Figura 4.11).

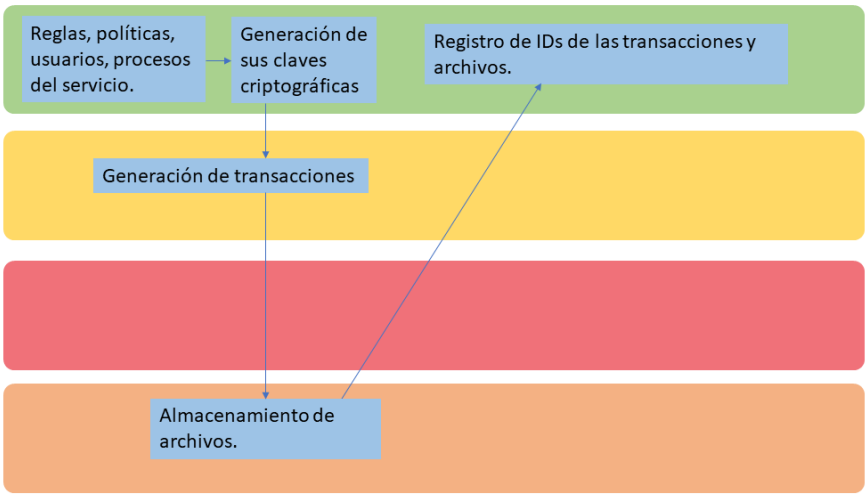

Figura 4.10: Flujo 3: se utilizan las capas 1, 2 y 4

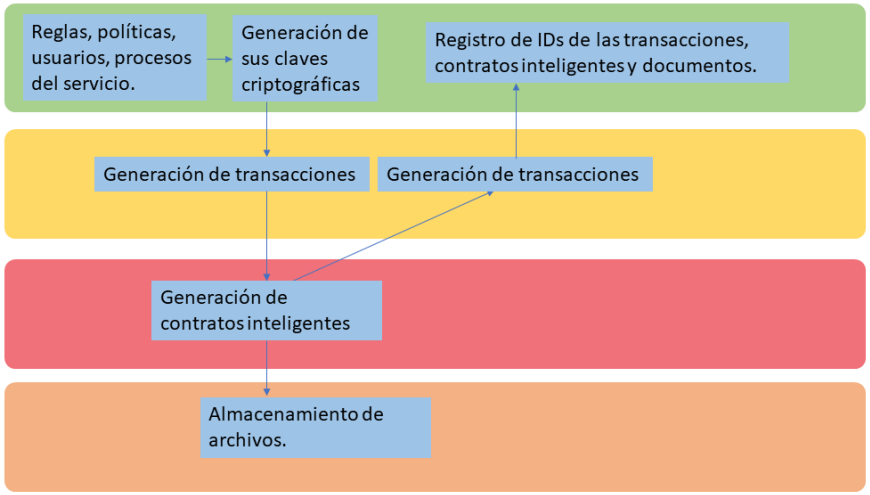

Figura 4.11: Flujo 4: se utilizan las 4 capas

Al implementar esta estructura entre bloques y capas, la velocidad de las transacciones depende del número de capas que utilice la aplicación. Así, cuanto mayor sea el número de capas utilizadas, menor será la velocidad de validación de la información de los bloques. Por el contrario, cuanto menor sea el número de capas utilizadas, mayor será la velocidad de validación de la información de los bloques.

En la Figura 4.12 se pueden observar las 4 capas y un ejemplo de conexiones escalonadas entre bloques de las cuatro capas, donde los bloques superiores corresponden a los de la capa 1 y el bloque inferior a la capa 4.

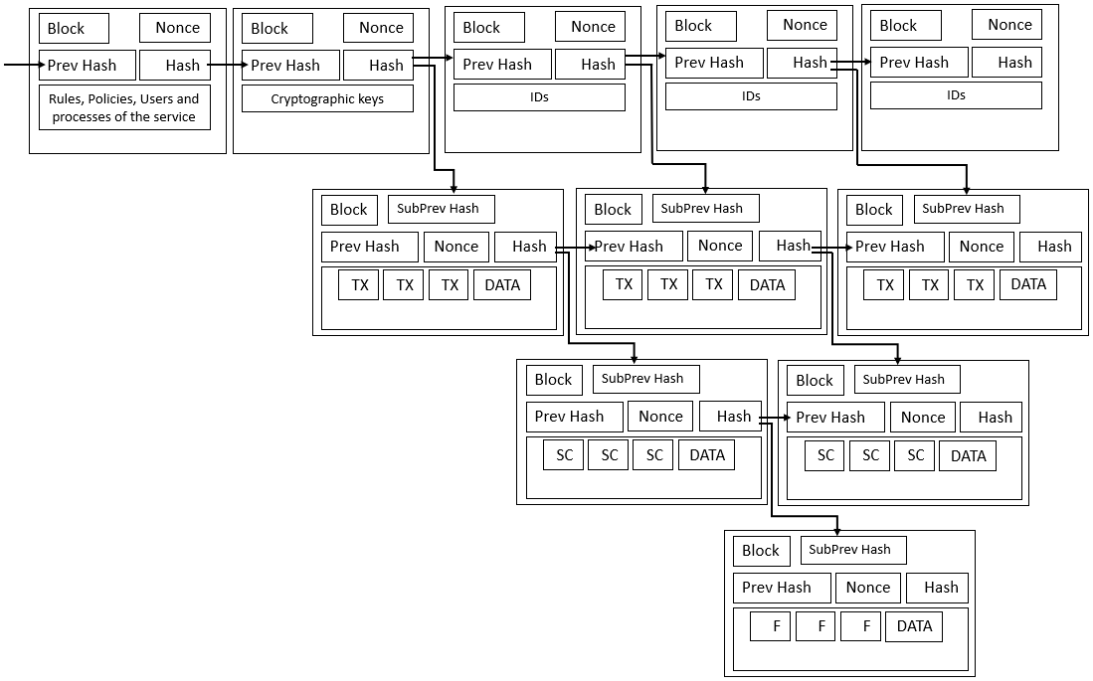

Figura 4.12: Esquema de contenido general de los bloques

En el capítulo 5, se definen las propiedades, el modelado de la propuesta y un caso de estudio que ayuda a ejemplificar el funcionamiento de la arquitectura multicapas. Es de recalcar que la arquitectura multicapas es genérica y es hasta cuando se va a implementar en un caso de uso cuando se definen determinadas restricciones o consideraciones, de la misma manera que se define cual de los 4 posibles flujos de información sera el que se utilizara.

Capítulo 5

Resultados

5.1. Validación de integridad de los bloques

La propuesta de un modelo multicapas representa una posible solución a la problemática inicial planteada, permite descentralizar aun más la información y al separarla se realiza una mejor gestión para no disminuir la eficiencia del sistema y aumentar la funcionalidad del mimo sin sacrificar velocidad de transacciones.

La integridad de la información es un objetivo muy importante y es considerado crucial en los sistemas actuales, esto debido a que si no se garantiza la integridad de la información de nada nos sirve tener un sistema eficaz y/o eficiente. En el caso de la arquitectura propuesta mediante múltiples capas de blockchain la integridad de la información es heredada de las mismas características originales de las blockchain.

En el caso de las capas 2, 3 y 4 al ser de tipo blockchain privada la integridad de la información recae principalmente en los nodos de confianza que serán los encargados de procesar la información, y con el conjunto de técnicas criptográficas propias de los algoritmos de consenso que utilizan es posible identificar con facilidad cuando un nodo quiere actuar de manera deshonesta.

En el caso de la capa 1 al ser de tipo blockchain pública, esta si requiere de atención. Debido a que no se conoce el origen de los nodos que se unen a la red para verificar y el objetivo es que cualquiera que quiera participar pueda hacerlo, la información es susceptible a ser modificada maliciosamente, es por ello que se opto por utilizar el algoritmo de consenso PoW (proof of work) que ha demostrado ser fuerte frente a estos acontecimientos. La comprobación de la integridad de esta capa vale la pena ser analizada a detalle y es retomada del trabajo [1] donde es presentada la primer criptomoneda y la más popular, Bitcoin. En mencionado

trabajo se comprueba que es muy difícil modificar maliciosamente la información contenida en los bloques y lograr hacerla pasar por datos validos.

A continuación la probabilidad de que un atacante alcance la generación de bloques honesta. En el hipotético caso de que un atacante este tratando de generar una cadena alternativa con la finalidad de hacer dicha cadena como la valida ante los demás nodos.

$x =$ es la probabilidad de que un nodo honesto sea quien valide y agregue a la cadena el siguiente bloque.

$y =$ es la probabilidad de que un nodo tramposo sea quien valide y agregue a la cadena el siguiente bloque.

$y_n =$ es la probabilidad de que el nodo tramposo alcance a la cadena integra desde n bloques atrás

$$x_n = \left\{ \begin{array}{ll} 1 & si\, x \leq y \\ (y/x)^n & si\, x > y \end{array} \right\}$$

Asumiendo que el poder computacional (y no la cantidad de nodos) de los participantes honestos es mayor al poder computacional de los participantes que quieren modificar la información a su beneficio (nodos deshonestos), es comprobable que la probabilidad cae de forma exponencial a medida que aumenta el número de bloques que el atacante tiene que alcanzar. Con las probabilidades en su contra, si no tiene un golpe de suerte que lo haga avanzar desde el principio, sus oportunidades se irán desvaneciendo a medida que se va quedando atrás.

El tiempo que necesitaría esperar un determinado receptor para estar seguro que una transacción es valida y que no va a cambiar se obtiene de la siguiente manera: se asume que el emisor es un nodo malicioso que quiere que el receptor crea (al menos por un tiempo) que la información maliciosa que ha logrado agregar a un bloque (por ejemplo intentar gastarse un mismo token dos veces) cambien dicha información a su conveniencia. El receptor eventualmente recibirá un aviso cuando esto suceda, pero el emisor espera que ya sea demasiado tarde. Una vez que la transacción se ha efectuado, el emisor malicioso debe apurarse y en secreto validar nuevos bloques en una cadena paralela que contiene una versión alternativa de su transacción. El receptor espera hasta que la transacción se ha añadido al bloque y n bloques se hayan agregado tras él. No sabe la cantidad de progreso que ha realizado el atacante, pero asumiendo que los bloques honestos han tomado la media de tiempo esperada por bloque, el potencial de progreso del atacante será una distribución de Poisson con un valor esperado:

$$\lambda = n\frac{y}{x}$$

Para obtener la probabilidad de que un nodo deshonesto pueda agregar los últimos bloques, multiplicamos la densidad de Poisson para cada aumento en el progreso que podría haber realizado, por la probabilidad que podría haber alcanzado desde ese punto:

$$\sum_{k=0}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \cdot \left\{ \begin{array}{ll} (y/x)^{n-k} & si\, k \leq n \\ 1 & si\, k > n \end{array} \right\}$$

Reordenándola para evitar la suma infinita:

$$1 - \sum_{k=0}^{n} \frac{\lambda^k e^{-\lambda}}{k!} \left(1 - (y/x)^{(n-k)}\right)$$

A continuación se calculan los resultados en incrementos desde $y = 0.05$ hasta $y = 0.25$ que representa como máxima probabilidad del 25 % de nodos deshonestos agreguen el siguiente bloque, deteniéndose cuando el resultado sea < 0.01 que significa que la probabilidad es de menos del 1 % de que la cadena deshonesta le gane a la cadena honesta.

Se calculan los resultados de probabilidades en los primeros 10 bloques, en incrementos de 0.05 desde $y = 0.05$ hasta $y = 0.25$ que representa como máxima probabilidad del 25 % de nodos deshonestos agreguen el siguiente bloque (ver la Tabla 5.1).

No Bloque	**5 %**	**10 %**	**15 %**	**20 %**	**25 %**
1	10.1203	20.45	30.9698	41.5899412	52.23124
2	1.2646	5.0977	11.5041	20.39285	31.54438
3	**0.1641**	1.3172	4.42278	10.32416	19.611529
4	0.0216	**0.3455**	1.725299	5.299784	12.3512397
5	0.0028	0.09136	**0.678378**	2.741554	7.8357012
6	0.0003	0.02428	0.268047	1.425061	4.9942598
7	0.00005	0.006473	0.106264	**0.743191**	3.193482
8	0.000006	0.0017299	0.0422269	0.3885	2.046816
9	0.0000009	0.0004631	0.016809	0.203	1.314218
10	0.00000001	0.0001241	0.00670056	0.01066	**0.84501**

Tabla 5.1: Probabilidad de éxito de un atacante disminuyendo conforme se van generando más bloques en la cadena

Se observa que en el caso de $y = 0.05$ en el bloque número 3 el atacante tiene menos del 1 % de éxito y lo mismo ocurre en el bloque 4 cuando $y = 0.10$, para

$y = 0.15$ el bloque es el número 5, mientras que para $y = 0.20$ es en el bloque 7 y para $y = 0.25$ es en el bloque 10.

El análisis muestra que aunque el o los atacantes tengan buena probabilidad inicial de vulnerar información, a medida que se van agregando más bloques a la cadena, la probabilidad de tener éxito disminuye exponencialmente (ver Figura 5.1). También se debe resaltar que para que sea valido el análisis se supone que el atacante debe mantener su poder de computo inicial del ataque y continuar sin ser detectado.

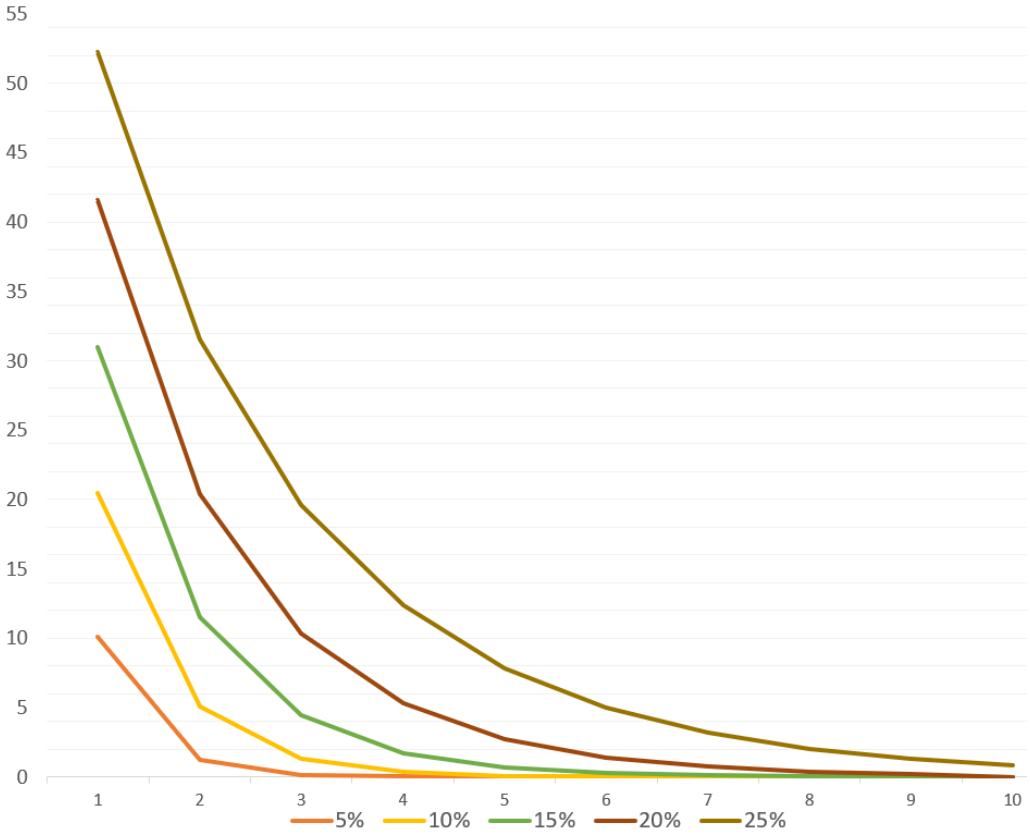

Figura 5.1: Tendencia exponencial a la baja conforme se generan más bloques

En una red blockchain bien descentralizada es difícil que un solo nodo tenga el 25 % de poder de procesamiento o más, pero aun así se realizaron los cálculos suponiendo que el atacante inicie con 30 %, 35 %, 40 % y 45 % y el bloque en el cual tendría el atacante menos del 1 % de éxito seria el bloque 16, 27, 58 y 220 respectivamente.

y = 0.30
n = 14 resultado = 0.0133922629836278240

n = 15 resultado = 0.0101007621731109090
n = 16 resultado = 0.0076218996722338698

y = 0.35
n = 25 resultado = 0.0123344309003193130
n = 26 resultado = 0.0105008448836457210
n = 27 resultado = 0.0089412262267228714

y = 0.40
n = 56 resultado = 0.0110752938128566660
n = 57 resultado = 0.0102888832266503500
n = 58 resultado = 0.0095586163575425206

y = 0.45
n = 218 resultado = 0.0102844661876967370
n = 219 resultado = 0.0100884749665984280
n = 220 resultado = 0.0098962398140198023

Por lo tanto, desde $y = 0.05$ hasta $y = 0.45$ los resultados son los siguientes:

y = 0.05 n = 3
y = 0.10 n = 4
y = 0.15 n = 5
y = 0.20 n = 7
y = 0.25 n = 10
y = 0.30 n = 16
y = 0.35 n = 27
y = 0.40 n = 58
y = 0.45 n = 220

Se observa como al acercarse al 50 % cada vez se requieren más bloques para lograr disminuir la probabilidad de éxito a menos del 1 %,por lo que también se encontró un comportamiento exponencial a la alza, cuando $y \geq 0.5$ la integridad de los bloques es comprometida puesto que significa que la mitad o más de la mitad de los nodos tienen el mayor poder de procesamiento, es por ello, que la descentralización es de gran importancia, a mayor descentralización mayor la dificultad para que un conjunto de nodos logren sobrepasar el 50 % de poder de procesamiento.

Por otra parte se observa que la probabilidad de que atacantes con suficiente poder computacional, al intentar modificar la información y querer hacerla pasar como buena, deberían tener varios golpes de suerte seguidos y encontrar los hashes validos más rápido que los demás nodos para dar la cadena deshonesta como valida y que el tiempo juega un papel en su contra, porque a medida que la cadena crece su probabilidad va disminuyendo exponencialmente y eventualmente ya no

representaran un riesgo y eso suponiendo que no sean detectados y bloqueados antes.

En la siguiente sección se establecen las definiciones y propiedades que definen la arquitectura única de la propuesta multicapas.

5.2. Modelado de la propuesta

En esta sección, presentamos el modelado utilizando teoría de conjuntos para definir las propiedades de la propuesta. Para esto, primero presentaremos la notación utilizada para formalizar nuestra propuesta. Esto incluye cada uno de los elementos del modelo presentado en 4.1. El elemento principal de nuestra propuesta es una capa. Así presentamos la siguiente definición.

Definición 5.1 (Capa) *Una Capa es una separación lógica en nuestro modelo que procesa los tipos de información predefinidos. También incluye un conjunto de bloques. Se denota como C.*

Cada capa es soportada por nodos que se encargan de almacenar una copia de las cadenas.

Definición 5.2 (Nodos) *Los nodos almacenan colectivamente bloques de la cadena de bloques y son responsables de la fiabilidad de los datos almacenados. Se denota como p.*

Los bloques son donde la información es guardada y se define de la siguiente manera.

Definición 5.3 (Bloque) *Un bloque es un componente que almacena diferentes tipos de información y es generado por los p. Los bloques de la capa 1 se denotan por x, los de la capa 2 por y, los de la capa 3 por z y los de la capa 4 por d.*

En los bloques se van guardando las transacciones que se van generando y estas se definen de la siguiente forma.

Definición 5.4 (Transacción) *Una transacción es una transferencia de valor y crea relaciones en diferentes tipos de datos, se transmite a los p_i para ser validado. Se denota como T_y.*

Para controlar la velocidad con la que los bloques son generados es necesario agregar la unidad de tiempo que permitirá configurar la generación acorde a las necesidades del servicio a utilizar, por lo tanto la unidad de tiempo se define de la siguiente manera.

Definición 5.5 (Unidad de Tiempo) *Una unidad de tiempo es el tiempo que toma la generación de un bloque. Se denota como ut.*

Para la capa 3 es necesario crear contratos inteligentes que van a generar una o varias transacciones de manera autónoma.

Definición 5.6 (Smart contract) *Un contrato inteligente es un conjunto de transacciones (T_y) que se ejecutarán en un tiempo determinado (ut) una vez que se cumplen las condiciones que son predeterminadas. Un contrato inteligente es previamente aceptado y firmado por los p_i que participan en él. Se denota como SC.*

Los datos necesarios para que un servicio digital pueda ser operativo debe contemplarse y se define de la siguiente forma.

Definición 5.7 (Conjuntos de datos de aplicación) *Los datos de aplicación pueden ser conjuntos de reglas, usuarios, políticas y procesos que contienen la configuración del servicio y que se añaden a los bloques de la capa 1. Se denotan como x_i^r.*

Es importante guardar en cada bloque el tiempo exacto en el que el bloque es creado, debido a que los bloques quedan ordenados cronológicamente y este campo se define de la siguiente manera.

Definición 5.8 (Timestamp) *Una marca de tiempo es un valor numérico que indica la hora exacta en que se generó un bloque. Se denota como timestamp.*

Otra configuración importante es la descentralización que va a obtener cada capa, puesto que de esta depende una parte de la seguridad y una parte de la velocidad de transacciones por lo que se define de la siguiente forma.

Definición 5.9 (Nivel de Descentralización) *El nivel de descentralización se refiere al número de nodos necesarios para que un protocolo de consenso funcione correctamente. Cuanto mayor sea el número de nodos, mayor será el nivel de descentralización. Se denota como dc.*

Finalmente se define el modelo que contempla las 4 capas propuestas.

Definición 5.10 (Modelo) *Un modelo denotado como P contiene un conjunto de capas $C_1, ..., C_4$, donde C_1 es un conjunto de bloques que pertenecen a la capa 1, C_2 es un conjunto de bloques que pertenecen a la capa 2, C_3 es un conjunto de bloques que pertenecen a la capa 3 y C_4 es un conjunto de bloques que pertenecen a la capa 4.*

Blockchain utiliza criptografía asimétrica, por lo que los nodos utilizan un par de claves. Entonces, las claves de un nodo p_1 son k_{p1} (clave pública) y k_{p1}^{-1} (clave privada). Blockchain también utiliza una función hash h que, dado un mensaje m, creará el resultado hash m'. Esta operación se denomina $h(m) = m'$.

A partir de estas definiciones, se definen las siguientes propiedades. Éstas describen la generación de los bloques y la conexión entre las capas inferiores.

La tasa de transacción es la capacidad de procesar diferentes cantidades de información en un periodo de tiempo determinado, a mayor cantidad de información se dificulta mantener y/o disminuir el tiempo necesario para su procesamiento. Es por ello que la tasa de transacción es afectada por el tamaño de los bloques, el nivel de descentralización y la velocidad de generación de los bloques, las siguientes propiedades hacer referencia a cada una de estas.

Propiedad 5.1 ***Velocidad de generación de los bloques***

Dependiendo del servicio digital se debe configurar una generación de bloques adecuada, sin embargo, siempre se debe cumplir lo siguiente:

En la Capa 1 el conjunto $C_1 = \{x_1, x_2, ...x_n\}$ *tal que* x_i *son bloques que se generan secuencialmente cada* ut_1 *unidades de tiempo.*

En la Capa 2 el conjunto $C_2 = \{y_1, y_2, ...y_n\}$ *tal que* y_i *son bloques que se generan secuencialmente cada* ut_2 *unidades de tiempo.*

En la Capa 3 el conjunto $C_3 = \{z_1, z_2, ...z_n\}$ *tal que* z_i *son bloques que se generan secuencialmente cada* ut_3 *unidades de tiempo.*

En la Capa 4 el conjunto $C_4 = \{d_1, d_2, ...d_n\}$ *tal que* d_i *son bloques que se generan secuencialmente cada* ut_4 *unidades de tiempo.*

Por tanto, los bloques se generan con la siguiente relación temporal $ut_1 > ut_2 \geq ut_3 > ut_4$.

Esta propiedad se logra mediante el protocolo de consenso utilizado en cada capa.

En C_1 se usa PoW como algoritmo de consenso y tiene p_i nodos, entonces la validación de un bloque x_{n+1} creado por p_1 requiere:

- En el mejor caso, cuando la mayoría de los nodos están de acuerdo, se requiere $\frac{p_i-1}{2}$ mensajes enviados por p_1.
- En el peor caso, donde los nodos deberán estar de acuerdo, se requiere $(p_i) - 1$ mensajes enviados por p_1 a todos los nodos de la capa.

En C_2 se usa PBFT as como algoritmo de consenso y tiene p_j nodos, entonces la validación de un bloque y_j creado por p_1 requiere:

- En el mejor caso, cuando la mayoría de los nodos están de acuerdo, se requiere $\frac{p_j-1}{2}$ mensajes enviados por p_1.
- En el peor caso, 2/3 de los nodos deben estar de acuerdo, $\frac{2p_j-1}{3}$ mensajes enviados por p_1 a esos nodos.

En C_3 tambien se usa PBFT as como algoritmo de consenso y tiene p_s nodos, entonces la validación de un bloque z_s creado por p_1 requiere:

- En el mejor caso, cuando la mayoría de los nodos están de acuerdo, se requiere $\frac{p_s-1}{2}$ mensajes enviados por p_1.

- En el peor caso, 2/3 de los nodos deben estar de acuerdo, $\frac{2p_s-1}{3}$ mensajes enviados por p_1 a esos nodos.

En C_4 se utiliza IPFS y no requiere de algoritmo de consenso porque solo almacena los archivos generados de manera descentralizada, pero si tiene p_e nodos. Entonces al menos nodos deben estar disponibles entre todos los nodos de la capa $p_e - ((p_e) - 2)$.

Por lo tanto en el peor escenario es $(p_i) - 1 > \frac{2p_j-1}{3} \geq \frac{2p_s-1}{3} > p_e - ((p_e) - 2)$ por lo tanto la propiedad 5.1 se cumple, los bloques de la capa 4 se validarán y agregarán a la cadena más rápido que los bloques de la capa 3. Los bloques de la capa 3 y 2 tendrán un comportamiento similar, aunque los bloques de la capa 3 tendrán menos nodos en comparación con la capa 2, por lo que deberían agregar los bloques más rápido. Finalmente, los bloques de la capa 1 tardarán más en validarse y agregarse a la cadena en comparación con las otras capas.

El nivel de descentralización se refiere a dos aspectos. Primero, es la cantidad de nodos que mantienen una copia completa de la información (descentralización arquitectónica) y, segundo, cómo estas copias se mantienen actualizadas. Por tanto, esta propiedad depende del protocolo de consenso y del tipo de blockchain que se utilice en cada capa.

Propiedad 5.2 ***Nivel de descentralización***

Sea C_1 la capa 1, C_2 la capa 2, si x_i en C_1 y y_j en C_2 entonces $\forall x : f(x_i) = x_{i+1}$ donde f recibe un bloque x y genera el siguiente bloque x_{i+1} ambos en C_1, $\exists x : g(x_i) = y_j$ donde g une un bloque x_i en C_1 con un bloque y_j en C_2. Ambas funciones f y g generan el nivel de descentralización dc_1.

Sea C_2 la capa 2, C_3 la capa 3, si y_i en C_2 y z_j en C_3 entonces $\forall y : f(y_i) = y_{i+1}$ donde f recibe un bloque y y genera el siguiente bloque y_{i+1} ambos en C_2, $\exists y : g(y_i) = z_j$ donde g une un bloque y_i en C_2 con un bloque z_j en C_3. Ambas funciones f y g generan el nivel de descentralización dc_2.

Sea C_3 la capa 3, C_4 la capa 4, si z_i en C_3 y d_j en C_4 entonces $\forall z : f(z_i) = z_{i+1}$ donde f recibe un bloque z y genera el siguiente bloque z_{i+1} ambos en C_3, $\exists z : g(z_i) = d_j$ donde g une un bloque z_i en C_3 con un bloque d_j en C_4. Ambas funciones gf y g generan el nivel de descentralización dc_3.

Sea C_4 la capa 4 y d_j en C_4 entonces $\forall d : f(d_i) = d_{i+1}$ donde f recibe un bloque d y genera el siguiente bloque d_{i+1} ambos en C_4. La función f genera el nivel de descentralización dc_4.

Por tanto, los niveles de descentralización cumplen la siguiente relación $dc_1 > dc_2 > dc_3 > dc_4$.

Si C_1, C_2, C_3 y C_4 tienen p_i, p_j, p_s y p_e nodos respectivamente, entonces cada uno tendrá una copia actualizada de la cadena de bloques. Esto se logra distribuyendo un nodo válido a todos los nodos vecinos.

- En el peor de los casos, cada nodo envía el bloque validado (x_i, y_i, z_i y d_i) al resto de los nodos. Si todos los nodos realizan la misma acción, el total de bloques validados enviados es $p(p_n) - 1$.

Pero asumiendo que $i > j > s > e$, siendo lo mejor para cada algoritmo de consenso de cada capa, entonces $p_i(p_n) - 1 > p_j(p_n) - 1 > p_s(p_n) - 1 > p_e(p_n) - 1$ por lo tanto la propiedad 5.2 se cumple.

El tamaño de bloque influye significante en la rápida distribución a todos los nodos, es por ello que dependiendo del servicio digital se debe configurar un tamaño de bloque adecuado, pero se debe cumplir lo siguiente.

Propiedad 5.3 ***Tamaño de bloques***

Sea x_i en C_1, y_i en C_2, z_i en C_3 y d_i en C_4 entonces el tamaño máximo de la información contenida dentro de los bloques, tiene un límite tal que $|x_i| < |y_i| < |z_i| < |d_i|$.

Usualmente el tamaño del bloque lo define el diseñador de la aplicación. Una aplicación puede decidir utilizar PoW para incluir una imagen como parte de la transacción, con todas las implicaciones en tiempo y costo que podría tener esta decisión. En nuestro modelo, se obliga a que cada capa almacenará cierto tipo de información en cada bloque, por lo tanto, limitará su tamaño y se recomiendan los siguientes:

- C_1 almacena incides y llaves criptográficas en los bloques x_i que deben tener un tamaño de 0.5 MB a 1 MB como máximo.
- C_2 almacena las transacciones (T_i) en los bloques y_i que deben tener un tamaño de 1.1 MB a 2 MB como máximo.

- C_3 almacena los smartcontracts (SC) en los bloques z_i que deben tener un tamaño de 2.1 MB a 5 MB como máximo.
- C_4 almacena los archivos en los bloques x_i que deben tener un tamaño de 5.1 MB a 10 MB como máximo.

Si tomamos el peor de los casos en cada una de las capas, entonces $1MB < 2MB < 5MB < 10MB$, por lo tanto, se cumple la propiedad 5.3.

Con estas propiedades se limita el número de bloques por capa, así como su tamaño máximo. De este modo, se asegura el nivel de descentralización en cada capa y la tasa de transacción de los bloques.

Dependiendo del caso de uso, será necesario utilizar un número diferente de capas. La siguiente sección describe la arquitectura general y los diferentes casos en los que no se utilizan todas las capas.

5.3. Arquitectura general de flujos de información

En esta sección, se presenta la arquitectura general del modelo mostrando cómo el número de capas a utilizar depende de la información creada por una aplicación específica. Esto hace que el modelo sea lo suficientemente flexible como para ser utilizado en una diversidad de escenarios. Para demostrarlo, se presentan 4 flujos de información diferentes que representan los casos de uso más comunes. Es importante mencionar que estos casos son representativos pero no restrictivos. Estos casos se presentan como diagramas de secuencia en los que el flujo de mensajes entre entidades se presenta con flechas. Las entidades son el administrador del sistema y las 4 capas definidas en la Def. 5.1.

En el primer escenario, la aplicación requiere almacenar los índices de llaves y transacciones por lo que no es necesario almacenar archivos y contratos inteligentes (ver Figura 5.2). En este caso se utilizan las dos primeras capas, la primera para almacenar los datos operativos del servicio y la segunda para procesar y almacenar las transacciones que se producen en el servicio. La capa 1, que tiene los datos operativos, genera las claves criptográficas necesarias y detona las transacciones en la capa 2, que, una vez procesadas, informan a la capa 1 para que almacene sus respectivos índices.

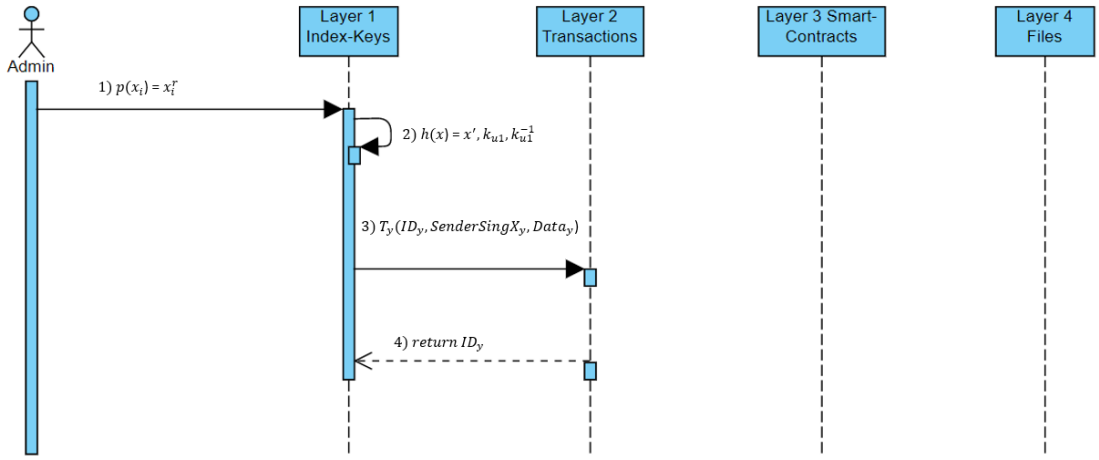

Figura 5.2: Caso 1: Llaves e índices y transacciones

A continuación se describe el proceso descrito en la figura 5.2.

1. El administrador registra y configura el conjunto de las reglas, políticas, usuarios y procesos del servicio en un bloque (x_i^r).

2. Se generan las llaves criptográficas necesarias, como hashes $(x^{'})$, llaves públicas k_{u1} y privadas k_{u1}^{-1} necesarias para asegurar la integridad y seguridad de la información.

3. Los usuarios autorizados para realizar transacciones realizan transacciones firmadas con los datos complementarios necesarios, por lo tanto se genera la transacción T_y con los parámetros mínimos, identificado de la transacción ID_y, la firma de emisor $(SenderSingX_y)$ y los datos de la transacción $(Data_y)$.

4. Una vez realizada la transacción se regresa el identificador de la misma a la capa 1 para ser guardado ID_y.

En el segundo escenario, el almacenamiento de archivos no es necesario, entonces la capa 4 no se utiliza. La capa 1 genera las claves criptográficas y las transacciones necesarias en la capa 2, que a su vez generarán contratos inteligentes en la capa 3. Posteriormente, estos contratos crearán una o varias transacciones en la capa 2 y que enviarán índices para ser almacenados en la capa 1 (ver Figura 5.3).

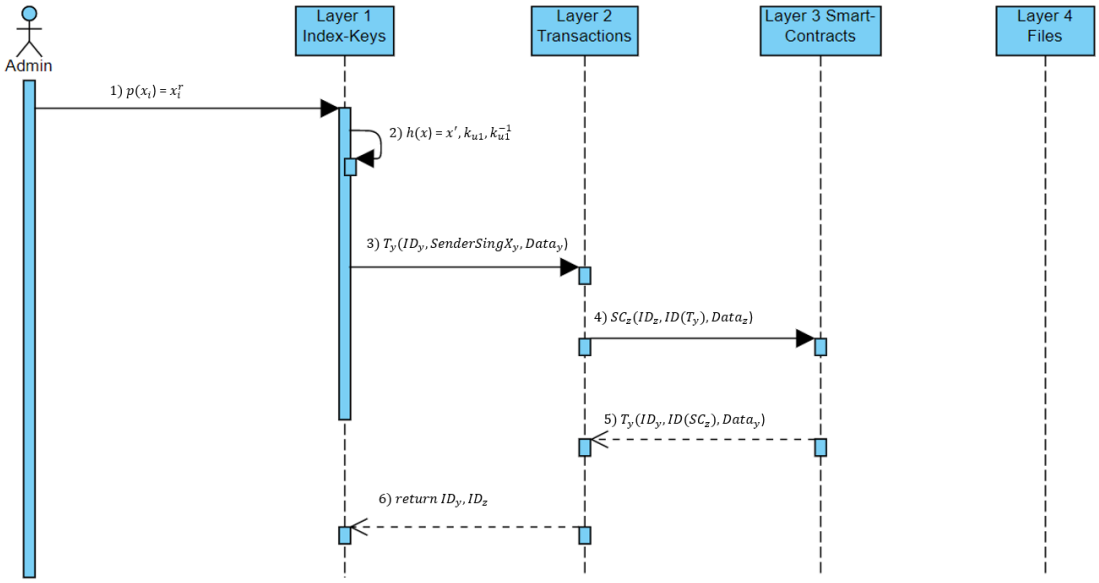

Figura 5.3: Caso 2: Llaves e índices, transacciones y smartcontracts

A continuación se describe el proceso descrito en la figura 5.3.

1. El administrador registra y configura el conjunto de las reglas, políticas, usuarios y procesos del servicio en un bloque (x_i^r).

2. Se generan las llaves criptográficas necesarias, como hashes ($x^{'}$), llaves públicas k_{u1} y privadas k_{u1}^{-1} necesarias para asegurar la integridad y seguridad de la información.

3. Los usuarios autorizados para realizar transacciones realizan transacciones firmadas con los datos complementarios necesarios, por lo tanto se genera la transacción T_y con los parámetros mínimos, identificado de la transacción ID_y, la firma de emisor ($SenderSingX_y$) y los datos de la transacción ($Data_y$).

4. Algunas transacciones pueden generar contratos inteligentes SC_z para automatizar futuras transacciones en los servicios, que como parámetros mínimos lleva el identificador del contrato inteligente (ID_z), información de la transacción que lo genera ($ID(T_y)$) y los datos del contrato inteligente $Data_z$.

5. Los contratos inteligentes generan una o varias transacciones al ejecutarse, por cada transacción T_y con los parámetros mínimos, identificado de la transacción ID_y, información del contrato inteligente que genera la transacción ($ID(SC_z)$) y los datos de la transacción ($Data_y$).

6. Una vez realizadas las transacciones se regresan los identificadores de las transacciones ID_y y contratos inteligentes ID_z a la capa 1 para ser guardados.

En el tercer escenario, el almacenamiento de archivos es necesario pero los contratos inteligentes no lo son. Entonces, la capa 1 genera claves criptográficas para posteriormente generar las transacciones necesarias en la capa 2. Una vez completadas, los archivos necesarios se almacenan en la capa 4 e informan de los índices correspondientes a la capa 1 para que los almacene (ver Figura 5.4).

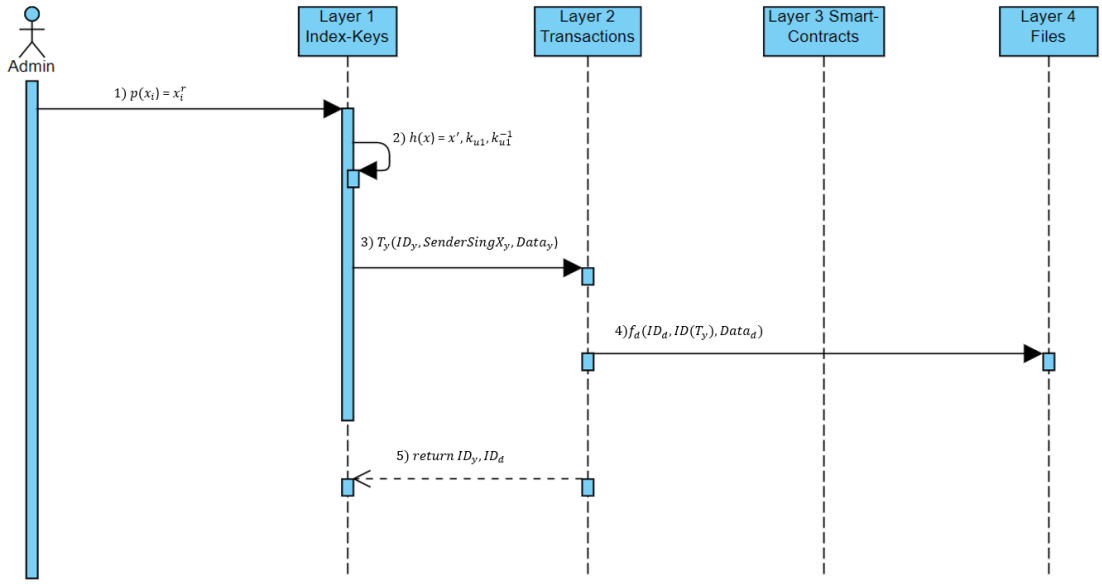

Figura 5.4: Caso 3: Llaves e índices, transacciones y archivos

A continuación se describe el proceso descrito en la figura 5.4.

1. El administrador registra y configura el conjunto de las reglas, políticas, usuarios y procesos del servicio en un bloque (x_i^r).

2. Se generan las llaves criptográficas necesarias, como hashes $(x^{'})$, llaves públicas k_{u1} y privadas k_{u1}^{-1} necesarias para asegurar la integridad y seguridad de la información.

3. Los usuarios autorizados para realizar transacciones realizan transacciones firmadas con los datos complementarios necesarios, por lo tanto se genera la transacción T_y con los parámetros mínimos, identificado de la transacción ID_y, la firma de emisor $(SenderSingX_y)$ y los datos de la transacción $(Data_y)$.

4. Algunas transacciones pueden generar y almacenar archivos f_d en la capa 4, los cuales como parámetros mínimos lleva el identificador del archivo (ID_d), información de la transacción que lo genera ($ID(T_y)$) y los datos del archivo $Data_d$.

5. Una vez realizadas las transacciones se regresan los identificadores de las transacciones ID_y y de los archivos generados ID_d a la capa 1 para ser guardados.

Por último, en el cuarto escenario se requieren los índices de llaves, transacciones, contratos inteligentes y archivos, por lo que se utilizan todas las capas. En este caso, la capa 1 genera sus respectivas claves criptográficas para posteriormente generar transacciones en la capa 2. Éstas, a su vez, crean contratos inteligentes en la capa 3, donde se generan archivos que se almacenan en la capa 4. Esto creará una o más transacciones en la capa 2, y estas transacciones almacenan sus respectivos índices en la capa 1 (ver Figura 5.5).

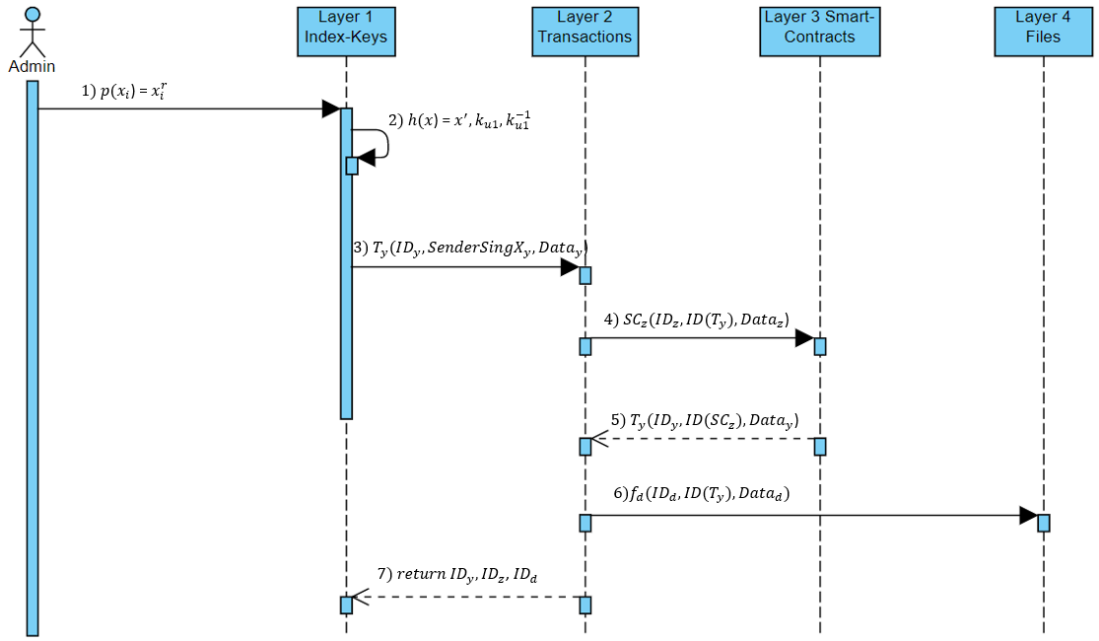

Figura 5.5: Caso 4: Llaves e índices, transacciones, smartcontracts y archivos

A continuación se describe el proceso descrito en la figura 5.5.

1. El administrador registra y configura el conjunto de las reglas, políticas, usuarios y procesos del servicio en un bloque (x_i^r).

2. Se generan las llaves criptográficas necesarias, como hashes ($x^{'}$), llaves públicas k_{u1} y privadas k_{u1}^{-1} necesarias para asegurar la integridad y seguridad de la información.

3. Los usuarios autorizados para realizar transacciones realizan transacciones firmadas con los datos complementarios necesarios, por lo tanto se genera la transacción T_y con los parámetros mínimos, identificado de la transacción ID_y, la firma de emisor ($SenderSingX_y$) y los datos de la transacción ($Data_y$).

4. Algunas transacciones pueden generar contratos inteligentes SC_z para automatizar futuras transacciones en los servicios, que como parámetros mínimos lleva el identificador del contrato inteligente (ID_z), información de la transacción que lo genera ($ID(T_y)$) y los datos del contrato inteligente $Data_z$.

5. Los contratos inteligentes generan una o varias transacciones al ejecutarse, por cada transacción T_y con los parámetros mínimos, identificado de la transacción ID_y, información del contrato inteligente que genera la transacción ($ID(SC_z)$) y los datos de la transacción ($Data_y$).

6. Algunas transacciones pueden generar y almacenar archivos f_d en la capa 4, los cuales como parámetros mínimos lleva el identificador del archivo (ID_d), información de la transacción que lo genera ($ID(T_y)$) y los datos del archivo $Data_d$.

7. Una vez realizadas las transacciones se regresan los identificadores de las transacciones (ID_y), contratos inteligentes (ID_z) y archivos guardados (ID_d) a la capa 1 para ser guardados.

Para mostrar cómo se puede utilizar el modelo, analizamos un caso de estudio que implementa el caso cuatro, es decir, utiliza 4 capas.

5.4. Caso de estudio: Un sistema de infracciones en un blockchain multicapa

Un posible caso de uso del blockchain multicapas es el sistema de infracciones. En los gobiernos el sistema de infracciones es administrado por un departamento especifico de policía. Dicho departamento tiene el control del sistema y si el

departamento es corrupto es posible modificar o eliminar infracciones. Es posible también modificar los ingresos finales por la recaudación del cobro de las infracciones, que en muchos casos son los fondos para obras especificas en las ciudades.

Al implementar el blockchain multicapas se descentraliza el control del sistema de infracciones en un departamento. Al guardar las infracciones en los bloques del blockchain ya no se pueden cambiar o eliminar. Cambiar los datos una vez registrados es posible pero al permitir que nodos externos al departamento participen se convierte en una tarea muy difícil teóricamente y en la practica resulta imposible.

Al utilizar contratos inteligentes es posible automatizar el estado de las infracciones pagadas e incluso se puede enlazar si se quisiera a un cobro automático al infractor.

Una posible forma de implementar los nodos en las 4 capas es como se muestra en la Figura 5.6. Donde se destaca la división regional de un país en estados y municipios. Diferentes países podrían tener más divisiones regionales donde el modelo se puede adaptar.

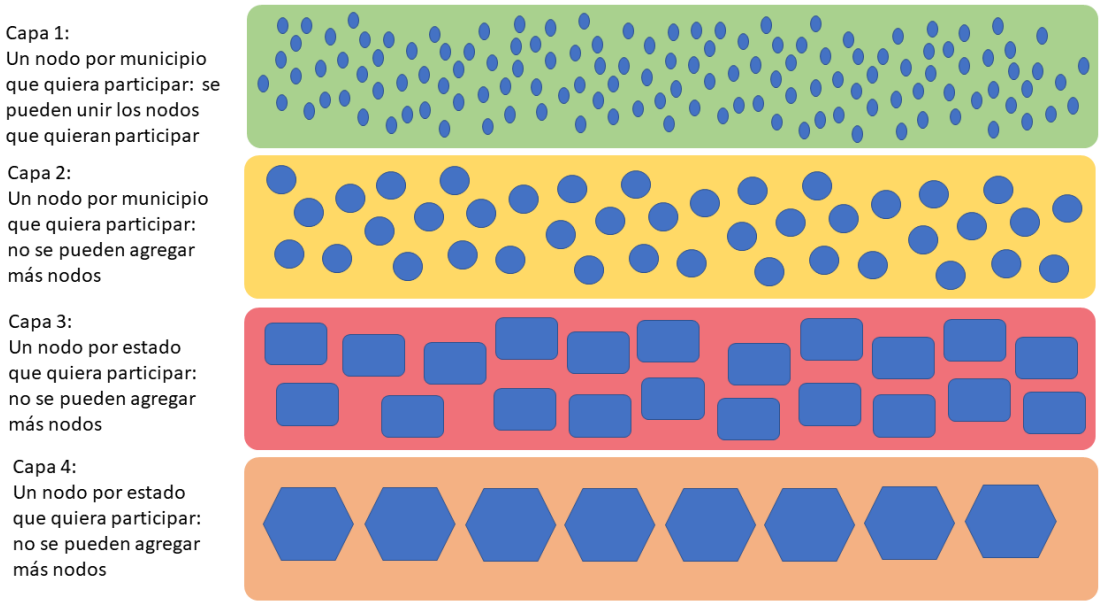

Figura 5.6: Propuesta de distribución de nodos en las 4 capas

Las capas 4, 3 y 2 son nodos preestablecidos e identificados, mientras que los nodos de la capa 1 pueden ser anónimos y no necesariamente identificados. El modelo puede funcionar con más de 3 nodos en cada capa, aunque a mayor cantidad de nodos mayor descentralización y seguridad de la información.

La capa 4 almacenaría los documentos de las infracciones como tickets, comprobantes de pago y facturas.

La capa 3 contendría los contratos inteligentes del pago de infracciones o de anulación de infracciones en caso de que procediera que la infracción es nula por cuestiones legales. También podría contener los códigos para enlazar a cobros automáticos cuando el usuario lo desee así.

La capa 2 tendría las transacciones del sistema que en este caso son las infracciones y los pagos de las mismas. Los bloques de la capa 2 ya requieren de nodos autorizados donde el objetivo es tener una alta velocidad de transferencias por segundo para manejar las infracciones y los pagos.

La capa 1 tendría los identificadores de las transacciones, identificadores de los contratos inteligentes, identificadores de los documentos y la información de los agentes autorizados para crear las infracciones (nombres, firmas electrónicas, etc.) Los bloques de la capa 1 son bloques que cualquier ciudadano puede consultar y verificar. Son los bloques que funcionan con la mayor transparencia al público y por tanto son fáciles de auditar. También nodos voluntarios se les permite aportar poder de cómputo para validar los bloques de esta capa y contribuyen a hacer más confiable la información al descentralizar a los encargados de validar la misma.

En la Figura 5.7 se puede ver como la capa 1 tiene relación con todas las capas al almacenar los índices, mientras que la las transacciones tienen relación con los contratos inteligentes y con los documentos.

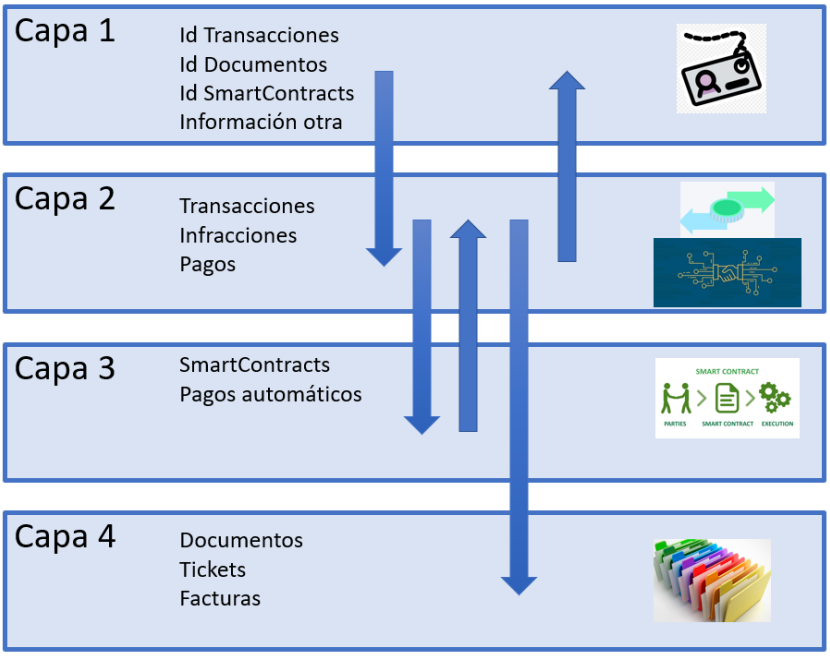

Figura 5.7: Interacciones entre capas

Este posible caso de estudio debe adaptarse dependiendo de las diferencias

que existan entre los distintos lugares y políticas que existan donde se deba implementar.

El flujo de la información seria como se muestra en la Figura 5.8, donde se configura en la capa uno las reglas, políticas, usuarios, etc. del servicio digital y posteriormente se pueden registrar los oficiales autorizados para emitir infracciones. Al ser registrados se generan sus respectivas llaves criptográficas y con eso quedan habilitados para emitir infracciones que serian generadas en la capa 2. En este caso también se puede generar un contrato inteligente en la capa 3 cada que se emita una infracción que permitirá automatizar el cumplimiento de una infracción. En este caso una infracción puede ser pagada o imputada y la ventaja del contrato inteligente es que automáticamente detecta el pago o la imputación emitida por la autoridad correspondiente. Posteriormente a que ocurra cualquiera de los dos eventos se guarda un archivo comprobante en la capa 4 y también se genera la transacción de cumplimiento que a su vez guarda los índices en la capa 1 donde quedara guardada la evidencia correspondiente.

En caso de que se quiera consultar determinada infracción en el pasado es muy simple rastrearla y por las propiedades de blockchain que protegida la información contra cualquier cambio futuro.

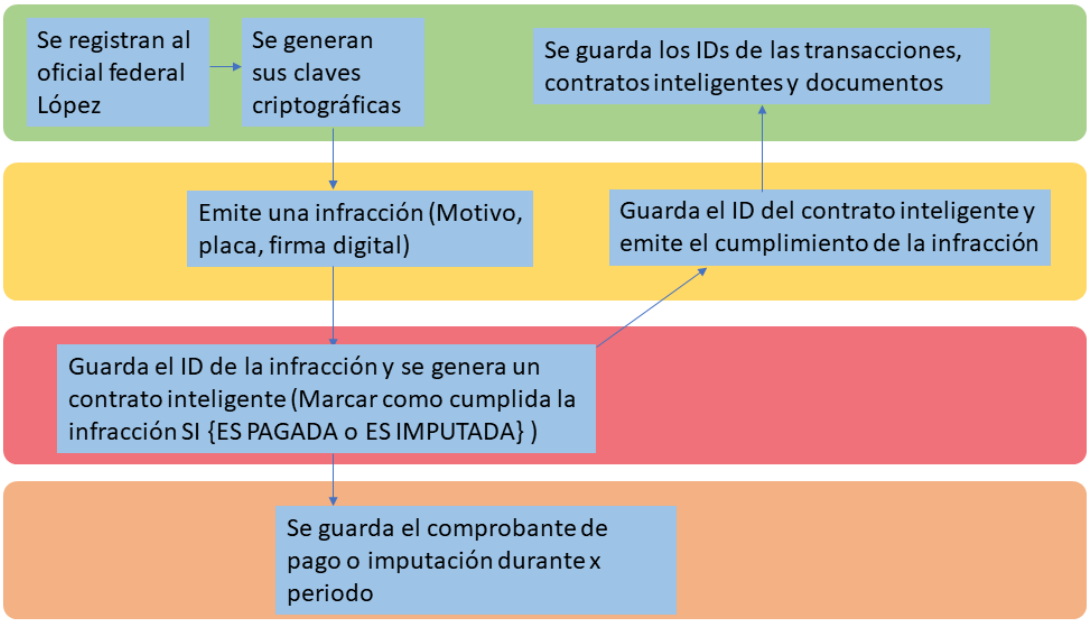

Figura 5.8: Ejemplo de flujo de la información utilizando las 4 capas del caso de estudio en un sistema de infracciones

5.4.1. Posible forma de implementar el caso de estudio

Para una posible implementación del modelo multicapa propuesto se podrían utilizar algunos de los módulos de distintos frameworks existentes para el frontend y para el backend. Para el caso de las capas en blockchain primero se deben configurar cada capa con su respectivo algoritmo de consenso para posteriormente ir añadiendo los nodos que van a participar. En las capas 2, 3 y 4 se deberá ir otorgando el acceso a nodos seleccionados de confianza, mientras que para la capa 1 se deberá publicar el software mínimo para descargar los archivos necesarios para convertirse en un nodo más de la red blockchain.

En la Figura 5.9 se esquematiza el frontend, backend y las 4 capas de la propuesta.

Figura 5.9: Esquema de lenguajes, algoritmos de consenso y frameworks para una posible implementación

En la parte del frontend se tienen marcos múltiples de html, js y css que se pueden utilizar para el diseño de las interfaces, de la misma manera frameworks como vue.js, react y angular entre otros.

En el backend existe AWS amplify, Parce, Back4App, Backendless y Firebase para almacenamiento en la nube.

La capa 1 puede utilizar el código abierto de BitcoinDevKit, configurarse y

adaptarse para posteriormente ir agregando los primeros nodos de la red y también debe configurarse la dificultad del algoritmo de consenso PoW para establecer la dificultad y la velocidad de generación de los bloques en relación a las necesidades del servicio digital que se quiera utilizar. Después debe liberarse a la red para que cualquier nodo que se quiera unir pueda hacerlo y participar en los consensos de la información lo que permitirá descentralizar el control de la red (solo de la capa 1). Para las capa 2, 3 y 4 como se había mencionado anteriormente se usaran nodos predefinidos que tendrán un determinado nivel de confianza para funcionar. La capa 2 encargada de las transacciones bajo el algoritmo de consenso PBFT con la que se garantiza un nivel de transacciones muy competitivo. La capa 3 deberá configurase el algortimo de consenso de DPoS y podra utilizar contratos inteligentes que se pueden implementar en Inpura, Solidity, Truffle y que el la blockchain de la criptomoneda de Etherum también se podrían interconectar como respaldo. La capa 4 usara IPFS para almacenar los archivos que se necesiten de manera segura y rápida.

5.4.2. Consideraciones al implementar el caso de estudio

La propuesta del modelo multicapa basado en blockchain podría ser utilizado en algunos de los servicios digitales de los gobiernos, con el objetivo de transparentar la información, combatir la corrupción y lograr que los ciudadanos se interesen y participen en los procesos de hacer confiable la información.

Hay algunas consideraciones a tomar en cuenta para aumentar la efectividad debido a que existen riesgos que pueden afectar al objetivo. Algunas de las consideraciones son las siguientes:

1. Los requerimientos de hardware y software de los equipos que quieran participar no tienen restricciones, equipos con características heterogéneas son validos para participar. Cualquier equipo que quiera participar puede unirse a la red. Equipos con mayor capacidad del calculo de hashes por segundo aportaran más a la red.

2. Los participantes externos no podrán hacer cambios a los datos contenidos en los bloques.

3. Los voluntarios que participen en la red pudieran o no obtener alguna recompensa. Al contribuir con poder de computo los gobiernos podrían fijar posturas para recompensar a los participantes o para no hacerlo, lo deseable es que no exista ninguna recompensa para no generar intereses ajenos al los propósitos de origen. Sin embargo, no se descarta que se puedan repartir recompensas de participación.

4. Múltiples personas y organizaciones no gubernamentales podrían estar interesadas en contribuir a pesar de que no exista una recompensa. Las universidades publicas son candidatas para realizar pruebas descentralizadas. Incluso pueden beneficiarse si se imparten cursos de blockchain, donde una practica viable es pedirle a los alumnos que participen voluntariamente en una red de este tipo para complementar su aprendizaje.

5. Los equipos que participen podrían ser restringidos a estar localizados dentro del mismo país. El uso de VPNs permitiría que equipos de otros países participaran en el sistema, en caso de querer restringirlo abría que buscar e implementar un mecanismo que lo impida.

6. Una posible restricción para que solo ciudadanos pertenecientes a un país participen, es administrar un control de acceso al blockchain mediante alguna identificación oficial tipo id vigente del país que lo implemente.

Podrían surgir más consideraciones, pero estas dependerían del caso de uso que se le quiera dar a la propuesta.

Capítulo 6

Conclusiones y trabajo futuro

En este capítulo se describe la discusión y conclusiones como primera parte. Posteriormente se describen los posibles trabajos futuros identificado en la investigación.

6.1. Conclusiones

La propuesta de un blockchain multicapas fue diseñada para ser utilizada por los gobiernos para tener un mayor control, transparencia y funcionalidad. Pero también puede ser utilizada en otros ámbitos.

La propuesta divide y organiza los diferentes tipos de información utilizada en 4 capas enfocándose en entregar un servicio digital eficiente con una alta tasa de transacciones. Esta separación admite el almacenamiento de archivos sin afectar las transacciones y los tiempos de ejecución de los contratos inteligentes y la capacidad de almacenamiento.

El modelo se formalizó en tres propiedades: (a) Velocidad de generación de los bloques, (b) Nivel de Descentralización y (c) Tamaño de Bloque, que cumple nuestro diseño de sistema en diferentes niveles dependiendo de la información manejada por cada capa.

Luego, se describió el funcionamiento del modelo utilizando cuatro casos comunes de uso. Esto demuestra la flexibilidad de nuestro modelo, ya que se puede adaptar a diferentes requisitos del sistema, lo que lo convierte en una generalización del problema presentado: ofrecer confianza y transparencia en los procesos complejos del sistema.

Los contribuyentes también serian ampliamente beneficiados al poder revisar la forma en que se utilizan sus contribuciones en el gobierno. También para los ciudadanos que quieran podrán participar en la validación de los bloques de la pri-

mera capa, contribuyendo a hacer más confiable la información al descentralizara y agregado más nodos imparciales a la red. Es importante mencionar que, dependiendo de los requisitos específicos del sistema, se debe realizar un análisis sobre el origen de dichos ciudadanos (nodos) y, si es necesario, establecer restricciones para los ciudadanos extranjeros.

Otra ventaja es que se deja el camino listo para agregar funcionalidades del IoT de las smartcities donde se podrán interconectar con los smart contracts que hace posible el blockchain validando y ejecutando los contratos automáticamente. También es posible adaptar su funcionalidad a una posible criptomoneda que pudiera emitir el banco central u otra que se quisiera utilizar para realizar transacciones y pagos.

6.2. Trabajo futuro

El modelo multicapas propone una solución novedosa a la integración de blockchain a múltiples servicios digitales, sin embargo, la propuesta aun puede tener posibles mejoras, es por ello que algunos trabajos futuros que pueden realizarse son:

- Probar con diferentes algoritmos de consenso las capas del blockchain multicapa para verificar si existe una posible mejora o si es posible realizar ajustes de optimización.
- Implementar y simular la carga de transacciones a la que podría verse sometida la red de blockchain multicapa para validar que la velocidad de transacciones sea la suficiente para proporcionar el o los servicios que se requieran.
- Diseñar la conexión de los servicios que se pueden automatizar con los contratos inteligentes.
- Crear un algoritmo de consenso a la medida para el blockchain multicapa.
- Probar el modelo con un entorno IoT, dados los bajos recursos computacionales que tienen la mayoría de estos dispositivos, probar las propiedades descritas ofrecerá una evidencia más sólida del funcionamiento del modelo.

Finalmente, para aumentar la confianza y la transparencia, se propone abrir el código de estos servicios digitales gubernamentales. De esta forma, todos pueden revisar y validar su implementación y funcionamiento. Poder descargar y comparar versiones utilizando hashes criptográficos para validar que el código que se

desplegó es el mismo que se está ejecutando. Agradecemos el apoyo a CONACyT con la beca número 701979.

Capítulo 7

Artículos, ponencias y estancia de investigación

7.1. Artículos

1. “La adopción del Blockchain por la nueva era digital como un sistema descentralizado para el uso y creación de nuevos servicios digitales”. Fernando Rebollar, Marco A. Ramos y Rosa M. Valdovinos. Publicado en la revista Komputer Sapiens. ISSN 2007-0691 Indexing: CONACYT, LatIndex.

2. “A multilayered model based on blockchain that forties the integrity and security of public information”. Fernando Rebollar, Rocío Aldeco-Pérez, Rosa M. Valdovinos y Marco A. Ramos. Aceptado para su futura publicación en Research in Computing Science. ISSN 1870-4069 (formerly ISSN 1665-9899) Indexing: DBLP, LatIndex, Periodica.

3. “Un blockchain multicapas para servicios digitales de gobiernos”. Fernando Rebollar, Rocío Aldeco-Pérez, Rosa M. Valdovinos y Marco A. Ramos. Aceptado para su futura publicación en la revista Integrare de la Facultad de ingeniería.

4. “Modeling a multilayered blockchain framework for digital services that governments can implement”. Fernando Rebollar, Rocío Aldeco-Pérez y Marco A. Ramos. Aceptado para su futura publicación en la revista Journal of Intelligent and Fuzzy Systems, a jcr-thompson journal.

7.2. Ponencias

1. "La adopción del Blockchain por la nueva era digital como un sistema descentralizado para el uso y creación de nuevos servicios digitales", 1er coloquio de investigación en ingeniería y 10mo Curso-Taller Temas actuales en ciencia del agua, octubre 2018.

2. "Blockchain ¿Una alternativa contra la corrupción en información?", Coloquio Doctoral LII Semana de Ingeniería, mayo 2019.

3. "La adopción del blockchain por la nueva era digital", 2do coloquio de investigación en ingeniería y 11vo Curso-Taller Temas actuales en ciencia del agua, noviembre 2019.

4. "A multilayered model based on blockchain that fortifies the integrity and security of public information", 19th Mexican International Conference on Artificial Intelligence (MICAI), octubre 2020.

5. "Un blockchain multicapas para servicios digitales de gobiernos", 3er coloquio de investigación en ingeniería y 12vo Curso-Taller Temas actuales en ciencia del agua, diciembre 2020.

6. "Un Modelo de Blockchain Multicapas para Aumentar la Confianza en los Servicios Digitales", "1er Seminario Virtual de Divulgación en Computación, junio 2021.

7. "Modeling a multilayered blockchain framework for digital services that governments can implement", 8th International Symposium on Language & Knowledge Engineering, noviembre 2021.

7.3. Estancia de investigación

1. Se realizó una estancia de investigación virtual en el IIMAS-UNAM (Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas de la Universidad Nacional Autónoma de México) con la Dra. Rocío Alejandra Aldeco Pérez. El periodo fue de 6 meses (enero - junio) 2021.

Referencias

[1] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Web: https://bitcoin.org/bitcoin.pdf consultado el 07-06-2019*, pages 1–9, 2008. v, 2, 16, 17, 19, 20, 22, 55

[2] Fernando Rebollar, Marco A. Ramos, and Rosa M. Valdovinos. La adopción del blockchain por la nueva era digital como un sistema descentralizado para el uso y creación de nuevos servicios digitales. *Komputersapiens*, 1(12):17–21, 2020. v, 26

[3] Xinle Yang, Xiaohu DavidChen, Sha Zhou, and Ryan Wang. The moac platform: Advancing performance with layered multi-blockchain architecture for enhanced smart contracting. *Web: https://www.moac.io/uploads/MOAC_White_Paper.pdf Accessed 6 Mar 2020*, 2018. v, 37, 38, 41

[4] Jeremy Lipschultz. *Free expression in the age of the Internet: Social and legal boundaries*. Routledge, 2018. 1

[5] Maximiliano Zito. La sustentabilidad de internet de las cosas. *Cuadernos del Centro de Estudios en Diseño y Comunicación. Ensayos*, pages 1–3, 2018. 1

[6] Maged N Kamel Boulos and Najeeb M Al-Shorbaji. On the internet of things, smart cities and the who healthy cities. *International journal of health geographics*, 1:5–7, 2014. DOI: 10.1186/1476-072X-13-10. 1

[7] Maged N Kamel Boulos, Bernd Resch, David N Crowley, John G Breslin, Gunho Sohn, Russ Burtner, William A Pike, Eduardo Jezierski, and Kuo-Yu Slayer Chuang. Crowdsourcing, citizen sensing and sensor web technologies for public and environmental health surveillance and crisis management: trends, ogc standards and application examples. *International journal of health geographics*, 1:67–68, 2011. DOI: 10.1186/1476-072X-10-67. 1

[8] Gerald C Kane. The dark side of the digital revolution. *MIT Sloan Management Review*, 57(3):1–8, 2016. 2

[9] Hemraj Saini, Yerra Shankar Rao, and TC Panda. Cyber-crimes and their impacts: A review. *International Journal of Engineering Research and Applications*, 2(2):202–209, 2012. DOI: 10.1.1.417.1369. 2

[10] José Ramón Gil-García, Judith Mariscal, and Fernando Ramírez. Gobierno electrónico en méxico. *Documento de Trabajo de la División de Administración Pública del CIDE,(214)*, 2010. 2

[11] Fernando Tricas Lamana. *El gobierno electrónico: servicios públicos y participación ciudadana*. Laboratorio de Alternativas, 2007. 2

[12] MyungSan Jun. Blockchain government-a next form of infrastructure for the twenty-first century. *Journal of Open Innovation: Technology, Market, and Complexity*, 4(1):7, 2018. 2

[13] Melanie Swan. *Blockchain: Blueprint for a new economy*. O'Reilly Media, 2015. 2, 3, 14, 20, 22

[14] David Schwartz, Noah Youngs, Arthur Britto, et al. The ripple protocol consensus algorithm. *Ripple Labs Inc White Paper*, 5:1–8, 2014. 3, 4, 17, 23

[15] Vitalik Buterin. A next-generation smart contract and decentralized application platform. *Web: https://github.com/ethereum/wiki/wiki/White-Paper consultado el 07-06-2019*, 2014. 3, 4, 17, 23

[16] G. Zyskind, O. Nathan, and A. Pentland. Decentralizing privacy: Using blockchain to protect personal data. In *Security and Privacy Workshops (SPW), 2015 IEEE*, pages 180–184. IEEE, 2015. DOI: 10.1109/SPW.2015.27. 3

[17] Yehuda Lindell. Highly-efficient universally-composable commitments based on the ddh assumption. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 446–466. Springer, 2011. DOI: 10.1007/978-3-642-20465-4_25. 3

[18] Xukai Zou, Huian Li, Yan Sui, Wei Peng, and Feng Li. Assurable, transparent, and mutual restraining e-voting involving multiple conflicting parties. In *INFOCOM, 2014 Proceedings IEEE*, pages 136–144. IEEE, 2014. DOI: 10.1109/INFOCOM.2014.6847933. 3

[19] Svein Ølnes. Beyond bitcoin enabling smart government using blockchain technology. In *International Conference on Electronic Government and the Information Systems Perspective*, pages 253–264. Springer, 2016. DOI: 10.1007/978-3-319-44421-5_20. 3

[20] Kibin Lee, Joshua I James, Tekachew Gobena Ejeta, and Hyoung Joong Kim. Electronic voting service using block-chain. *The Journal of Digital Forensics, Security and Law: JDFSL*, pages 123–125, 2016. DOI: 10.15394/jdfsl.2016.1383. 3

[21] J. Leon Zhao, Shaokun Fan, and Jiaqi Yan. Overview of business innovations and research opportunities in blockchain and introduction to the special issue. *Springer*, pages 3–7, 2016. DOI: 10.1186/s40854-016-0049-2. 3, 4

[22] Morgen E Peck. Blockchain world-do you need a blockchain? this chart will tell you if the technology can solve your problem. *IEEE Spectrum*, 54(10):38–60, 2017. 4

[23] Jesse Yli-Huumo, Deokyoon Ko, Sujin Choi, Sooyong Park, and Kari Smolander. Where is current research on blockchain technologya systematic review. *PloS one*, pages 10–12, 2016. DOI: 10.1371/journal.pone.0163477. 4, 25

[24] Konstantinos Christidis and Michael Devetsikiotis. Blockchains and smart contracts for the internet of things. *Ieee Access*, pages 2292–2303, 2016. DOI: 10.1109/ACCESS.2016.2566339. 4, 42

[25] Osi Momoh. Segwit (segregated witness). *Web: https://www.investopedia.com/terms/s/segwit-segregated-witness.asp consultado el 07-06-2019*, 2017. 4, 25

[26] Prableen Bajpai. What is bitcoin unlimited? *Web: https://www.investopedia.com/news/what-bitcoin-unlimited/ consultado el 07-06-2019*, 2016. 4, 25

[27] Real Academia Española and España Madrid. *Diccionario de la lengua española*. Real Academia Española, 1992. 7

[28] Andrew S Tanenbaum and Maarten Van Steen. *Distributed systems: principles and paradigms*. Prentice-Hall, 2007. 8, 9

[29] George F Coulouris, Jean Dollimore, and Tim Kindberg. *Distributed systems: concepts and design*. pearson education, 2005. 9

[30] Vitalik Buterin. The meaning of decentralization. *https://medium.com/@VitalikButerin/the-meaning-of-decentralization-a0c92b76a274 consultado el 07-06-2019*, 2017. 10, 11

[31] José Pastor Franco, Miguel Ángel Sarasa López, and José Luis Salazar Riaño. *Criptografía digital: fundamentos y aplicaciones*. Prensas Universitarias de Zaragoza, 2001. 11

[32] Jonathan Katz, Alfred J Menezes, Paul C Van Oorschot, and Scott A Vanstone. *Handbook of applied cryptography*. CRC press, 1996. 11

[33] Gustavus J Simmons. A survey of information authentication. *Proceedings of the IEEE*, 76(5):603–620, 1988. DOI: 10.1109/5.4445. 12, 13

[34] Prerna Mahajan and Abhishek Sachdeva. A study of encryption algorithms aes, des and rsa for security. *Global Journal of Computer Science and Technology*, pages 1–9, 2013. 12

[35] Thomas Beth and Dieter Gollmann. Algorithm engineering for public key algorithms. *IEEE Journal on selected areas in communications*, 7(4):458–466, 1989. 14

[36] Usman W Chohan. Cryptocurrencies: A brief thematic review. *Web: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3024330 consultado el 10-06-2019*, pages 1–9, 2017. DOI: 10.2139/ssrn.3024330. 14

[37] Jan Lansky. Possible state approaches to cryptocurrencies. *Journal of Systems Integration*, 9(1):19–31, 2018. DOI: 10.20470/jsi.v9i1.335. 15

[38] David L Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981. 15

[39] David Chaum. Blind signatures for untraceable payments. In *Advances in cryptology*, pages 199–203. Springer, 1983. DOI: 10.1007/978-1-4757-0602-4_18. 15

[40] David Chaum. Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, 1985. 15

[41] David Chaum, Amos Fiat, and Moni Naor. Untraceable electronic cash. In *Conference on the Theory and Application of Cryptography*, pages 319–327. Springer, 1988. DOI: 10.1007/0-387-34799-2_25. 15

[42] Laurie Law, Susan Sabett, and Jerry Solinas. How to make a mint: the cryptography of anonymous electronic cash. *Am. UL Rev.*, 46:1131–1133, 1996. 16

[43] Usman W Chohan. The double spending problem and cryptocurrencies. *Available at SSRN 3090174*, 2017. DOI: 10.2139/ssrn.3090174. 16

[44] Adam Back et al. Hashcash-a denial of service counter-measure, 2002. 16, 20

[45] Martin Haferkorn and Josué Manuel Quintana Diaz. Seasonality and interconnectivity within cryptocurrencies-an analysis on the basis of bitcoin, litecoin and namecoin. In *International Workshop on Enterprise Applications and Services in the Finance Industry*, pages 106–120. Springer, 2014. DOI: 10.1007/978-3-319-28151-3_8. 17

[46] Marco Alberto Javarone and Craig Steven Wright. From bitcoin to bitcoin cash: a network analysis. *arXiv preprint arXiv:1804.02350*, pages 1–5, 2018. DOI: 10.1145/3211933.3211947. 18

[47] Ian Grigg. Eos-an introduction. *Web: http://209.197.106.187/papers/EOS_Una_Introduccio?n-IanGrigg-Trad_MAM+AIH-20180717.pdf consultado el 10-06-2019*, pages 1–15, 2017. 18

[48] J Lawrence Carter and Mark N Wegman. Universal classes of hash functions. *Journal of computer and system sciences*, 18(2):143–154, 1979. DOI: 10.1016/0022-0000(79)90044-8. 20

[49] Nicolas T Courtois, Marek Grajek, and Rahul Naik. Optimizing sha256 in bitcoin mining. In *International Conference on Cryptography and Security Systems*, pages 131–144. Springer, 2014. 20

[50] C Retamal, J Roig, and J Tapia. La blockchain: fundamentos, aplicaciones y relación con otras tecnologías disruptivas. *Economía industrial*, 405:33–40, 2017. 21

[51] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang. An overview of blockchain technology: Architecture, consensus, and future trends. In *2017 IEEE International congress on big data (BigData congress)*, pages 557–564. IEEE, 2017. DOI: 10.1109/BigDataCongress.2017.85. 23

[52] J. Leon Zhao, Shaokun Fan, and Jiaqi Yan. Overview of business innovations and research opportunities in blockchain and introduction to the special issue. *Financial Innovation*, pages 28–29, 2016. DOI: 10.1186/s40854-016-0049-2. 24, 25

[53] Ali Dorri, Salil S Kanhere, Raja Jurdak, and Praveen Gauravaram. Blockchain for iot security and privacy: The case study of a smart home. In *2017 IEEE international conference on pervasive computing and communications workshops (PerCom workshops)*, pages 618–623. IEEE, 2017. DOI: 10.1109/PERCOMW.2017.7917634. 25

[54] Serguei Popov. The tangle. *Web: http://www.tangleblog.com/wp-content/uploads/2016/11/IOTA_Whitepaper.pdf consultado el 07-06-2019*, pages 1–25, 2017. 26, 27

[55] Wei Ren and Randal W Beard. Consensus algorithms for double-integrator dynamics. *Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications*, pages 77–104, 2008. 28

[56] Michael J Fischer. The consensus problem in unreliable distributed systems (a brief survey). In *International conference on fundamentals of computation theory*, pages 127–140. Springer, 1983. 28

[57] Arati Baliga. Understanding blockchain consensus models. *Persistent*, 4:1–14, 2017. 28

[58] Juan Garay and Aggelos Kiayias. Sok: A consensus taxonomy in the blockchain era. In *Cryptographers? Track at the RSA Conference*, pages 284–318. Springer, 2020. 28

[59] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 4(3):382–401, 1982. 29

[60] Cristina Pérez Solà and Jordi Herrera Joancomartí. Bitcoins y el problema de los generales bizantinos. *Web: https://rua.ua.es/dspace/bitstream/10045/40444/1/RECSI-2014_44.pdf consultado el 20-11-2019*, pages 241–246, 2014. 30

[61] Deepak K Tosh, Sachin Shetty, Xueping Liang, Charles A Kamhoua, Kevin A Kwiat, and Laurent Njilla. Security implications of blockchain cloud with analysis of block withholding attack. In *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pages 458–467. IEEE Press, 2017. 30

[62] Marko Vukolić. The quest for scalable blockchain fabric: Proof-of-work vs. bft replication. In *International workshop on open problems in network security*, pages 112–125. Springer, 2015. 31

[63] Daniel Larimer. Transactions as proof-of-stake. *Web: http://www.t.ly/GEd0z consultado el 20-11-2019*, pages 1–9, 2013. 31

[64] Daniel Larimer. Delegated proof-of-stake (dpos). *Bitshare whitepaper*, 2014. 32

[65] Ramakrishna Kotla, Lorenzo Alvisi, Mike Dahlin, Allen Clement, and Edmund Wong. Zyzzyva: speculative byzantine fault tolerance. *ACM SIGOPS Operating Systems Review*, 41(6):45–58, 2007. 32

[66] Miguel Castro and Barbara Liskov. Practical byzantine fault tolerance and proactive recovery. *ACM Transactions on Computer Systems (TOCS)*, 20(4):398–461, 2002. DOI: 10.1145/571637.571640. 33

[67] Cheng Li and Liang-Jie Zhang. A blockchain based new secure multi-layer network model for internet of things. In *2017 IEEE International Congress on Internet of Things (ICIOT)*, pages 33–41. IEEE, 2017. 35

[68] Shaimaa Badr, Ibrahim Gomaa, and Emad Abd-Elrahman. Multi-tier blockchain framework for iot-ehrs systems. *Procedia Computer Science*, 141:159–166, 2018. 36

[69] Edward Y Chang, Shih-Wei Liao, Chun-Ting Liu, Wei-Chen Lin, Pin-Wei Liao, Wei-Kang Fu, Chung-Huan Mei, and Emily J Chang. Deeplinq: Distributed multi-layer ledgers for privacy-preserving data sharing. In *2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*, pages 173–178. IEEE, 2018. 36

[70] Rishi Broto Chakraborty, Manjusha Pandey, and Siddharth Swarup Rautaray. Managing computation load on a blockchain ? based multi ? layered internet ? of ? things network. *Procedia Computer Science*, 132:469 – 476, 2018. International Conference on Computational Intelligence and Data Science. 38

[71] David López and Bilal Farooq. A multi-layered blockchain framework for smart mobility data-markets. *Transportation Research Part C: Emerging Technologies*, 111:588 – 615, 2020. 39

[72] Seyoung Huh, Sangrae Cho, and Soohyung Kim. Managing iot devices using blockchain platform. In *2017 19th international conference on advanced communication technology (ICACT)*, pages 464–467. IEEE, 2017. 41

[73] Elisa Bertino, Kim-Kwang Raymond Choo, Dimitrios Georgakopolous, and Surya Nepal. Internet of things (iot): Smart and secure service delivery. *ACM Trans. Internet Technol.*, 16(4), December 2016. 41

[74] Elisa García-Morales. Luces y sombras sobre el impacto del blockchain en la gestión de documentos. *Anuario ThinkEPI*, 12:345–351, 2018. 42

[75] Mathis Steichen, Beltran Fiz, Robert Norvill, Wazen Shbair, and Radu State. Blockchain-based, decentralized access control for ipfs. In *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pages 1499–1506. IEEE, 2018. 45