ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

"DISEÑO Y APLICACIÓN DE UN SISTEMA DE SEGURIDAD DESCENTRALIZADO MEDIANTE LA TECNOLOGÍA BLOCKCHAIN PARA APLICACIONES WEB"

CARINA ARACELI CARRILLO VILLALVA

Trabajo de Titulación modalidad: Proyectos de Investigación y Desarrollo, presentado ante el Instituto de Posgrado y Educación Continua de la ESPOCH, como requisito parcial para la obtención del grado de:

MAGISTER EN SEGURIDAD TELEMATICA

RIOBAMBA - ECUADOR

Agosto - 2021

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

CERTIFICACIÓN:

EL TRIBUNAL DEL TRABAJO DE TITULACIÓN CERTIFICA QUE:

El Trabajo de Titulación Modalidad Proyectos de Investigación y Desarrollo, denominado: "**Diseño y aplicación de un sistema de seguridad descentralizado mediante la tecnología Blockchain para aplicaciones web**", de responsabilidad de la señorita Carina Araceli Carrillo Villalva, ha sido prolijamente revisado y se autoriza su presentación.

Tribunal:

Ing. Luis Hidalgo Almeida; Ph.D.	LUIS EDUARDO HIDALGO ALMEIDA Firmado digitalmente por LUIS EDUARDO HIDALGO ALMEIDA Nombre de reconocimiento (DN): c=EC, o=BANCO CENTRAL DEL ECUADOR, ou=ENTIDAD DE CERTIFICACION DE INFORMACION-ECIBCE, l=QUITO, serialNumber=0000445780, cn=LUIS EDUARDO HIDALGO ALMEIDA Fecha: 2021.08.19 15:59:25 -05'00'
PRESIDENTE DEL TRIBUNAL	**FIRMA**
Ing. Renny Geovanny Montalvo Armijos; Mag.	Firmado electrónicamente por: **RENNY GEOVANNY MONTALVO ARMIJOS**
DIRECTOR	**FIRMA**
Ing. Wilian Xavier Sánchez Labré; Mag.	Firmado electrónicamente por: **WILIAN XAVIER SANCHEZ LABRE**
MIEMBRO DEL TRIBUNAL	**FIRMA**
Ing. Cristian Fabricio Viteri Silva; Mag.	Firmado electrónicamente por: **CRISTIAN FABRICIO VITERI SILVA**
MIEMBRO DEL TRIBUNAL	**FIRMA**

Riobamba, agosto 2021

DERECHOS INTELECTUALES

Yo, Carina Araceli Carrillo Villalva, declaro que soy responsable de las ideas, doctrinas y resultados expuestos en este **Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo**, y que el patrimonio intelectual del mismo pertenece a la Escuela Superior Politécnica de Chimborazo.

CARINA ARACELI CARRILLO VILLALVA
No. Cédula: 060361900-8

DECLARACIÓN DE AUTENTICIDAD

Yo, Carina Araceli Carrillo Villalva, declaro que el presente **Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo**, es de mi autoría y que los resultados del mismo son auténticos y originales. Los textos constantes en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor, asumo la responsabilidad legal y académica de los contenidos de este Trabajo de Titulación de Maestría.

Riobamba, agosto 2021

CARINA ARACELI CARRILLO VILLALVA
No. Cédula: 060361900-8

DEDICATORIA

El presente trabajo va dedicado a Dios, por ser el motor de mi vida y permitirme alcanzar una nueva meta a nivel académico.

A mis padres por su apoyo incondicional y por ser un ejemplo de constancia y dedicación; gracias por sus enseñanzas y su amor.

A mis hermanos, Lourdes y Cristian; como a mi hermana del alma, Verito por estar siempre presentes, apoyándome en los momentos más difíciles, para no dejarme desmayar.

A todas las personas que han colaborado de una u otra manera para la culminación exitosa de este trabajo.

Carina Carillo V.

AGRADECIMIENTO

Agradezco a Dios por cada día de vida, y por permitirme alcanzar nuevas metas a nivel personal y académico; y a mi familia por su apoyo incondicional en todo momento.

Mi agradecimiento a la Escuela Superior Politécnica de Chimborazo por el desarrollo favorable de la maestría, a todos los docentes facilitadores, por compartir sus conocimientos y experiencias, en especial al tutor y miembros del tribunal por ser un gran aporte para la realización de la investigación.

Y, no puedo dejar de agradecer a mis compañeros, en especial a mi querida amiga Verónica Paucar, con quienes compartimos gratos momentos durante el tiempo de estudios.

Carina Carrillo V.

TABLA DE CONTENIDO

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

ÍNDICE DE GRÁFICOS

ÍNDICE DE ANEXOS

RESUMEN

El presente trabajo tiene como objetivo la elaboración de un sistema de seguridad descentralizado mediante la tecnología BLOCKCHAIN para aplicaciones web, para implementar una capa adicional de seguridad al sistema de autenticación. Se diseñaron e implementaron 2 prototipos en el lenguaje Nodejs, el prototipo 1 contó con un sistema de autenticación tradicional (usuario, contraseña), mientras que el prototipo 2 implementó una capa adicional basada en blockchain y contratos inteligentes para registrar a los usuarios y llevar un control de sus accesos, apoyándose en la firma digital que permite identificar unívocamente a cada usuario en el servidor web, siendo los componentes principales del sistema propuesto la aplicación web, el usuario y el contrato inteligente. La autenticación en el sistema web propuesto se realiza en dos etapas: la primera al proporcionar el nombre de usuario y la contraseña a la aplicación / sitio web deseado, y la segunda es descifrando la Clave de un solo Uso (OTP) que se genera y se envía a través de un contrato inteligente que solicitó la aplicación web para autorizar al usuario, donde los bloques de registros se almacenan en la blockchain. Para la demostración de la hipótesis se evaluaron los 2 prototipos mediante las herramientas Vooki y Owasp ZAP y en base al análisis de vulnerabilidades presentadas se concluyó que el sistema propuesto mejora el nivel de seguridad en una aplicación web al reducir en un 83.33% las vulnerabilidades presentadas, cabe recalcar que el sistema está exento de los ataques comunes como XSS, inyección SQL porque no hay un servidor central donde se almacena la información de inicio de sesión. Finalmente, el sistema propuesto hereda las características de la tecnología blockchain que son la descentralización, la inmutabilidad, el uso de hashes criptográficos y las firmas digitales; parámetros que brindan un valor agredado a la seguridad de un sistema/aplicación web.

PALABRAS CLAVES: <AUTENTICACIÓN>, <BLOCKCHAIN>, <AUTENTICACIÓN DE DOS FACTORES>, <SMART CONTRACT>, <SISTEMA WEB DESCENTRALIZADO>, <SEGURIDAD WEB>.

ABSTRACT

The present work aims to develop a decentralized security system using BLOCKCHAIN technology for web applications, to implement an additional layer of security to the authentication system. Two prototypes were designed and implemented in the Nodejs language, prototype 1 had a traditional authentication system (user, password), while prototype 2 implemented an additional layer based on blockchain and smart contracts to register users and keep track of their accesses, relying on the digital firm that allows to univocally identificate each user in the webserver, being the main components of the proposed system the web application, the user, and the smart contract. Authentication in the proposed web system is performed in two stages: the first by providing the username and password to the desired application/website, and the second is by decrypting the One Time Key (OTP) that is generated and sent through a smart contract requested by the web application to authorize the user, where the blocks of records are stored in the blockchain. For the demonstration of the hypothesis the 2 prototypes were evaluated using Vooki and Owasp ZAP tools and based on the analysis of vulnerabilities presented it was concluded that the proposed system improves the level of security in a web application by reducing by 83.33% the vulnerabilities presented, it should be emphasized that the system is exempt from common attacks such as XSS, SQL injection because there is no central server where the login information is stored. Finally, the proposed system inherits the characteristics of blockchain technology which are decentralization, immutability, the use of cryptographic hashes, and digital signatures; parameters that provide added value to the security of a web system/application.

Keywords: <AUTHENTICATION>, <BLOCKCHAIN>, <TWO-FACTOR AUTHENTICATION>, <SMART CONTRACT>, <DECENTRALIZED WEB SYSTEM>, <WEB SECURITY>.

CAPÍTULO I

1. INTRODUCCIÓN

A medida que la tecnología avanza y cambia constantemente, el activo más valioso, la información, está expuesta a nuevas vulnerabilidades, como el caso del ransomware Wannacry en 2017, que se estima infectó a más de 200.000 máquinas en unos 150 países" (Savita Mohurle, 2017); razón por la cual la ciberseguridad se ha convertido en un área de gran importancia. Otro tipo de ataque es el spoofing que consiste en el robo de identidad. Por lo cual es necesario, desarrollar métodos de autenticación para mejorar la seguridad de nuestras aplicaciones basados en tecnologías emergentes como es el caso de blockchain, una de las tecnologías más disruptiva en estos tiempos. Por otro lado, la tecnología blockchain se está situando como una de las tecnologías más disruptivas de los últimos años que permite la transferencia de información con un alto nivel de seguridad, para lo cual no se requiere de un nodo central que identifique y certifique la información, sino que está distribuida en múltiples nodos independientes entre sí que la registran y la validan.

El presente trabajo se enfoca en el desarrollo de un nuevo método de autenticación basado en la tecnología blockchain con la finalidad de mejorar la seguridad en las aplicaciones web, para lo cual se desarrollará un sistema descentralizado de seguridad basado en cadenas de bloques para aplicaciones web, para lo cual es necesario entender el funcionamiento de la tecnología blockchain, las plataformas de desarrollo y los Smart Contracts, aspectos indispensables para este trabajo investigativo.

La propuesta se apoya en blockchain y contratos inteligentes para registrar a los usuarios y llevar un control de sus accesos, apoyándose en la firma digital que permite identificar unívocamente a cada usuario en el servidor web. Gracias al uso de blockchain, cada acceso se registra de forma inmutable. Para demostrar el funcionamiento se ha desarrollado un sistema de acceso a recursos en el que los usuarios pueden añadir recursos, de forma que el resto de usuarios puedan solicitar el acceso a ellos.

Este trabajo investigativo inicia con la investigación y el análisis de la tecnología blockchain, sus características, ventajas, desventajas, funcionamiento, algoritmos, los métodos criptográficos empleados y su campo de aplicación, así como las herramientas necesarias para el desarrollo de aplicaciones descentralizadas; como la suite Truffle, y el lenguaje Solidity.
Para el desarrollo del prototipo se ha realizado una planificación basados en los recursos, y características que tendrá la red blockchain sobre la que se realizará el despliegue de los contratos inteligentes. En base a la planificación se realizó la implementación de la Dapp, para lo cual se diseñó la API que permita que las operaciones del servidor se comuniquen con la blockchain. En lo que respecta al frontend se tomaron en cuenta los componentes planificados que permitan la comunicación adecuada con las funciones establecidas en los contratos inteligentes.
Finalmente se procedió a realizar las pruebas para determinar la validez de la hipótesis planteada, así como el proceso realizado para su implementación.

1.1. Problema de Investigación.

1.1.1. Planteamiento del problema

En una sociedad sumergida en el mundo del internet y la tecnología, las personas se han convertido en consumidores digitales de múltiples sitios web. Por lo general en estos sitios web se busca identificar a los usuarios para ofrecer servicios pagados, basándose en inicios de sesión tradicionales que por lo general utilizan un identificador, como un nombre de usuario o dirección de correo electrónico, que actuará como la identidad del usuario en ese servicio y una contraseña que se utiliza para afirmar que el usuario es quien dice ser, método que ha sido vulnerado y por ende es considerado inseguro. Las contraseñas usadas son demasiado fáciles de adivinar (por ejemplo, contraseña123, un nombre, fecha de nacimiento, etc.), o se pueden obtener mediante un ataque de fuerza bruta. Además, se puede agregar la seguridad inadecuada alrededor del sistema que contiene el repositorio de contraseñas o directorio de usuarios, al ser sistemas centralizados.

Aunque, para evitar que una sola cuenta se vea comprometida o contraseña perdida que pone en peligro el resto de las identidades en línea de un usuario, se recomienda utilizar diferentes contraseñas para todos y cada uno de los servicios en los que el usuario está registrado. Esto es un pequeño problema si el usuario está registrado en múltiples servicios.

Los administradores de contraseñas populares como LastPass o OnePass, que permiten al usuario generar y administrar contraseñas, también pueden verse comprometidos. Esta aplicación de

terceros almacena su contraseña en un servidor centralizado que puede permitir fácilmente que un ataque comprometa su inicio de sesión.

Otro enfoque para resolver el problema de tener que recordar varias contraseñas es el inicio de sesión único (SSO). Los sistemas SSO adoptan un enfoque diferente los servicios anteriormente mencionados, en lugar de construir sobre el nombre de usuario y la contraseña proporcionan al usuario la capacidad de iniciar sesión en un servicio mediante la autenticación dirigida hacia un tercero llamado proveedor de identidad. En la práctica, esto significa que cualquier servicio puede agregar un botón a su página de autenticación permitiendo a sus usuarios iniciar sesión utilizando, por ejemplo, su cuenta de Google. El botón llevará al usuario a la página de inicio de sesión de Google donde el usuario puede iniciar sesión con sus datos de Google, cuando es autenticado, el usuario volverá al servicio original donde el usuario inició sesión como la identidad vinculada con esa cuenta de Google. El enfoque SSO tiene muchos beneficios, como la facilidad de uso, la simplicidad y la capacidad para que servicios más pequeños implementen soluciones estandarizadas de autenticación segura basadas en grandes actores como Google. Sin embargo, un gran inconveniente con las soluciones SSO que existen hoy en día es que el usuario final debe confiar en el proveedor de identidad para proteger su privacidad. Otro inconveniente es la disponibilidad del sistema, si el proveedor de identidad no continua ofreciendo sus servicios no habrá forma de acceder a las cuentas en varios servicios donde ese proveedor de identidad se ha utilizado como autenticación.

En la actualidad uno de los enfoques más comunes que se utiliza para aumentar la seguridad de la autenticación es el uso de dos factores en lugar de usar solo uno.

Para mitigar los problemas de los mecanismos de autenticación usados en los sistemas web es necesario el uso de nuevas tecnologías que permitan asegurar los procesos de autenticación, en vista de las nuevas formas de ataque que se desarrollan.

Por lo cual, se propone el uso de la tecnología blockchain, que se sustenta en fuertes principios criptográficos y que cuenta con la implementación de una clave privada y una clave pública para garantizar la transferencia de información de forma confiable necesarios para el diseño del método de autenticación propuesto que permita asegurar un sitio web.

1.2. Formulación del problema

¿El resultado de las pruebas del sistema de seguridad descentralizado mediante la tecnología BLOCKCHAIN permitirá establecer mecanismos de seguridad en aplicaciones web?

1.2.1. Sistematización del problema

- ¿Es posible que la tecnología blockchain pueda contribuir en la creación de un método seguro de autenticación para sistemas web?
- ¿Puede blockchain mejorar la seguridad en los sistemas web?

1.3. Justificación de la investigación

1.3.1. Justificación Teórica

En la actualidad dependemos en gran medida de la tecnología, llegando a tener toda nuestra información en un sistema o aplicación que depende del internet, pero a medida que la tecnología avanza aparecen nuevos métodos que vulneran la seguridad de los mismos, por este motivo se desarrollara un método de autenticación basado en la tecnología blockchain que permita asegurar el proceso de acceso a un sistema web.

Existe una serie de alternativas diferentes a los sistemas de solo contraseña, incluida la biometría y el uso de autenticación multifactor. Ahora los sistemas de autenticación se están construyendo con tecnología blockchain, que está diseñada sobre la premisa de que un sistema descentralizado es mejor que uno centralizado. Existen múltiples copias de datos compartidos, visibles públicamente, en la cadena de bloques y todas las transacciones se almacenan como bloques y se reconcilian entre los miembros con una frecuencia establecida. Esto evita que un ataque a un servidor individual en la cadena de bloques comprometa los datos en su conjunto.

Con la finalidad de alcanzar los objetivos propuestos, se emplearán diversas técnicas de investigación como la revisión documental de guías y manuales de referencia sobre la tecnología blockchain y la plataforma ethereum, como la programación en solidity para el desarrollo de los contratos inteligentes.

El centro de la autenticación de bloque sería un ID que puede ser verificado por cualquier tercero y puede mostrar la información necesaria. El secreto de esta verificación es el ECDSA (algoritmo de firma digital de curva elíptica). Cuando se agrega un ID a blockchain, un servicio de emisión de identificación enlaza una clave pública de forma predeterminada y luego transfiere la propiedad de la clave privada al usuario. Esto permite al usuario, y sólo al usuario, firmar una firma que se puede verificar contra la clave pública almacenada en la cadena de bloques. Esta identificación de un usuario será la fuente descentralizada de autenticación. (D. Johnson, 2001)

1.3.2. Justificación Metodológica

Para el desarrollo del presente estudio se requiere una investigación proveniente de diferentes fuentes de información tales como tesis de grado, sitios web, libros y documentos electrónicos, siendo referenciadas con el uso de las normas APA 6ta Edición.

Mientras que para la fase de desarrollo de la aplicación web se usará una metodología incremental, que permite construir la aplicación de manera progresiva, en cada etapa se agrega una nueva funcionalidad la cual permite un desarrollo flexible. (Maida, 2015)

Los resultados obtenidos, luego de aplicar un proceso investigativo científico que permita demostrar la validez de la hipótesis, podrán ser una referencia para futuros estudios.

1.3.3. Justificación Práctica

Para la demostración práctica del proyecto se desarrolló dos prototipos, el primero desarrolla un sistema de autenticación tradicional para un sitio web y en el segundo prototipo se aplica el método de autenticación basado en la tecnología blockchain y el contrato inteligente, que ofrece las propiedades de descentralización, inmutabilidad y transparencia para mejorar la seguridad del sitio. La aplicación también utiliza la autenticación de dos factores para mejorar aún más la seguridad de sus usuarios.

Para la medición del nivel de cumplimiento se utilizan herramientas de control de desarrollo seguro. El análisis de herramientas y los casos de estudio se realizan en un ambiente de pruebas implementado en un servidor local y un un servidor real publicado en la web.

1.4. Objetivos de la investigación

1.4.1. Objetivo General

Elaborar un sistema de seguridad descentralizado mediante la tecnología BLOCKCHAIN para aplicaciones web.

1.4.2. Objetivos Específicos

- Realizar un estado de arte sobre la tecnología blockchain para sistemas de seguridad en aplicaciones web.
- Diseñar un método de autenticación que mejore la seguridad de los sistemas web mediante Blockchain.
- Elaborar una cadena descentralizada blockchain para la gestión de información del sistema de acceso para la interacción con los contratos inteligentes, por medio de la librería Web3 de JavaScript.
- Implementar y evaluar el diseño propuesto mediante pruebas la integridad y confidencialidad de la información de acceso en aplicaciones web.

1.5. Hipótesis

Un sistema de autenticación basado en la tecnología blockchain si mejora la seguridad en un sistema web.

CAPÍTULO II

2. MARCO TEÓRICO

2.1. Antecedentes del problema

Se realiza una revisión de los artículos científicos e investigaciones referentes a la seguridad web en lo que respecta a métodos de autenticación de un usuario en un servidor web.

En la actualidad existen varios trabajos de investigación que toman como punto clave la tecnología blockchain para el proceso de acceso y autenticación de los usuarios, en el caso de los proyectos desplegados en redes públicas se da mayor prioridad a la privacidad.

Uno de los trabajos de investigación como base a nuestro estudio es el "Diseño e implementación de un sistema y autenticación y acceso a área restringidas mediante el uso de una red blockchain y aplicaciones descentralizas para el manejo de reportes" que estudia e implementa un sistema de control de accesos a áreas restringidas por medio de una red blockchain local permitiendo el registro y el control de perfiles, así como la generación de reportes del control de acceso realizada por los usuarios, al conectar a la blockchain varios nodos, como una raspberry, una aplicación descentralizada, se puede observar que la información se mantiene en cualquiera de los nodos y está disponible todo el tiempo de vida de la blockchain local. (Ávila & Toquica, 2018)

En el artículo científico titulado "Town crier: An authenticated data feed for smart contracts", analiza el uso de contratos inteligentes y los implementan en un sistema autenticado para el suministro de datos llamado Town Crier (TC), el mismo que actúa como puente entre los Smart contracts y los sitios web que se conecten. Combina un frontend basado en blockchain con un backend basado en hardware. La implementación está desarrollada para sitios web que habiliten https y asegura la confidencialidad; además permite la solicitud de datos privados con parámetros cifrados. En el artículo se describen los principios de diseño y la arquitectura de TC y analiza la implementación que utiliza las Extensiones de protección de software (SGX) introducida por Intel para proporcionar datos al sistema de contrato inteligente Ethereum. (Fan Zhang, 2016)

El proyecto de investigación denominado "Aseguramiento de Dispositivos IoT con Blockchain e Infraestructura de Clave Pública" analiza y determina los beneficios de las nuevas tecnologías disruptivas como Blockchain al unirlas a otras implementaciones tradicionales de protección, como son las infraestructuras de clave pública para el manejo de claves criptográficas, distribuyendo y expandiendo sus posibilidades, lo que permite asegurar los dispositivos IoT que se conecten a la red, garantizando su acceso remoto confiable mediante redes como Internet. Para el desarrollo del proyecto se establece una metodología de auditoría basada en riesgos que les permite evaluar los prototipos presentados para medir las mejoras obtenidas en términos de seguridad, para lo cual se evaluó un dispositivo IoT configurado por defecto y otro mediante la implementación y despliegue de una infraestructura de clave pública distribuida bajo la tecnología Blockchain. (Balmaseda-Aranda, 2018)

Por otro lado, el trabajo de investigación "Gestión de identidades descentralizadas con Blockchain", plantea un novedoso método para la gestión de identidades de forma descentralizada, para lo cual se recurre a la tecnología blockchain y sus componentes principales. Toda la investigación se plasma en un prototipo desarrollado con la tecnología Hyperledger Indy. (Fernández, 2018)

En el artículo científico titulado "A TOTP-Based Two Factor Authentication Scheme for Hyperledger Fabric Blockchain", proponen un nuevo método para evitar la vulnerabilidad de autenticación de Token de reclamo, proporcionado por PrivateBlockChain, usando tecnología Hyperledger Fabric v1.0 y el método de autenticación JWT. El sistema implementado genera tokens OTP y códigos de autenticación de usuario que generan una contraseña adicional basada en el tiempo de permanencia en los servidores de autenticación existentes. La arquitectura propuesta no puede espiar un token de acceso porque no envía un token de autenticación desde el servidor de autenticación al cliente. Además, dado que no se puede obtener la información del código de autenticación del usuario, existe la ventaja de que se puede garantizar la seguridad del usuario incluso si se calcula el TOTP. Sin embargo, desde el punto de vista del rendimiento, existe el inconveniente de que el tiempo de ejecución de la autenticación aumenta y la velocidad de la transacción disminuye. (Park, Hwang, & Kim, 2018)

En el artículo científico titulado "Ensuring data integrity using blockchain technology" la tecnología Blockchain puede asegurar la integridad de los archivos almacenados en la base de datos, gracias a las propiedades que ofrece la tecnología como son autenticación y auditoría. La cantidad de posibles amenazas a la integridad de los datos se puede disminuir garantizando las dos propiedades restantes de los datos: confidencialidad y disponibilidad. Según el modelo Clark-Wilson las características de un sistema de integridad seguro se compone de los siguientes factores:

transacciones bien definidas, separación de funciones, autenticación, auditoría, principio de mínimo privilegio, control objetivo y control sobre la transferencia de privilegios. (Zikratov, Kuzmin, Akimenko, Niculichev, & Yalansky, 2017)

El trabajo de investigación titulada "Blockchain Based Access Control Services" se presenta un nuevo diseño, presenta un nuevo enfoque de diseño para los servicios de control de acceso que aprovechan los contratos inteligentes proporcionados por la tecnología blockchain. La idea clave de nuestra propuesta es codificar las políticas de control de acceso como contratos inteligentes ejecutables en una cadena de bloques. inteligentes completamente distribuidas. En el proyecto basado en blockchain, también los administradores de atributos requeridos para la evaluación de las políticas de control de acceso son administrados por blockchain. Además analiza la viabilidad de la propuesta para la implementación de referencia de trabajo utilizando políticas XACML y contratos inteligentes escritos de Solidity implementados en Ethereum. (Maesa, Mori, & Ricci, 2018)

El siguiente trabajo de investigación desarrollado por (Pagán & María, 2019), bajo el título "Sistema de autenticación robusto" emplea la tecnología Blockchain para desarrollar el sistema ChainAuth basado en la identidad digital (Blockchain based ID as a Service – BIDaaS). Aprovecha los beneficios de la cadena de bloques para realizar una correcta validación de las identidades garantizando la seguridad y la inmutabilidad de los datos. ChainAuth se apoya en dos grandes propiedades la integridad y la autenticidad. La comunicación con el sistema ChainAuth es sobre una capa TLS (Transport Layer Security) en su versión 1.3, sin embargo, toda la comunicación entre el partner y la aplicación de usuario dependerá del partner, aunque la aplicación soporta HTTPS y TLS.

Por último el trabajo de investigación realizado por (Sanchez Mora & Jerez Vargas, 2019) titulado "Sistema descentralizado para la verificación y autenticación de certificados académicos utilizando la tecnología Blockchain", describe un sistema para el desarrollo de las aplicaciones web y móvil utilizando la tecnología blockchain hyperledger, con el fin de disminuir los índices de falsificación de los diplomas académicos, con un caso de aplicación en la Universidad Distrital Francisco José de Caldas. Para la interoperabilidad del sistema web y móvil configuran y desarrollan API REST con hyperledger composer, con el propósito de generar los métodos de respuesta get, post, put y delete además, con la utilización del formato JSON se procesa con las tecnologías ReactJs y React native para el desarrollo de los respectivos sistemas. Para los procesos de análisis, diseño e implementación se selecciona como metodología desarrollo de software SCRUM, estableciendo para cada sprint sus actividades correspondientes, con el fin de realizar entregas parciales para cumplir con cada uno de los objetivos propuestos. El sistema principalmente está enfocado para

cualquier persona que dese conocer la autenticidad de los diplomas, además de llevar un control automatizado por parte la universidad sin necesidad de depender de un servidor que este activo durante la ejecución del mismo.

2.2. Bases teóricas

2.2.1. Blockchain

El concepto de cadena de bloques o Blockchain, es un término que se da a conocer a partir del año 2008, con el seudónimo de Satoshi Nakamoto y que actualmente se conoce por ser la plataforma base de la criptomoneda Bitcoin. Pero realmente blockchain es una tecnología disruptiva que consiste en una base de datos distribuida (libro mayor) con fuertes capacidades criptográficas, en donde cada transacción se verifica por consenso por la mayoría de los participantes en el sistema. (Bashir, 2017)

La primera implementación de blockchain, fue pensada para el intercambio de dinero digital entre dos partes, evitando hacer partícipe a instituciones fiscales, para ello, se incluyen sistemas y mecanismos para que ambas partes confíen en el intercambio. (Nakamoto, 2008)

Hoy en día, la mayor parte de los servicios que se usan, tienen una estructura centralizada, manteniendo todos los datos en una única ubicación conocida como servidor. En este caso, la cadena de bloques tiene una estructura distribuida, marcando la diferencia respecto a las otras convencionales: la información esta replicada en todos los nodos que componen la blockchain, los datos una vez registrados y validados no se pueden modificar; de hecho, las transacciones se conservan en un histórico desde el nacimiento de la tecnología. Por todo esto, sólo hace falta un único nodo para conservar los datos de la red, evitando la pérdida de la información.

Blockchain, ofrece una base de datos distribuida, basada en una secuencia creciente de bloques; bloques que al ser públicos conforman un sistema abierto que aumenta el nivel de confianza de quienes lo usan. A pesar de que el sistema de cadena de bloques es abierto, a la vez es semi anónimo toda vez que los usuarios se identifican con claves pseudónimos, no con sus datos reales. Con lo anterior, se decanta la idea de cómo el blockchain, proporciona solidez, seguridad y transparencia a los contenidos de información y de datos, lo que constituye una gran ventaja en un mundo completamente globalizado e informático. (Dolader, 2017)

Actualmente se enmarca dentro de las Tecnologías de Registro Distribuidas (Distributed Ledger Technologies, DLT), siendo predecesora de las mismas.

A más del campo financiero la tecnología blockchain tiene un amplio campo de aplicación como: cadenas de suministros, sistemas de votaciones, entre otras.

2.2.2. Características de Blockchain

A la vista de lo anteriormente expuesto, las características de la red blockchain más relevantes son:

- **Transparencia:** Partiendo de la base de que todos los usuarios de las redes blockchain tienen acceso al libro registro, ello implica que todos tienen la información sobre las transacciones que se efectúan por el grupo. Es más, en determinadas redes, los usuarios que no forman parte de la red también pueden consultar el contenido de la cadena de bloques. Así ocurre, por ejemplo, en las redes Bitcoin o Ethereum. A esto se añade, además, que se trata de protocolos informáticos de código abierto, por lo que el acceso al diseño de la programación es también libre.

  Esta transparencia, sin embargo, no significa que podamos conocer al autor de las transacciones en todo caso. En algunos tipos de redes los usuarios no necesitan identificarse de forma personal para acceder y operar en la correspondiente red blockchain.

  Las transacciones son visibles, pero vinculadas a un código. Esta característica ha ocasionado que se hayan vinculado algunas de estas redes a actividades ilícitas por el carácter anónimo en la actuación que permiten en ciertos casos.

- **Irrevocabilidad:** Una vez que la información se incorpora a una red blockchain, en general (salvo ciertas excepciones), no es posible eliminarla de allí. En otras palabras, no hay marcha atrás. La información es poseída por todos los usuarios, por lo que es imposible eliminarla de la red. Los datos incorporados a la cadena de bloques se distribuyen a todos y cada uno de los nodos que intervienen en ella.

- **Inmutabilidad:** Como consecuencia del encadenamiento sucesivo de los bloques basado en la criptografía (los hash), el contenido de la cadena de bloques es inmutable.

  Si un nodo decide cambiar el contenido de la cadena de bloques alterando una transacción ya realizada e incluida en un bloque, provocará que el contenido de su versión del libro registro varíe, un cambio que será fácilmente identificable por el resto de los nodos. Por lo

tanto, a la hora de someter a aprobación una nueva transacción, estos no aceptarán su versión del registro, puesto que el contenido será distinto.

Estas tres propiedades son atribuibles de forma general a las redes blockchain. Sin embargo, existen otros parámetros que los desarrolladores tienen en cuenta y deciden a la hora de configurar estas redes, dependiendo de la función a la que cada red esté destinada, y que permiten matizar lo que acabamos de exponer. En particular, en función de las decisiones sobre algunos de estos parámetros, las redes blockchain pueden ser públicas o privadas:

- **Redes públicas:** no exigen a los usuarios el cumplimiento de ningún requisito para poder unirse a ellas (requisitos de identificación) y no existe ninguna jerarquía entre los nodos, por lo que cualquier nodo puede convertirse en nodo validador si lo desea. El contenido de la cadena de bloques es transparente y visible para todos los usuarios (en algunos casos, incluso para aquellos que no son usuarios de la red). Puesto que estas redes no exigen permiso o invitación alguna para poder acceder y participar, reciben el calificativo de permissionless. Para evitar el fraude, los nodos validadores, además de realizar las operaciones de validación, deben resolver un conjunto de problemas criptográficos antes de poder incorporar un nuevo bloque a la cadena de bloques (este tipo de sistema recibe el nombre de proof-of-work, como el ideado en su día por Nick Szabo). Puesto que para realizar estas tareas los nodos validadores deben poner a disposición de la red su poder computacional, con los gastos energéticos y a nivel de infraestructura que ello comporta, reciben una compensación por realizar esta tarea. En gran parte de las redes públicas, este incentivo se traduce en la recepción de una pequeña comisión al primer nodo validador que consigue resolver el problema criptográfico. Estos nodos validadores son también conocidos como «mineros » y su acción como «minar» o «minería». (Bashir, 2017)

- **Redes privadas:** un grupo limitado de actores conserva el poder de acceder, comprobar y añadir transacciones al libro registro.

En cuanto a todas las características mencionadas anteriormente, reunidas ya ayudan a describir la cadena de bloques como un sistema seguro para compartir cualquier tipo de información.

Además, lo que también hace que blockchain sea un sistema seguro es la forma en que funciona una transacción. Siempre que se intente realizar una transacción, esta debe ser validada por los otros

nodos de la cadena de bloques. Además, la transacción necesita tanto la clave pública como la privada de cada componente para ser ejecutada.

2.2.3. Propiedades de Blockchain

Entre las principales propiedades de blockchain se pueden considerar:

- ***Replicación P2P (peer-to-peer)***

  Estas redes son un conjunto de ordenadores conectados entre sí llamados nodos en los que se permite el intercambio directo de información, sin necesidad de que esa información pase antes por un servidor central. (Miethereum, 2020)

- ***Inmutabilidad***

  Es la propiedad más deseada para mantener la atomicidad de las transacciones de blockchain. Una vez que se registra una transacción, no se puede modificar. Si las transacciones se transmiten a la red, casi todos tienen una copia. Con el tiempo, cuando se agregan más y más bloques a la cadena de bloques, la inmutabilidad aumenta y, después de un cierto tiempo, se vuelve completamente inmutable. Que alguien altere los datos de tantos bloques en una serie no es prácticamente factible porque están protegidos criptográficamente. Por lo tanto, cualquier transacción que se registre permanece para siempre en el sistema.

- ***Resistente a la falsificación***

  Una solución descentralizada donde las transacciones son públicas es propensa a diferentes tipos de ataques. Los intentos de falsificación son los más obvios de todos, especialmente cuando realiza transacciones de valor. Se pueden utilizar hashes criptográficos y firmas digitales para garantizar que el sistema sea resistente a la falsificación. Ya aprendimos que es computacionalmente inviable falsificar la firma de otra persona. Si realiza una transacción y firma un hash, nadie podrá modificar la transacción más tarde y decir que firmó una transacción diferente. Además, no puede reclamar más tarde que nunca realizó la transacción, porque fue usted quien la firmó.

- ***Estado coherente del libro mayor***

  Las propiedades que acabamos de comentar aseguran que el libro mayor sea consistente en todo momento, hasta cierto punto. Imagine una situación en la que algunos nodos desean deliberadamente que una transacción no se realice y sea rechazada. O, si de alguna manera algunos nodos no están sincronizados con el libro mayor y, por lo tanto, no están al tanto de algunas transacciones que tuvieron lugar mientras estaban fuera de línea, entonces para ellos una transacción puede parecer fraudulenta. Entonces, cómo garantizar el consenso entre los participantes es algo que debe manejarse con mucho cuidado. Recuerda el problema de los generales bizantinos. El tipo de consenso adecuado para una situación dada juega el papel más importante para garantizar la estabilidad de una solución descentralizada.

- ***Flexible***

  La red debe ser lo suficientemente resistente como para soportar fallas temporales en los nodos, la falta de disponibilidad de algunos nodos informáticos a veces, la latencia de la red y la caída de paquetes, etc.

- ***Auditable***

  Blockchain es una cadena de bloques, en donde los nodos están unidos entre sí a través de hashes. Dado que los bloques de transacciones están vinculados hasta el bloque génesis, es posible la auditabilidad y se debe asegurar de que no se rompa a ningún costo. Además, si uno quiere verificar si una transacción tuvo lugar en el pasado, entonces dicha verificación debería ser más rápida.

- ***Seguridad***

  En cuanto a todas las características mencionadas anteriormente, en forma conjunta ayudan a describir la cadena de bloques como un sistema seguro para compartir cualquier tipo de información.

  Además, lo que también hace que blockchain sea un sistema seguro es la forma en que funciona una transacción. Siempre que se intente realizar una transacción, esta debe ser validada por los otros nodos de la cadena de bloques. Además, la transacción necesita tanto la clave pública como la privada de cada componente para ser ejecutada.

La cadena de bloques está formada por muchos nodos que comparten información. Para cerrar toda la cadena de bloques, sería necesario cerrar todos los nodos, por lo que, mientras un nodo todavía esté en ejecución, la cadena de bloques seguirá funcionando perfectamente.

2.2.4. Arquitectura de la Blockchain

La arquitectura blockchain consta de algunos conceptos fundamentales como descentralización, firma digital, minería e integridad de datos.

- ***Descentralización:*** en lugar de que una autoridad central domine a otras en el ecosistema, blockchain, se distribuye explícitamente el control entre todos los pares de la cadena de transacciones.

- ***Firma digital:*** Blockchain permite el intercambio de valores transaccionales mediante claves públicas mediante el mecanismo de un signo digital único, es decir, un código para descifrar conocido por todos en la red y claves privadas conocidas solo por el propietario para crear la propiedad.

  Las firmas digitales se basa en una combinación de criptografía asimétrica y algoritmos hash para proporcionar una forma para que los destinatarios del mensaje afirmen la autenticidad e integridad del mensaje.

  Un mensaje firmado se crea y se envía de la siguiente manera:

  1. Se calcula un resumen de hash del mensaje con un algoritmo de hash conocido por el receptor.

  2. Se cifra el resumen de hash, pero en lugar de utilizar la clave pública del receptor, se utiliza la clave privada del remitente.

  3. Se envía este resumen de hash cifrado junto con el mensaje no cifrado.

  Cuando el receptor desee verificar el mensaje recibido, seguirá los siguientes pasos:

  1. Descifra la firma recibida con la clave pública del supuesto remitente.

2. Calcula un resumen de hash del mensaje recibido. Si la firma descifrada y el hash son iguales, entonces el receptor ha verificado la autenticidad y la integridad del mensaje. Si por otro lado el descifrado la firma no es igual al hash calculado, entonces el remitente no es quien afirma ser o el mensaje ha sido modificado o dañado durante la transmisión.

- ***Minería:*** en un sistema distribuido, cada usuario extrae y profundiza en los datos que luego se evalúan de acuerdo con las reglas criptográficas y también se usan a los mineros para la confirmación y verificación de las transacciones.

- ***Integridad de los datos:*** Algoritmos complejos y acuerdos entre los usuarios aseguran que los datos de transacción, una vez acordados, no pueden ser interceptados.

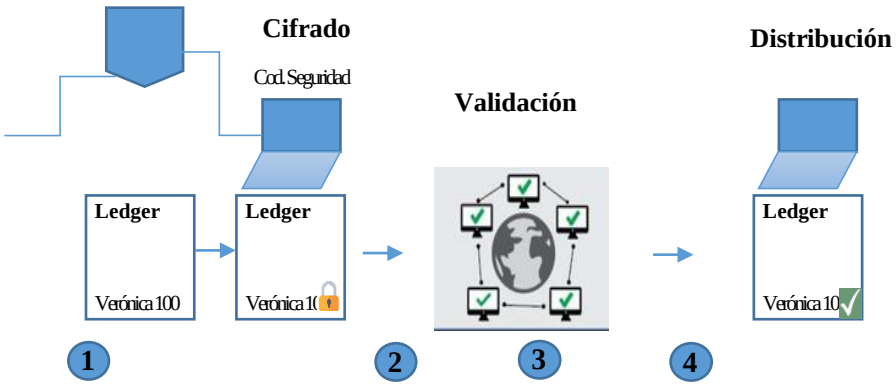


Figura 1-2. Arquitectura de la Blockchain

Realizado por: Carrillo Carina, 2020

En la figura [1-2] se muestra varias transacciones que se pueden resumir en 4 fases:

1. La transacción es añadida al ledger (libro mayor), cifrada mediante un código de seguridad.

El código de la transacción es enviado a lo largo de la red donde la autenticidad del código es confirmado sin comprometer la información privada y eliminando la necesidad de la confirmación de la transacción por parte de una entidad central.

2. Una vez que la transacción es confirmada y validada por varias partes, la transacción se graba en cada uno los ledgers pertenecientes a cada uno de los nodos de la blockchain; lo que permite tener un registro permanente e inmutable de la transacción.

3. La información de la transacción es grabada en el ledger público, y la transacción es finalizada.

2.3. Funcionamiento de la Blockchain

Blockchain es un controlador digital que almacena la transacción públicamente después de verificar la transacción por nodos. La estructura básica de la tecnología blockchain se muestra en la figura [2-2]. Cada transacción es validada por los nodos y las transacciones están aseguradas por la función hash de criptografía:

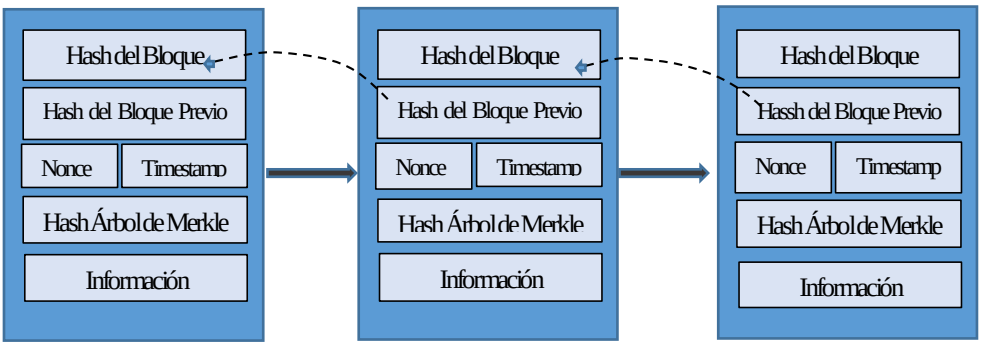


Figura 2-2. Estructura de un bloque en Blockchain

Fuente: (Dolader, 2017)

Una transacción está vinculada por su valor de hash de transacción anterior. Una vez que la transacción se agrega a la cadena de bloques, nadie puede modificarla o alterarla, pero esa transacción se puede ver abiertamente, lo que brinda transparencia al sistema. Blockchain utiliza algunas de las pruebas de concepto y prueba de trabajo, así como la prueba de concepto de participación para validar la transacción.

Blockchain es una base de datos distribuida donde se ejecutan transacciones. Así que lo primero que debemos pensar es qué es una transacción.

Una transacción es un hecho que ocurre entre dos partes. En el caso de las monedas digitales es sencillo entenderlo, y es el ejemplo que siempre se pone: “Verónica envía 100 dólares a Cristian”. Esta información es enviada a todos los nodos que componen la red de Blockchain. Todos tienen

una copia exacta de los datos existentes en la base de datos distribuida. Cuando esta transacción llega, es ejecutada en todos los nodos de la red, por lo que para poder falsear la transacción se deberían falsear todos los nodos de la red.

Las transacciones se caracterizan por ser:

- Atómicas, por lo que no puede ocurrir solo parte de la misma, o se ejecuta completa o no se ejecuta.
- Independientes unas de otras, por lo que no pueden interactuar entre ellas o interferir unas en otras.
- Inspeccionables, que quiere decir que es posible visualizar lo que ha ocurrido en las mismas.
- Inmortales: Lo que pasó pasó y no se puede modificar o eliminar.

Una vez que tenemos claro que Blockchain se basa en las transacciones, hay otro concepto que es igual de importante, y es el hashing. Una función hash se trata de una función que aplica un algoritmo sobre una entrada de datos y que genera un resultado con un tamaño predeterminado.
Los hash son funciones de una sola dirección: siempre van a devolver el mismo resultado dando la misma entrada, pero no se puede recrear la entrada a través del hash. Esto también significa que si algo cambia en la entrada el hash no será el mismo que el que originalmente surgió y es una forma rápida de saber si el contenido ha cambiado.

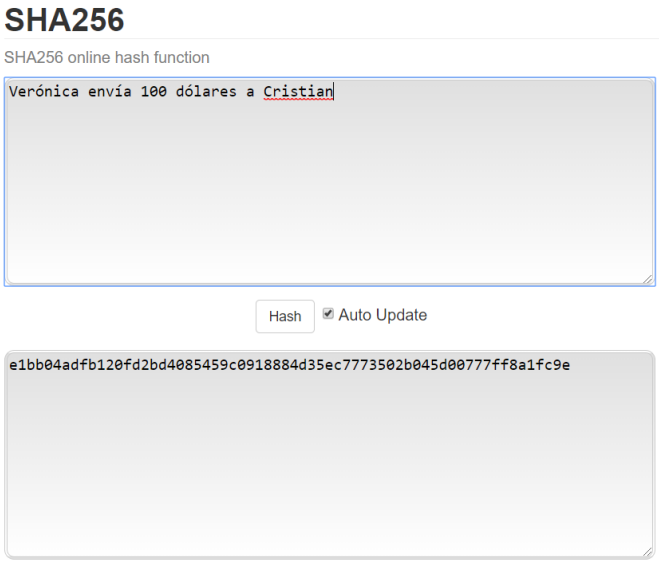


Figura 3-2. Código Sha-256 de una cadena

Realizado por: Carrillo Carina, 2019

Dependiendo de la implementación de Blockchain que estemos utilizando, el algoritmo puede ser diferente. Por ejemplo, en Bitcoin se utiliza SHA 256, pero en Ethereum se usa KECCAK-512. En el ejemplo de transacción "Verónica envía 100 dólares a Cristian" obtendríamos un hash como el que se muestra en la figura.

Lo bueno de este sistema, es que no importa si el texto de entrada es más largo o más corto, ya que el tamaño del hash siempre será el mismo, lo cual hace que sea más cómodo trabajar con ellos que directamente con el texto de entrada.

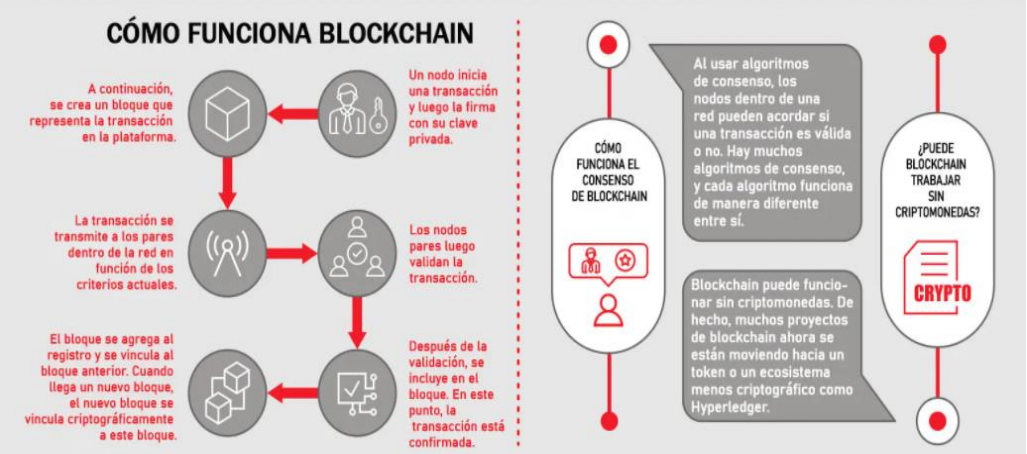


Figura 4-2. Esquema de funcionamiento de una blockchain

Fuente: (Rodriguez, 2019)

2.4. Los bloques de Blockchain.

Un bloque es un conjunto de transacciones y los hashes asocian un bloque con el siguiente. Por lo tanto, según lo explicado anteriormente, si algún dato del bloque es modificado el hash ya no será el mismo que el hash asociado a estos datos y, por lo tanto, el bloque será inválido. También contiene un nounce o nonce, el cual no es más que un número aleatorio, que no es posible predecir, y que trabaja en combinación con el hash como un elemento de control más para evitar la manipulación del bloque. Cuando un nuevo bloque ocurre, es necesario ejecutar el algoritmo que nos da el hash así como el nounce. A esto se le llama minar el bloque. Cada bloque contiene el hash del bloque anterior.

Los bloques de la blockchain, además de las transacciones realizadas, almacenan una serie de datos esenciales para el funcionamiento de la misma. Estos datos varían entre las diferentes cadenas de bloques, de forma general podemos destacar los siguientes campos (ver Figura 5-2).

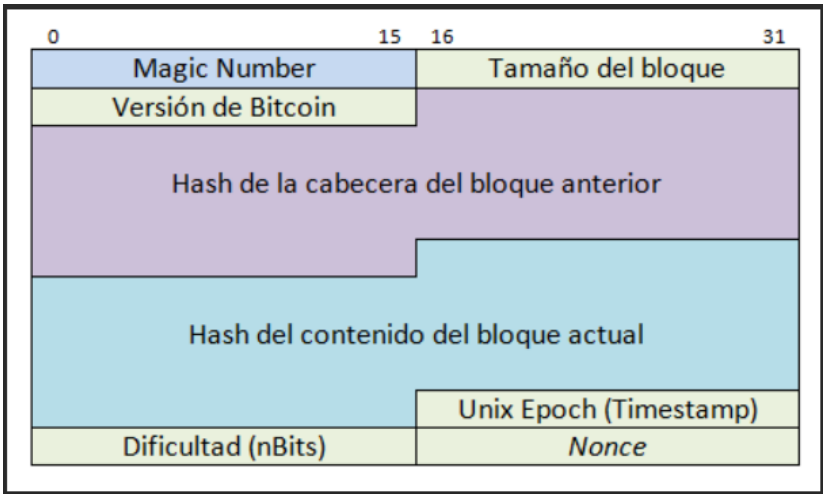


Figura 5-2. Ejemplo de bloque simplificado

Fuente: (Bitcoin, 2020)

Heigh: Indica la altura a la que se encuentra el bloque dentro de la cadena, es decir, indica el número de bloque en la blockchain.

Timestamp: Marca temporal que indica fecha y hora en la que se ha creado el bloque.

Datos o transacciones: Operaciones realizadas en la blockchain. Además del envío de algún token de un usuario a otro, pueden representar varias cosas, como pueden ser operaciones con los smart contracts. Cada transacción está firmada por el usuario que la realiza. Estas se almacenan en forma de árbol de Merkle para facilitar la verificación de las mismas.

Hash: El resultado de aplicar una función hash al bloque. Este, dependiendo del tipo de blockchain, tendrá determinado requisito, que puede ser que comience con un número determinado de ceros. Gracias a esto se regula la dificultad de minado.

Hash previo: Es el hash del bloque inmediatamente anterior en la cadena de bloques.

Nonce: En las redes de blockchain que utilizan como algoritmo de consenso la prueba de trabajo, el nonce es el número que se ha de calcular para que, al aplicar la función hash sobre el bloque, se cumpla el requisito del hash establecido por la dificultad de minado.

Como cada bloque almacena una copia del hash anterior, si un bloque es modificado, el siguiente ya no será válido, pues el hash almacenado no coincidirá. En caso de que un atacante quiera realizar un cambio, tendría que modificar el bloque que le interesa y todos los siguientes, con el consiguiente

coste computacional que esto conlleva. Este coste computacional, asociado al algoritmo de consenso de prueba de trabajo, viene de calcular el previamente mencionado nonce, lo que comúnmente se conoce como minado. Para que un bloque sea incluido en la blockchain, primero ha de minarse, es decir, hallar el valor del nonce que hace que el hash de dicho bloque comience con el número determinado de ceros establecido por la dificultad de minado actual del bloque.

- ***Transacciones.-*** Como se ha mencionado previamente, las transacciones dentro del bloque están firmadas por el usuario y se almacenan en forma de árbol de Merkle para agilizar la verificación de las mismas. Esto hace que si un usuario malintencionado quiere alterar una, aunque consiga modificar toda la cadena de bloques para que estos sean válidos, minando cada uno de ellos, este cambio será detectado en las propias transacciones por no verificarse la firma.

- ***Distribución.*** **–** Además de todas las medidas de seguridad expuestas, tenemos que la cadena de bloques se encuentra replicada en cada nodo de la red. Cualquier cambio se puede comprobar en el resto de copias de la cadena, tomando como cadena válida la que coincida en más nodos como correcta.

2.5. Implementaciones de Blockchain.

Existen varias implementaciones de blockchain de código abierto, entre ellas:

- ✓ Ethereum
- ✓ Quorum
- ✓ Corda
- ✓ Stellar
- ✓ Ripple
- ✓ Hyperledger
- ✓ Alastria

Esta red puede ser pública o privada. Si es pública, cualquier persona con acceso a Internet puede tener acceso a la cadena de bloques. En este tipo de entornos, la forma de que estos se financien es pagando por cada transacción, ejecución y almacenamiento que se realice sobre el sistema, por lo que suele ser caro. Lo bueno es que están distribuidos globalmente, por lo que no existe un único

punto de ataque. Estas implementaciones de Blockchain son mantenidas por la comunidad, por lo que siempre existe el riesgo de que en cualquier momento se descontinúen.

En el caso de los privados, el propietario puede controlar el gasto y quiénes son los participantes en la base de datos distribuida. Además se pueden controlar dónde están los nodos, que estén dentro de la región legal que se necesita.

Existe un tercer tipo conocido como consorcios. Estas redes tienen como participantes organizaciones conocidas. Un ejemplo de ello es Alastria, donde hay empresas conocidas como Repsol, Banco Santander, BME Innova, ICADE, Everis, Endesa, Banca Sabadell, BBVA, entre otros.

2.6. Criptografía.

Para comprender cómo funciona blockchain y conocer su seguridad, en gran parte basada en la inmutabilidad de sus datos y la integridad de los mismos, es necesaria una introducción de los conceptos criptográficos en los que se apoya esta tecnología.

- ***Funciones hash***

  Una función hash es una operación criptográfica que genera una cadena arbitraria a partir de un conjunto de datos de entrada (ver Figura 6-2). Normalmente, esta cadena es de longitud fija, establecida por el algoritmo que se emplee. En el caso de blockchain, se utiliza SHA-256, donde el tamaño de la cadena resultante, también denominada hash, es de 256 bits. Algunas características de SHA-256, como función hash criptográfica, son: su determinismo, es decir, que el hash de dos entradas exactamente iguales, siempre será el mismo; su dificultad para ser revertida, pues hasta ahora no se ha encontrado método para obtener los datos iniciales a partir de su hash; y su resistencia a las colisiones, pues es muy poco probable que dos cadenas diferentes tomen el mismo hash utilizando este algoritmo. Gracias a estas características, el hash se emplea dentro de la cadena de bloques para asegurar la integridad de los datos, pues el hash de cada bloque es almacenado en el siguiente de la cadena, de forma que, si un bloque es modificado, el hash del mismo no coincidirá con el que contiene el siguiente.

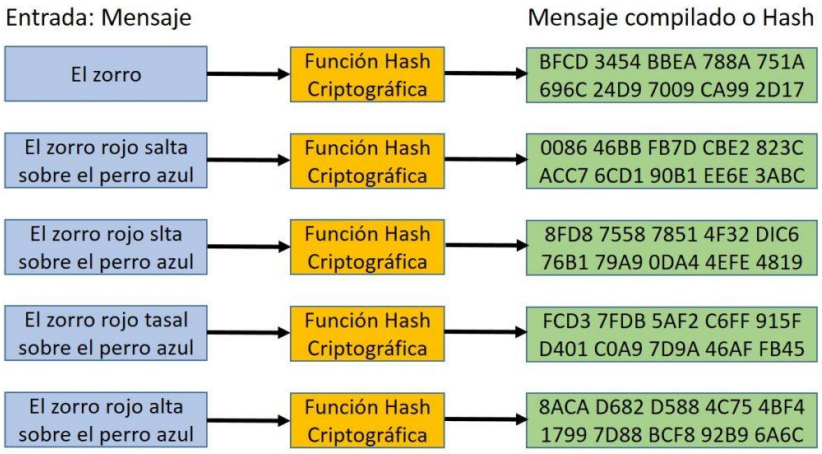


Figura 6-2. Esquema sobre el funcionamiento de un hash criptográfico

Fuente: (Miethereum, 2020)

Las propiedades de una función hash segura son:

- ✓ ***Computacionalmente eficiente:*** Las computadoras deben ser capaces de llevar a cabo la labor matemática necesaria para crear un hash en un período de tiempo muy corto.
- ✓ ***Determinista:*** Esto implica que el mismo mensaje (entrada) debe producir siempre el mismo digest (salida) cada vez que sea utilizado o consultado.
- ✓ ***Resistente a preimagen:*** Significa que la salida no debe revelar ningún dato en absoluto sobre la entrada. Es por eso que un hash debería tener siempre la misma longitud en el digest, independientemente del tamaño del mensaje. Tampoco debe darse ninguna pista sobre el contenido de tal mensaje, por lo que, al más mínimo cambio, el hash resultante debe ser por completo distinto.
- ✓ ***Resistente a colisión:*** Dos (o más) entradas diferentes no deberían producir la misma salida (digest).

- ***Infraestructura de clave pública y firma digital***

Cada usuario de la blockchain dispone de una clave pública, la cual es compartida con el resto de nodos de la red, e identifica al mismo, y una clave privada, a la que solo él tiene

acceso. Ésta se utiliza para firmar cada transacciones y verificar que se han realizado por dicho usuario. Es decir, cada vez que se añade una transacción a un bloque, esta es firmada por el usuario con su clave privada (ver Figura 7-2). El resto de usuarios de la red pueden verificar esta transacción empleando la correspondiente clave pública, de modo que si se cambia cualquier dato de la misma, la firma se modifica y se detecta el fraude.

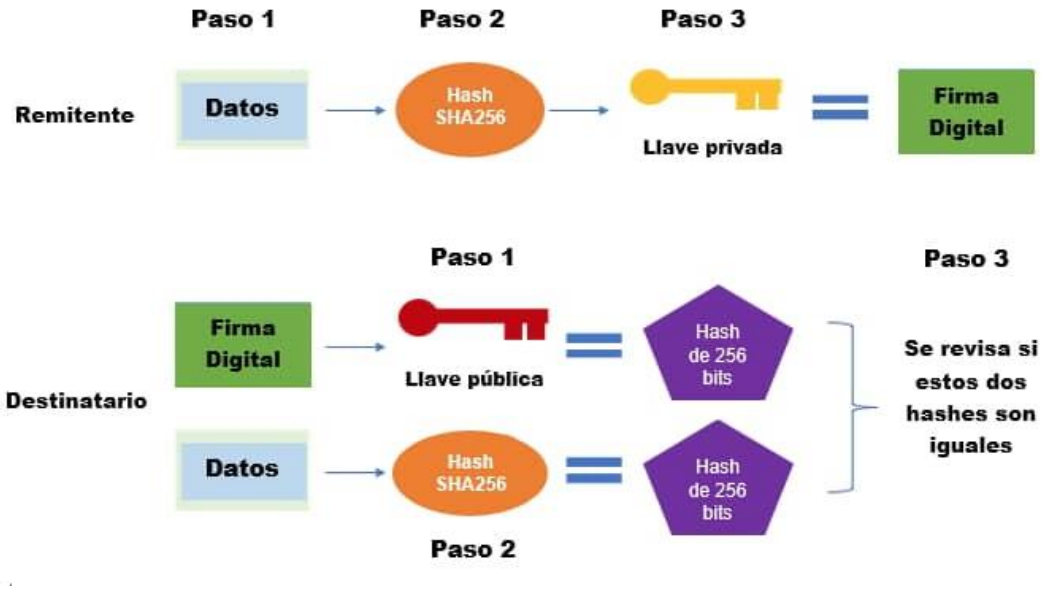


Figura 7-2. Esquema funcionamiento firma digital

Fuente: (Marshall, 2018)

- ✓ No importa la herramienta con la que fue creada la información que se va a preservar.
- ✓ Con esta técnica se indica si alguna función específica del software no continuará en el futuro.

- **Árboles de Merkle**

Los árboles de Merkle son estructuras de datos donde cada nodo interno (que no es hoja) almacena un hash de su nodo hijo, de forma que estos se van concatenando hasta llegar al nodo raíz (ver Figura 8-2). El hash del nodo raíz, al estar ligado al de todos sus hijos, permite la verificación de los mismos de forma segura y eficiente. Blockchain utiliza los árboles de Merkle para almacenar las transacciones dentro de los bloques.

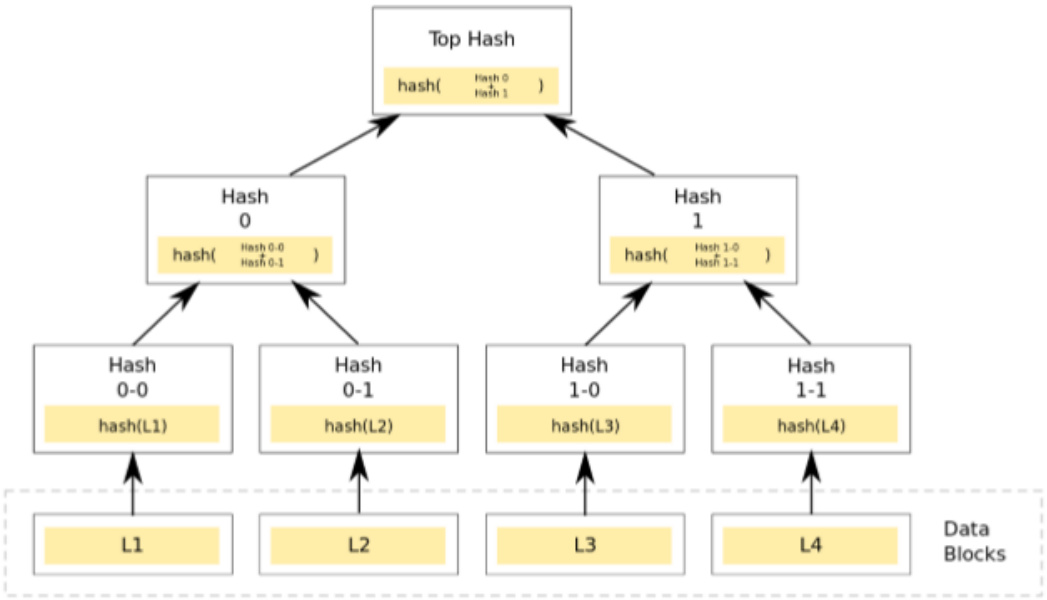


Figura 8-2. Ejemplo árbol de Merkle

Fuente: (Bashir, 2017)

Se puede diferenciar entre tres tipos de blockchain, atendiendo a la accesibilidad que tienen los usuarios. En primer lugar tenemos las blockchain públicas, las cuales son accesibles por cualquier usuario. Es decir, cualquier usuario podría participar en la red y leer las transacciones realizadas en la cadena. Bitcoin o Ethereum son ejemplos de estas. Por otra parte, las blockchain privadas establecen que sólo los usuarios escogidos formen parte de la red, y además son los únicos que pueden leer las transacciones. Hyperledger y R3 son los ejemplos más conocidos de las cadenas de bloques privadas. Por último, las blockchain híbridas combinan los conceptos de pública y privada, de modo que sólo determinados usuarios pertenecen a la red y pueden realizar transacciones, pero estas son públicas y accesibles por cualquier usuario. Un ejemplo de este tipo de blockchain es BigchainDB.

2.7. Algoritmos de consenso

Los algoritmos de consenso son mecanismos que permiten confirmar las transacciones, garantizando su integridad e inmutabilidad. Cada nodo de la red tiene que aplicar el algoritmo de consenso establecido para llegar a un acuerdo con el resto de nodos, de forma que compruebe que el nodo y las transacciones del mismo son válidas. Una vez se hayan realizado estas comprobaciones, se añade el bloque a la cadena y se replica en el resto de nodos. El algoritmo de

consenso más empleado es la prueba de trabajo (Proof of Work). Fue el primero que surgió, y aún se utiliza en la gran mayoría de redes de blockchain públicas, como son el caso de Bitcoin y de Ethereum. La explicación previa del funcionamiento de la cadena de bloques se ha realizado teniendo en cuenta este algoritmo, donde cada nodo de la red ha de verificar cada bloque, calculando el nonce atendiendo a la dificultad de minado. El gran inconveniente de la prueba de concepto es el gran coste computacional que conlleva, siendo este cada vez mayor a medida que crece la red, afectando a la escalabilidad de la misma. Con el objetivo de disminuir este coste computacional surge la prueba de participación (Proof of Stake), donde se busca llegar al consenso por medio de operaciones computacionalmente más sencillas. Esto se consigue porque se elige al creador del nodo de la red de manera determinista, en función de la participación o riqueza que este haya acumulado en la red. Por otro lado, tenemos la prueba de autoridad (Proof of Authority). Este algoritmo está pensado para su uso en redes privadas, y su rendimiento es más eficiente que el de la prueba de trabajo o la prueba de participación. En este se tienen en cuenta las identidades reales de los dueños de los nodos, de forma que se elige un conjunto de estos en los que se confíe para que sean los únicos que puedan añadir bloques en la blockchain. Este algoritmo no solo es el más eficiente, sino que es el que proporciona más escalabilidad a la red y es imprescindible en redes privadas donde la velocidad de las operaciones sea vital. (Bashir, 2017)

2.8. Smart Contract

Un smart contract, también conocido como contrato inteligente es un programa de computadora que consiste en un conjunto de reglas que se ejecutan en la blockchain. Con el auge de la tecnología blockchain en la última década, que muestra que tiene muchas áreas de aplicación. La integración de la tecnología blockchain y el contrato inteligente brinda mucha flexibilidad para desarrollar y diseñar, así como para implementar algunos de los problemas del mundo real en menos costo y tiempo sin la participación del sistema tradicional basado en terceros.

Un contrato inteligente es un programa de computadora que tiene propiedades autoverificables, autoejecutables y resistentes a la manipulación. El concepto de contrato inteligente fue propuesto por Nick Szabo en 1994. Permite ejecutar código sin los terceros. Un contrato inteligente consiste en el valor, la dirección, las funciones y el estado. Toma la transacción como entrada, ejecuta el código correspondiente y activa los eventos de salida. Dependiendo de la función lógica, los estados de implementación son cambios. Desde 2008, cuando la tecnología blockchain surge a través de la criptomoneda Bitcoin. La importancia de la integración inteligente de contratos de la tecnología blockchain se convierte en un área de enfoque para desarrollar, ya que permite que las transacciones

y la base de datos entre pares se puedan mantener públicamente de manera segura en un entorno confiable. Los contratos inteligentes son rastreables e irreversibles. Toda la información de la transacción está presente en un contrato inteligente y se ejecuta automáticamente.

El lenguaje de programación Solidity se usa para implementar el contrato inteligente en varias plataformas blockchain. Algunas características de un contrato inteligente son:

- El contrato inteligente es un código legible por máquina que se ejecuta en la plataforma blockchain.
- Los contratos inteligentes son programas impulsados por eventos
- Los contratos inteligentes son autónomos una vez creados, no es necesario monitorearlos
- Los contratos inteligentes se distribuyen.

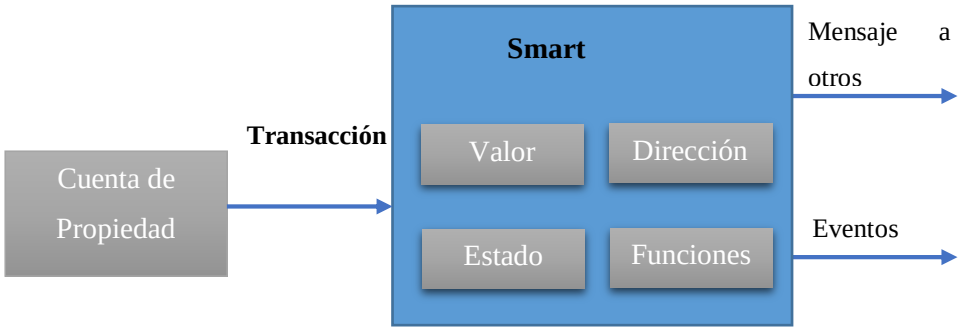


Figura 9-2. Esquema de un SmartContract

Realizado por: Carrillo Carina, 2020. Basado en (Miranda, 2018)

CAPÍTULO III

3. METODOLOGIA DE LA INVESTIGACIÓN

La metodología corresponde a los medios a través de los cuales se obtienen los datos y la información pertinente para llevar a cabo el desarrollo del estudio.

En este apartado se detallará los medios, procedimientos, y técnicas utilizados para el diseño del método de autenticación que asegure una aplicación web. Además se detallan las herramientas que se utilizaron para la construcción de los prototipos.

3.1. Tipo de investigación

La presente investigación es de tipo experimental, debido a que se verificará si la hipótesis planteada anteriormente tiene validez en el marco de todo el trabajo investigativo. Adicionalmente tiene un diseño transversal, puesto que se detalla la funcionalidad de cada algoritmo, proceso y sistema con todas sus implicaciones desde lo básico a lo complejo.

3.2. Diseño de la Investigación

La investigación es del tipo experimental, que a partir de las ventajas y características de la tecnología Blockchain se diseñó un método de autenticación basada en 2 factores para mejorar la seguridad de los sistemas web tradicionales y que fue evaluado mediante el uso de analizadores de vulnerabilidad.

3.3. Método y Técnicas de Investigación

3.3.1. Métodos

El método de la investigación es inductivo-deductivo, puesto que parte del estudio de la tecnología blockchain para aplicarla en el desarrollo de un método de autenticación descentralizado para mejorar la seguridad de los sistemas web.

El procedimiento a seguir es el siguiente:

1. Consulta documental (Registros, Internet, bibliografía científica, investigaciones realizadas en el país).
2. Análisis bibliográfico: con la finalidad de determinar la arquitectura más adecuada para el sistema distribuido.
3. Diseño de la cadena de bloques.
4. Programación de los Smart Contracts en la plataforma Ethereum.
5. Diseño de un sistema web seguro.

3.3.2. Técnicas

Se basa en revisión de fuentes de información bibliográficas primarias como Pruebas y Observación de resultados y secundarias como leyes, artículos científicos, tesis. Conferencias, revistas, etc. Las técnicas usadas en el presente trabajo de investigación son:

Las técnicas que se utilizaron en esta investigación son

- Búsqueda de Información: Permite obtener la información necesaria acerca del objeto de estudio de la investigación para su desarrollo, utilizando las fuentes secundarias disponibles.
- Pruebas: Realiza experimentos en escenarios de laboratorio.
- Observación: Permite determinar los resultados de las pruebas realizadas en los escenarios de laboratorio.
- Análisis: Determina los resultados de la investigación

3.4. Intrumentos

Para la investigación se ha optado por herramientas Open Source que permitieron la implementación de los escenarios y ejecutar las diferentes pruebas para la obtención de datos.
El uso de estas herramientas permitió levantar los ambientes de desarrollo adecuados para poder realizar la investigación sin inconvenientes.

3.4.1. Truffle.

Truffle es un entorno de desarrollo para Ethereum, que facilita el desarrollo de Smart Contracts y de aplicaciones que se comuniquen con la blockchain, denominadas DApps. Truffle es una de las tres herramientas que pertenecen a la Suite de Truffle (ver Figura 1-3).

Figura 1-3. Truffle Suite

Fuente: (Truffle, 2019)

Gracias a este entorno podremos testear los contratos inteligentes, compilarlos y migrarlos a la blockchain de una manera cómoda y sencilla.

3.4.2. Web3

Web3 es una colección de librerías que permiten interactuar con un nodo de una blockchain de Ethereum, utilizando una conexión HTTP, WebSocket o IPC. Se ha utilizado web3.js, la versión de JavaScript, pues el servidor web que queremos conectar con la blockchain ha sido desarrollado con NodeJS.

3.4.3. Node JS

Node.js fue ideado como un entorno de ejecución de JavaScript orientado a eventos asíncronos, Node.js está diseñado para crear aplicaciones network escalables. Actualmente es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor basado en el lenguaje de programación ECMAScript, asíncrono, con I/O de datos en una arquitectura orientada a eventos, por lo cual se lo escogió como lenguaje de programación para la implementación de los prototipos. (***Figura 2-3***).

Figura 2-3. Node JS

Fuente: (OpenJS Foundation, 2019)

3.4.4. Angular

Angular es un framework Javascript potente, muy adecuado para el desarrollo de aplicaciones frontend modernas, de complejidad media o elevada. El tipo de aplicación Javascript que se desarrolla con Angular es del estilo SPA (Single Page Application) o también las denominadas PWA (Progressive Web App).

El framework Angular ofrece una base para el desarrollo de aplicaciones robustas, escalables y optimizadas, que promueve además las mejores prácticas y un estilo de codificación homogéneo y de gran modularidad.

Figura 3-3. Angular

Fuente: (Google Inc., 2019)

3.4.5. JSON Web Token

Un JSON Web Token (JWT) es un token de acceso estandarizado según RFC 7519, que hace posible que dos partes intercambien datos de forma segura. Contiene toda la información importante sobre una entidad, lo que significa que no es necesario realizar consultas en la base de datos y no es necesario guardar la sesión en el servidor. Esta información puede ser verificada al estar firmada. En la estructura de un JWT (***ver Figura 4.3***) pueden distinguirse tres partes: la cabecera, la carga útil y la firma.

Figura 4-3. Estructura de un JSON Web Token

Fuente: (Sanchez, 2020)

JWT es especialmente popular en los procesos de autenticación. Sus mensajes cortos se pueden cifrar y transmitir de forma segura quién es el remitente y si tiene los derechos de acceso necesarios. Los propios usuarios solo entran en contacto indirecto con el token, por ejemplo, cuando ingresan nombres de usuario y contraseñas en una máscara. La comunicación real tiene lugar entre el cliente y el servidor.

3.4.6. Postman

Postman es una plataforma de colaboración para el desarrollo de API. Las funciones de Postman simplifican cada paso de la creación de una API y agilizan la colaboración para que pueda crear mejores API más rápido.

Figura 5-3. Postman

Fuente: (Postman, Inc, 2019)

3.4.7. Visual Studio Code

Visual Studio Code es un editor de código optimizado con soporte para operaciones de desarrollo como depuración, ejecución de tareas y control de versiones. Su objetivo es proporcionar solo las herramientas que un desarrollador necesita para un ciclo rápido de código, compilación y depuración y deja flujos de trabajo más complejos a IDE con funciones más completas, como Visual Studio IDE.

Figura 6-3. Visual Studio Code

Fuente: (Microsoft, 2019)

3.4.8. Heroku

Heroku es una plataforma que permite a los desarrolladores crear, ejecutar y administrar aplicaciones completamente en la nube. Heroku a diferencia de otras plataformas permite desarrollar prácticamente con cualquier lenguaje de programación: Ruby, Java, PHP, NodeJS, entreo otros.

Figura 7-3. Heroku

Fuente: (Salesforce, 2019)

3.4.9. Vooki

Vooki es un escáner de vulnerabilidades de aplicaciones web gratuito, fácil de usar que puede escanear fácilmente cualquier aplicación web y encontrar las vulnerabilidades. Vooki incluye un escáner de aplicaciones web, un escáner de API de descanso y una sección de informes.

3.4.10. Owasp Zap

El OWASP Zed Attack Proxy (ZAP) es una de las herramientas de seguridad gratuitas más populares del mundo y permite encontrar automáticamente vulnerabilidades de seguridad en las aplicaciones web mientras se desarrolla. Además permire realizar pruebas de seguridad manuales.

3.5. Fuentes de Información

Las fuentes de información que se usaron en este proyecto de investigación son:

Primarias

- Pruebas Aplicadas.
- Observación y análisis de los resultados obtenidos en las pruebas.

Secundarias

- Articulos Científicos.
- Tesis relacionadas con el tema de investigación.
- Libros relacionados al tema.
- Páginas Web con contenido seguro.
- Páginas oficiales de recursos tecnológicos.

3.6. Hipótesis

Un sistema de autenticación basado en la tecnología blockchain si mejora la seguridad en un sistema web.

3.6.1. Identificación de Variables

Variable Independiente: Sistema de autenticación basado en la tecnología blockchain.

Variable Dependiente: Incremento de la seguridad en el proceso de autenticación de un sistema web.

3.6.2. Operacionalización Conceptual de Variables

La Tabla 1-3, muestra la operacionalización conceptual de las variables determinadas.

Tabla 1-3. Operacionalización conceptual de variables

VARIABLE	TIPO	CONCEPTO
Sistema de autenticación basado en la tecnología blockchain.	Variable Independiente	Diseñar un nuevo modelo de autenticación basado en la tecnología blockchain y contratos inteligentes.
Incremento de la seguridad en el proceso de autenticación de un sistema web.	Variable Dependiente	Disminuir la probabilidad de vulnerabilidades en la seguridad de un sistema web..

Realizado por: Carrillo Carina, 2020

3.6.3. Operacionalización metodológica de variables

La Tabla 2-3, muestra la operacionalización metodológica de las variables determinadas.

Tabla 2-3. Operacionalización metodológica de variables

VARIABLE	INDICADOR	TÉCNICA	INSTRUMENTO/FUENTE
Sistema de autenticación basado en la tecnología blockchain.	Parámetros de Seguridad	Observación Evaluación Análisis	Aplicativo implementado con método propuesto.
Incremento de la seguridad en el proceso de autenticación de un sistema web.	Número de vulnerabilidades detectadas.	Observación y análisis de la evaluación aplicada	Ambiente de Pruebas

Realizado por: Carrillo Carina, 2020

3.7. Población y Muestra

3.7.1. Población

La población de estudio está conformada por las propiedades de tecnología blockchain implementadas en el sistema de seguridad propuesto:

- 1 Prototipo Web con un sistema de autenticación tradicional.
- 1 Prototipo Web aplicando el sistema de autenticación propuesto.

3.7.2. Selección de la Muestra

Para la selección de la muestra se consideró las propiedades más relevantes que permitirían mejorar la seguridad de un sistema web y que ayudaron a determinar un método de autenticación y que finalmente fueron probados al aplicar la propuesta en uno de los ambientes implementados como se define en la ***Tabla 3-3***.

Tabla 3-3. Tabla de parámetros a evaluar

No	Parámetro de seguridad
1	Descentralización
2	Inmutabilidad
3	Transparencia
4	Uso hashes criptográficos
5	Firmas digitales
6	Generación Tokens

Realizado por: Carrillo Carina, 2020

Para generalizar los resultados de la investigación que valide la propuesta del método planteado, se aplicaron pentesting que ayudaron a validar la aplicabilidad de los seis (6) parámetros definidos (***Tabla 3-3***).

3.8. Procedimientos Generales

Para la obtención de datos que validen la investigación se procedió a aplicar la observación con la finalidad de verificar si el sistema propuesto cubre todos los requerimientos de seguridad en el desarrollo y además se revisó los resultados obtenidos de la utilización de herramientas de pentesting aplicadas a los dos escenarios del aplicativo (***Tabla 4-3***).

Tabla 4-3. Técnicas de demostración de hipótesis

VARIABLE	INDICADOR	TÉCNICA
Diseño de un sistema de autenticación basado en la tecnología blockchain.	• Parámetros de seguridad	Observación y análisis
Nivel de Seguridad	• Número de vulnerabilidades detectadas	**Pruebas de Aplicativos** • Utilización de herramientas Vooki y Owasp ZAP para detección de vulnerabilidades.

Realizado por: Carrillo Carina, 2020

3.9. Instrumentos de Recolección de Datos

Para la recolección de datos para la investigación se utilizaron las herramientas Open Source explicadas anteriormente en la sección Instrumentos, como son Vooki, Owasp Zap y Postman que ayudaron a realizar la detección de vulnerabilidades en los dos escenarios.

3.10. Instrumentos para Procesar Datos Recopilados

Para el procesamiento de los datos obtenidos se utilizó software específico para tabulación y análisis estadístico como son Microsoft Excel y la herramienta de matemáticas de nivel educativo Online GeoGebra2 para realizar el gráfico de chi-cuadrado.

3.11. Ambiente de Pruebas

Para crear el ambiente de pruebas se diseñaron e implementaron dos aplicativos diseñados en Nodejs, los mismos que permitirán obtener y verificar los resultados.

3.11.1. Escenarios

Se detalla la arquitectura de cada uno de los prototipos diseñados. (Ver ***Figura 8-3 y Figura 9-3***):

Escenario 1

El aplicativo se implementó con el método tradicional de autenticación, basado en un usuario y una contraseña, los mismos que se almacenarán en una base de datos contruida en MySql y se publicó en un servicio de la nube llamado Heroku.

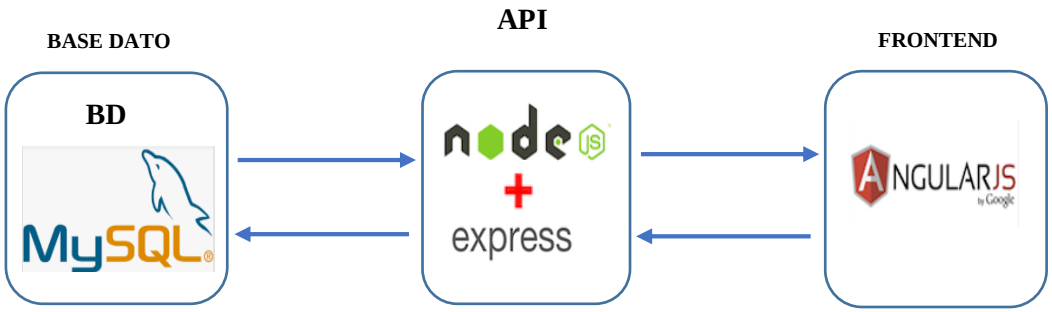


Figura 8-3. Arquitectura del Prototipo I

Realizado por: Carrillo Carina, 2020

Escenario 2

El aplicativo se implementó con el método propuesto y para la implementación del doble factor se creo una cadena blockchain, la cual guarda la información del login del usuario con sus métodos de criptografia, se diseño un contrato inteligente para almacenar la información en la blockchain con la finalidad de generar una cadena mas segura. El web service se publicó en un servicio de la nube llamado Heroku.

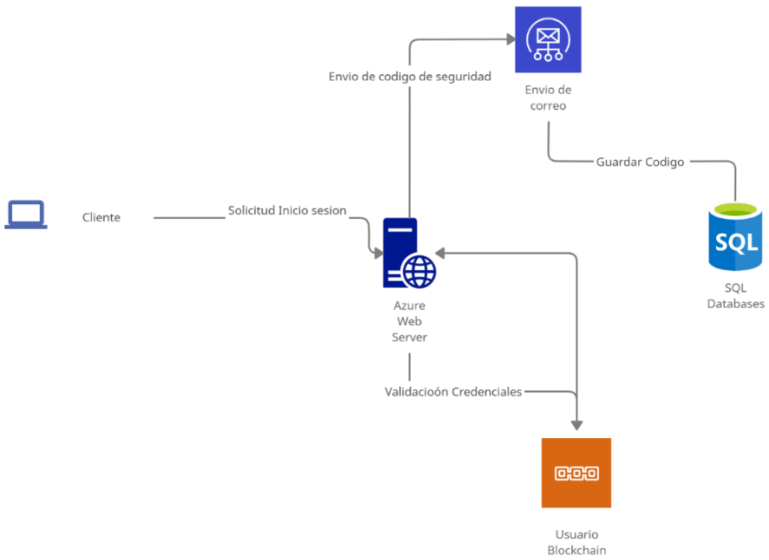


Figura 9-3. Arquitectura del Prototipo II

Realizado por: Carrillo Carina, 2020

3.11.2. Resultados

Para obtener los resultados necesarios para el estudio, se definieron las siguientes pruebas para cado uno de los escenarios:

Una vez implementado y publicados los Escenarios, se procedió a realizar la verificación del cumplimiento de aplicabilidad en cada uno de los parámetros definidos en la Tabla 3-3 usando la herramienta POSTMAN, ejecutando en éste cliente las acciones principales de HTTP (POST, PUT, GET, DELETE) y en base a las respuestas que se obtiene con la herramienta se registraron los datos en una tabla comparativa de los dos Escenarios.

Para la obtención de los datos para verificar el nivel de seguridad se utilizó la herramienta VOOKI que permitió realizar el pentesting y obtener las vulnerabilidades en cada uno de los Escenarios y

sobre dos acciones como son en el proceso de Autenticación y en la solicitud GET en sesión iniciada.

La herramienta Owasp Zap luego de establecer la configuración deseada, para encontrar vulnerabilidades que presenten los prototipos, con relación a los ataques inyección de SQL y XSS. Se debe iniciar la herramienta para su ejecución e ingresar la URL de las aplicaciones, por consiguiente, el programa ejecuta el análisis para poder encontrar las vulnerabilidades, es de resaltar que esta herramienta muestra cuatro tipos de alertas, como a continuación se enumeran:

1. Alertas con Alta prioridad.
2. Alertas con Prioridad media
3. Alertas con Baja prioridad.
4. Alertas informativas.

Posterior a la aplicación de las pruebas definidas para los dos escenarios, se recolectaron los datos numéricos necesarios para aplicar la distribución, con el objetivo de demostrar que el método de autenticación utilizado mejoró la seguridad de la aplicación web.

CAPÍTULO IV

4. RESULTADOS Y DISCUSIÓN

En este capítulo se evalua los dos escenarios propuestos descritos en el capítulo anterior para desarrollar la comprobación de las hipótesis.

4.1. Presentación de Resultados

En esta sección se presentan los resultados obtenidos en la investigación y sus respectivo análisis mediante métodos y técnicas definidos, además se determina su relación con los objetivos y la hipótesis planteada.

Analizando los resultados, se puede observar que el método de autenticación propuesto permite mejorar la seguridad de una aplicación web.

4.2. Procesamiento y Análisis

En la presente investigación se aplicó la observación y un chequeo de cumplimiento de los parámetros a evaluar para la demostración de la variable independiente y el uso de las herramientas pentesting para la demostración de la variable dependiente aplicando las pruebas sobre los dos escenarios planteados.

4.3. Valoración de la Variable Independiente

*4.3.1. **Variable Independiente:** Sistema de autenticación basado en la tecnología blockchain*

Para su valoración se procedió a realizar una observación y la verificación del cumplimiento del uso los parámetros planteados para la evaluación en cada escenario.

4.3.2. Indicador: Nivel de cumplimiento.

Nivel de cumplimiento de los parámetros de seguridad en la aplicación: Para la medición de este indicador se utilizó la escala de Likert (Matas, 2018) que permitió tener una valoración del nivel de cumplimiento de los parámetros de seguridad. (***Tabla 1-4***).

Tabla 1-4. Escala Likert Nivel de cumplimiento

Escala	**Muy bajo**	**Bajo**	**Medio**	**Alto**	**Muy Alto**
Valoración	1	2	3	4	5

Fuente: (Matas, 2018)
Realizado por: Carrillo Carina, 2020

La observación realizada sobre los escenarios definidos utilizando la herramienta POSTMAN en base a los resultados obtenidos de ésta y aplicando la escala anterior (Tabla 1-4), se define la siguiente Tabla 2-4.

Tabla 2-4. Parámetros evaluados en los escenarios definidos

	Parámetro	**Escenario 1**	**Escenario 2**
1	Descentralización	1	5
2	Inmutabilidad	2	5
3	Transparencia	2	2
4	Uso hashes criptográficos	2	5
5	Firmas digitales	2	5
6	Generación Tokens	2	4
	Total	**11**	**26**

Realizado por: Carrillo Carina, 2020

Análisis e Interpretación de Resultados:

De estos datos el porcentaje de seguridad se calcula como valor de 30 al 100% ya que indicaría que el sistema siempre implementa los seis (6) parámetros de seguridad a evaluar. Dicho esto, los valores de aplicabilidad obtenidos en la observación corresponden al 36.67% en el Escenario 1 y el 86.67% en el Escenario 2 al aplicar el método propuesto de lo que se puede apreciar una mejora con el sistema propuesto.

4.4. Valoración de la Variable Dependiente

4.4.1. Variable Dependiente: Nivel de Seguridad

Para realizar el proceso de valoración se utilizó la herramienta de pentesting Vooki y Owasp ZAP aplicada a los dos escenarios implementados.

4.4.2. Indicador: Vulnerabilidades del Sistema a nivel de API

Número de vulnerabilidades detectadas en la aplicación: Para realizar la medición de este indicador se utilizó la herramienta Vooki aplicando el scanner de vulnerabilidad a cada una de las APIs desarrolladas.

Identificación de vulnerabilidades sobre el Escenario 1 (API sin blockchain), (Figura 1-4)

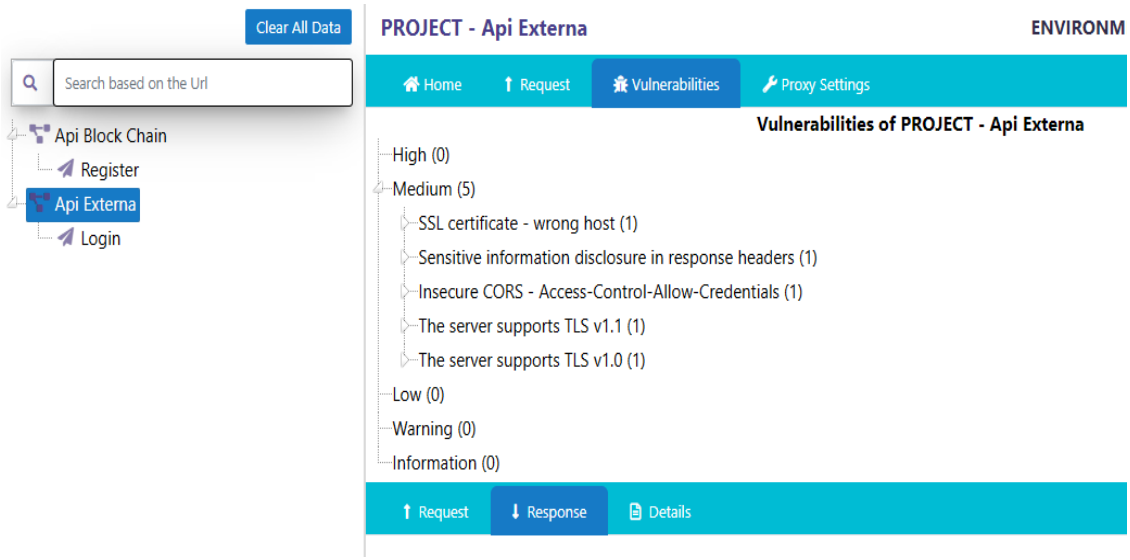


Figura 1-4. Captura de pentesting en Autenticación Vooki en Escenario 1

Realizado por: Carrillo Carina, 2020

A continuación se detalla las vulnerabilidades presentadas en el Prototipo I y se detalla brevemente el problema presentado.

Tabla 3-4. Vulnerabilidades detectadas en Escenario 1

Vulnerabilidad	Nivel	Número	Descripción
Detección de protocolo TLS versión 1.0 y 2.0	Medio	2	Se encuentra habilitado TLS para su uso en todas las conexiones abiertas.
Divulgación de información sensible en encabezados de respuesta	Medio	1	Presencia de información sensible de configuración en la aplicación.
Certificado SSL	Medio	1	El atributo 'commonName' (CN) del certificado SSL presentado para este servicio es para una máquina diferente.
CORS Inseguros	Bajo	1	Para la ejecución del pentensting se habilito el acceso a todas las urls con la opción de "access-control-allow-origin:*" para dar acceso a la herramienta.

Realizado por: Carrillo Carina, 2020

Identificación de vulnerabilidades sobre el Escenario 2 (API con blockchain), (Figura 2-4)

Servicio Restful Api Block Chain

Url: https://blockchainlogintest.herokuapp.com/register/users

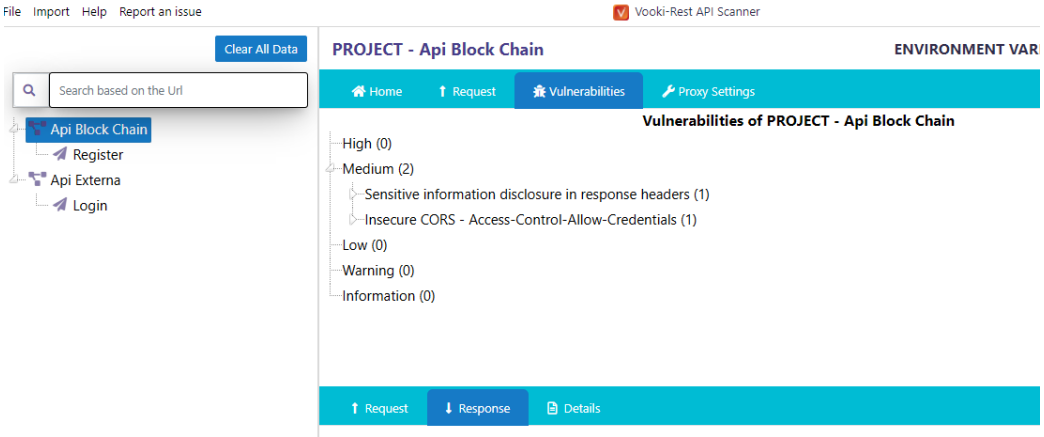


Figura 2-4. Captura de pentesting con Vooki en Escenario 2

Realizado por: Carrillo Carina, 2020

Tabla 4-4. Vulnerabilidades en Autenticación encontradas en Escenario 2

Vulnerabilidad	Nivel	Número	Descripción
Divulgación de información sensible en encabezados de respuesta	Medio	1	Presencia de información sensible de configuración en la aplicación.
CORS Inseguros	Bajo	1	Para la ejecución del pentensting se habilito el acceso a todas las urls con la opción de "access-control-allow-origin:*" para dar acceso a la herramienta

Realizado por: Carrillo Carina, 2020

4.4.3. Indicador: Vulnerabilidades del Sistema a nivel de Urls

Vulnerabilidades del Sistema a nivel de Urls detectadas en la aplicación: Para realizar la medición de este indicador se utilizó el escáner de la herramienta Owasp Zap.

Identificación de vulnerabilidades a nivel de Urls sobre el Escenario 1 (API sin blockchain), (Figura 3-4)

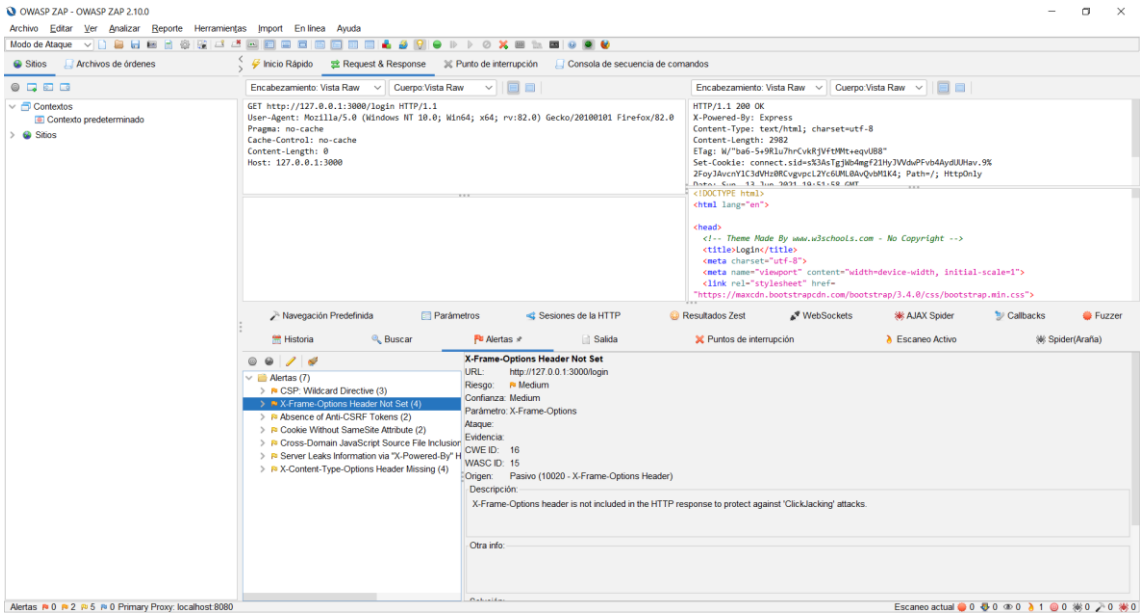


Figura 3-4. Captura de escaneo en Owasp Zap para el Escenario 1

Realizado por: Carrillo Carina, 2020

Identificación de vulnerabilidades a nivel de Urls sobre el Escenario 2 (API con blockchain), (Figura 4-4)

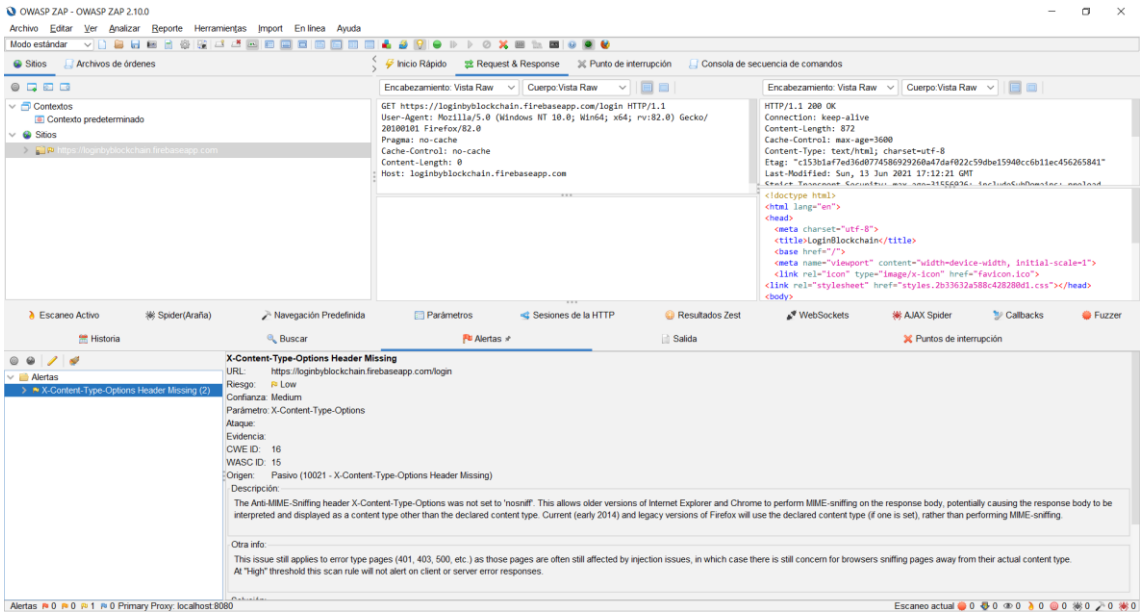


Figura 4-4. Captura de escaneo en Owasp Zap para el Escenario 2

Realizado por: Carrillo Carina, 2020

Una vez aplicado los pentesting sobre los dos escenarios y una vez obtenido los resultados de las vulnerabilidades se obtiene la siguiente ***Tabla 5-4*** con el total de ellos entre los dos indicadores seleccionando las vulnerabilidades sin repetición, además se anexan como 2 vulnerabilidades altas los ataques comunes como XSS e inyección SQL, ataques que en el escenario 2 fallaron porque no hay un servidor central donde se almacena la información de inicio de sesión:

Tabla 5-4. Resumen de resultados obtenidos en los pentesting

Vulnerabilidades encontradas	**Frecuencia**		**Porcentaje de Reducción**
	Escenario 1	**Escenario 2**	
Altas	2	0	100%
Medias	5	1	80 %
Bajas	5	1	80 %
Total	**12**	**2**	**83.33%**

Realizado por: Carrillo Carina, 2020

A continuación se representan los resultados obtendidos de forma gráfica para proceder a realizar el respectivo análisis de los resultados.

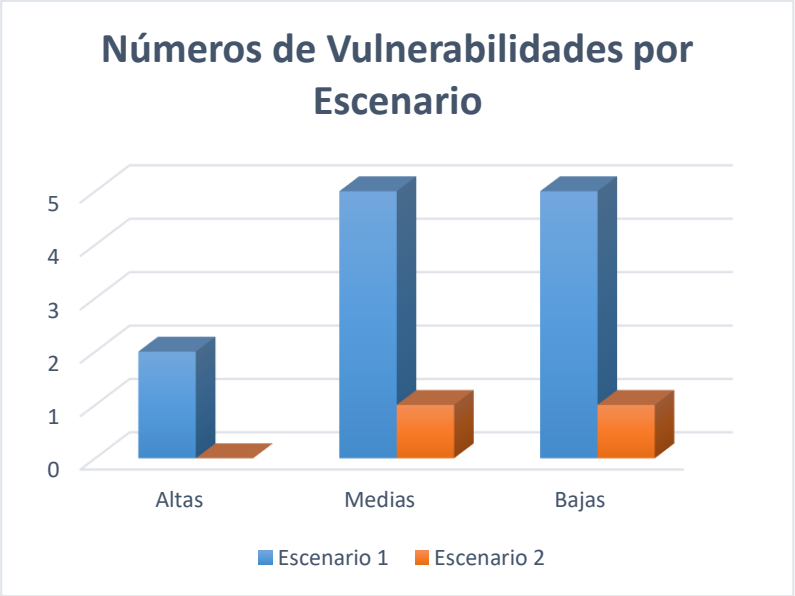


Gráfico 1-4. Número de vulnerabilidades por Escenario

Realizado por: Carrillo Carina, 2020

Análisis e Interpretación de Resultados:

Al incluir la tecnología blockchain en el método de autenticación podemos concluir que se ha reducido en un 100% las vulnerabilidades de Nivel Alto, en un 80% las vulnerabilidades de Nivel medio y bajo; dando como resultado total un 83.33% de reducción con la aplicación del método propuesto.

4.5. Comprobación Estadística de la Hipótesis

Para la comprobación de la hipótesis general "Un sistema de autenticación basado en la tecnología blockchain si mejora la seguridad en un sistema web", se utilizó estadística referencial aplicando la prueba Chi-Cuadrado(X^2).

Posterior a la realización de los diferentes análisis, y con los datos obtenidos se procede a definir la hipótesis de investigación *Hi* y la Hipótesis nula *Ho* a ser consideradas:

H_0: $\mu_{aud\_1} = \mu_{aud\_2} = \mu_{st\_1} = \mu_{st\_2}$

H_1: $\mu_i \neq \mu_j$ para al menos un i ≠ j

Con base en la expresión matemática de las hipótesis nula y alternativa, se definen las hipótesis estadísticas a ser consideradas en el presente estudio:

Hi: "Un sistema de autenticación basado en la tecnología blockchain **si** mejora la seguridad en un sistema web."

Ho: "Un sistema de autenticación basado en la tecnología blockchain **no** mejora la seguridad en un sistema web."

La siguiente ***Tabla 6-4*** contiene la información de las frecuencias de valores encontrados, los mismos que serán utilizados para el cálculo en la prueba.

Tabla 6-4. Frecuencias de Valores Encontrados

	Escenario 1	**Escenario 2**	**Total**
Vulnerabilidades Encontradas	10	2	12
Vulnerabilidades Solventadas	2	11	13
Total	12	2	25

Realizado por: Carrillo Carina, 2020

La siguiente fórmula se aplica a cada una de las celdas de las frecuencias observadas para calcular las frecuencias esperadas:

$$Fe = \frac{total\ columna * total\ fila}{N}$$

Donde:

N: Número total de frecuencias observadas.

Luego de aplicar la fórmula a cada uno de los datos de la Tabla 6-4 se consigue la tabla de frecuencias de los valores esperados, como muestra la Tabla 7-4.

Tras la aplicación de la fórmula en cada valor de la tabla anterior obtendremos la siguiente tabla:

Tabla 7-4. Frecuencias esperadas.

	Escenario 1	**Escenario 2**	**Total**
Vulnerabilidades Encontradas	5.76	6.24	12
Vulnerabilidades Solventadas	6.24	6.76	13
Total	12	13	25

Realizado por: Carrillo Carina, 2020

A continuación, se calcula el valor de x^2 mediante la siguiente fórmula:

$$x^2 = \sum \frac{(FO - FE)^2}{FE}$$

Donde:

FO: Frecuencia Observada por celda

FE: Frecuencia Esperada por celda

$$x^2 = \frac{(10 - 5.76)^2}{5.76} + \frac{(2 - 6.24)^2}{6.24} + \frac{(2 - 6.24)^2}{6.24} + \frac{(11 - 6.76)^2}{6.76}$$

$$x^2 = 3.12 + 2.88 + 2.88 + 2.66$$

$$x^2 = 11.54$$

El siguiente paso a seguir es el cálculo de los grados de libertad:

$$v = (r - 1)(k - 1)$$

Donde:

r: número de filas

k: número de Columnas

$$v = (2-1)(2-1)$$

$$v = 1$$

En base a la tabla de la distribución de Chi-Cuadrado (Figura 5-4), y determinando el valor de significancia de 0.05% obtenemos el punto crítico con 1 como valor de grados de libertad.

TABLA 3-Distribución Chi Cuadrado χ^2

P = Probabilidad de encontrar un valor mayor o igual que el chi cuadrado tabulado, ν = Grados de Libertad

ν/p	0,001	0,0025	0,005	0,01	0,025	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5
1	10,8274	9,1404	7,8794	6,6349	5,0239	3,8415	2,7055	2,0722	1,6424	1,3233	1,0742	0,8735	0,7083	0,5707	0,4549
2	13,8150	11,9827	10,5965	9,2104	7,3778	5,9915	4,6052	3,7942	3,2189	2,7726	2,4079	2,0996	1,8326	1,5970	1,3863
3	16,2660	14,3202	12,8381	11,3449	9,3484	7,8147	6,2514	5,3170	4,6416	4,1083	3,6649	3,2831	2,9462	2,6430	2,3660
4	18,4662	16,4238	14,8602	13,2767	11,1433	9,4877	7,7794	6,7449	5,9886	5,3853	4,8784	4,4377	4,0446	3,6871	3,3567
5	20,5147	18,3854	16,7496	15,0863	12,8325	11,0705	9,2363	8,1152	7,2893	6,6257	6,0644	5,5731	5,1319	4,7278	4,3515
6	22,4575	20,2491	18,5475	16,8119	14,4494	12,5916	10,6446	9,4461	8,5581	7,8408	7,2311	6,6948	6,2108	5,7652	5,3481
7	24,3213	22,0402	20,2777	18,4753	16,0128	14,0671	12,0170	10,7479	9,8032	9,0371	8,3834	7,8061	7,2832	6,8000	6,3458
8	26,1239	23,7742	21,9549	20,0902	17,5345	15,5073	13,3616	12,0271	11,0301	10,2189	9,5245	8,9094	8,3505	7,8325	7,3441
9	27,8767	25,4625	23,5893	21,6660	19,0228	16,9190	14,6837	13,2880	12,2421	11,3887	10,6564	10,0060	9,4136	8,8632	8,3428
10	29,5879	27,1119	25,1881	23,2093	20,4832	18,3070	15,9872	14,5339	13,4420	12,5489	11,7807	11,0971	10,4732	9,8922	9,3418
11	31,2635	28,7291	26,7569	24,7250	21,9200	19,6752	17,2750	15,7671	14,6314	13,7007	12,8987	12,1836	11,5298	10,9199	10,3410
12	32,9092	30,3182	28,2997	26,2170	23,3367	21,0261	18,5493	16,9893	15,8120	14,8454	14,0111	13,2661	12,5838	11,9463	11,3403
13	34,5274	31,8830	29,8193	27,6882	24,7356	22,3620	19,8119	18,2020	16,9848	15,9839	15,1187	14,3451	13,6356	12,9717	12,3398
14	36,1239	33,4262	31,3194	29,1412	26,1189	23,6848	21,0641	19,4062	18,1508	17,1169	16,2221	15,4209	14,6853	13,9961	13,3393
15	37,6978	34,9494	32,8015	30,5780	27,4884	24,9958	22,3071	20,6030	19,3107	18,2451	17,3217	16,4940	15,7332	15,0197	14,3389
16	39,2518	36,4555	34,2671	31,9999	28,8453	26,2962	23,5418	21,7931	20,4651	19,3689	18,4179	17,5646	16,7795	16,0425	15,3385
17	40,7911	37,9462	35,7184	33,4087	30,1910	27,5871	24,7690	22,9770	21,6146	20,4887	19,5110	18,6330	17,8244	17,0646	16,3382
18	42,3119	39,4220	37,1564	34,8052	31,5264	28,8693	25,9894	24,1555	22,7595	21,6049	20,6014	19,6993	18,8679	18,0860	17,3379
19	43,8194	40,8847	38,5821	36,1908	32,8523	30,1435	27,2036	25,3289	23,9004	22,7178	21,6891	20,7638	19,9102	19,1069	18,3376
20	45,3142	42,3358	39,9969	37,5663	34,1696	31,4104	28,4120	26,4976	25,0375	23,8277	22,7745	21,8265	20,9514	20,1272	19,3374
21	46,7963	43,7749	41,4009	38,9322	35,4789	32,6706	29,6151	27,6620	26,1711	24,9348	23,8578	22,8876	21,9915	21,1470	20,3372
22	48,2676	45,2041	42,7957	40,2894	36,7807	33,9245	30,8133	28,8224	27,3015	26,0393	24,9390	23,9473	23,0307	22,1663	21,3370
23	49,7276	46,6231	44,1814	41,6383	38,0756	35,1725	32,0069	29,9792	28,4288	27,1413	26,0184	25,0055	24,0689	23,1852	22,3369
24	51,1790	48,0336	45,5584	42,9798	39,3641	36,4150	33,1962	31,1325	29,5533	28,2412	27,0960	26,0625	25,1064	24,2037	23,3367
25	52,6187	49,4351	46,9280	44,3140	40,6465	37,6525	34,3816	32,2825	30,6752	29,3388	28,1719	27,1183	26,1430	25,2218	24,3366
26	54,0511	50,8291	48,2898	45,6416	41,9231	38,8851	35,5632	33,4295	31,7946	30,4346	29,2463	28,1730	27,1789	26,2395	25,3365
27	55,4751	52,2152	49,6450	46,9628	43,1945	40,1133	36,7412	34,5736	32,9117	31,5284	30,3193	29,2266	28,2141	27,2569	26,3363
28	56,8918	53,5939	50,9936	48,2782	44,4608	41,3372	37,9159	35,7150	34,0266	32,6205	31,3909	30,2791	29,2486	28,2740	27,3362
29	58,3006	54,9662	52,3355	49,5878	45,7223	42,5569	39,0875	36,8538	35,1394	33,7109	32,4612	31,3308	30,2825	29,2908	28,3361

Figura 5-4: Tabla de Distribución Chi Cuadrado

Fuente: Universidad Carlos III de Madrid-Departamento de Estadistica, 2019

$$x^2\ crítico = 3{,}8415$$

Dado los datos anteriores Ho debe ser aceptada si sucede el siguiente condicionante:

$$x^2\ Calculado \leq x^2\ crítico$$

Caso contrario se rechaza Ho y se acepta Hi.

Con los datos obtenidos anteriormente donde $x^2\ Calculado$ = 11.54 y x^2 critico = 3.8415 se puede aplicar el criterio de decisión y obtenemos que (***Gráfico 2-4***):

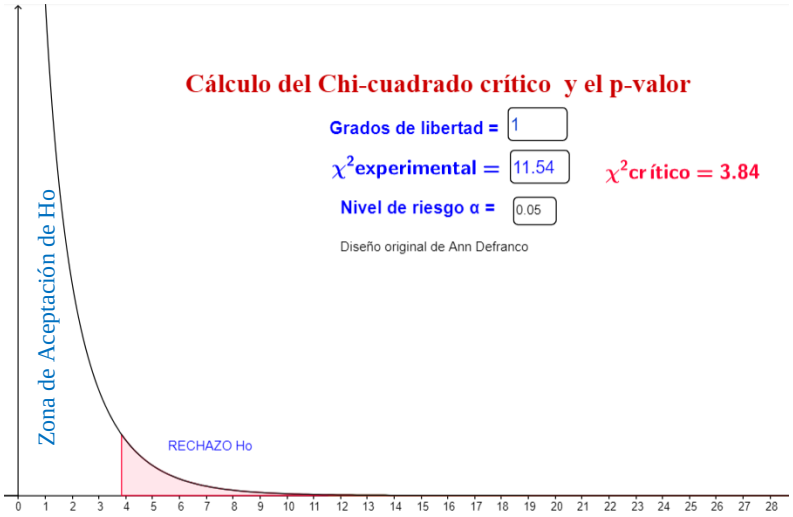


Gráfico 2-4. Chi-Cuadrado y Criterios de Aceptación de Ho

Fuente: GeoGebra, 2020
Realizado por: Carrillo Carina, 2020

Interpretación y Análisis

De acuerdo al gráfico el valor calculado de x^2 está dentro del sector de rechazo de la hipótesis nula Ho: "Un sistema de autenticación basado en la tecnología blockchain no mejora la seguridad en un sistema web.", por lo que se acepta la hipótesis alterna de la investigación Hi: "Un sistema de autenticación basado en la tecnología blockchain si mejora la seguridad en un sistema web.", con un nivel de significancia de $\alpha = 0.05$ para alcanzar un nivel de confianza del 95 %.

CAPÍTULO V

5. PROPUESTA

5.1. Determinación de la Propuesta

En este capítulo se describe el sistema propuesto para el marco 2FA basado en el contrato inteligente Blockchain para superar las vulnerabilidades que se basa en el uso de una autoridad central. El marco propuesto brinda una forma flexible, utilizable y segura de generar y validar un OTP que está encriptado y protegido contra la mayoría de los ataques comunes, que son inyección SQL, XSS y ataques de terceros. Los componentes del sistema propuesto incluyen la aplicación web, el usuario y el contrato inteligente de Ethereum. La autenticación se realiza en dos etapas: la primera al proporcionar el nombre de usuario y la contraseña a la aplicación / sitio web deseado, y la segunda es descifrando la OTP que se genera y se envía a través de un contrato inteligente que solicitó la aplicación web para autorizar al usuario, donde los bloques de registros se almacenan en la blockchain.

Uno de los beneficios de usar blockchain es la posibilidad de almacenar la base de datos de los usuarios en él. Por lo tanto, se puede deducir que la cadena de bloques no solo sirve para mantener un seguimiento inmutable de las transacciones, sino que también permite el almacenamiento de datos en ella.

Las ventajas de mantener la base de datos de usuarios almacenada en la cadena de bloques son:

- Mantener la base de datos en la cadena de bloques proporciona descentralización, lo que significa que si bien hay diferentes nodos que ejecutan la red, si alguno de ellos tiene un problema (por ejemplo, se apaga la electricidad del edificio), los otros mantienen la información segura, con respecto a ese hecho, una cadena de bloques se puede ejecutar con varios nodos diferentes distribuidos por todo el mundo. Esta descentralización proporciona seguridad no solo en caso de falla sino también en caso de un intento de corromper la cadena de bloques, si alguien intenta cambiar la información almacenada en un nodo siempre que los otros nodos no validen esta modificación en la cadena de bloques, no lo hará.

- Otro beneficio sería la inmutabilidad de los datos almacenados. Hasta ahora, algunos datos están almacenados en la cadena de bloques, será imposible cambiarlos o alterarlos. La cadena de bloques es una pista inmutable de bloques que almacenan información; este hecho evita la futura corrupción o alteración de datos por intenciones maliciosas.

- La confiabilidad también es un aspecto muy importante, si desea brindar confianza en su sistema. Al usar la tecnología blockchain, no es necesario confiar en terceros para verificar la información y las transacciones.

5.2. Diseño del Sistema Propuesto

El sistema propuesto basado en blockchain y controlado por un contrato inteligente que dicta el protocolo del sistema y actúa como una interfaz a la cadena de bloques para el gestión de identidades y atributos.

El contrato inteligente se centra en la entidad, que publica un conjunto de atributos, firmas y revocaciones en la cadena de bloques para su identidad. Cada entidad está representada en base a una dirección Ethereum que está controlada por una clave privada.

Debido a los costosos costos de gas asociados con el alamacenamiento en Ethereum se diseñó una blockchain personalizada para permitir el registro de los datos sensibles del usuario y para mantener la autenticidad de los datos proporcionando un hash criptográfico de los datos con el atributo en el blockhain.

5.2.1. Creación de la cadena de bloques

El proceso inicia en el constructor de la clase donde se inicia en memoria un arreglo, cada ves que se solicita la inserción de un bloque, se valida la información anterior ha ella, realizando un recorrido de en todos los bloques para identificar que no se genere un bloque repetido que pueda ocasionar conflictos con las consultas, esta cadena de bloques no tiene dependencias de bases de datos por los cual están protegidas a ataques de inyección SQL.

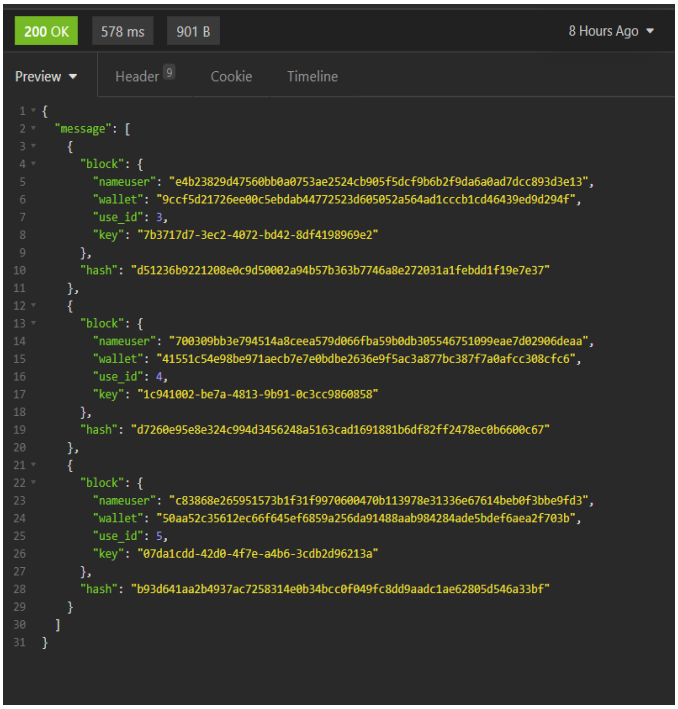


Figura 1-5. Bloques generados en la blockchain

Realizado por: Carrillo Carina, 2020

5.2.2. Despliegue del Smart Contract

Para la codificación de contratos inteligentes en solidity, se trabajó con Visual Studio Code, que es muy editor de código flexible y práctico para la codificación de solidity. Para la compilación, prueba e implementación nos apoyamos en la herramienta Truffle.

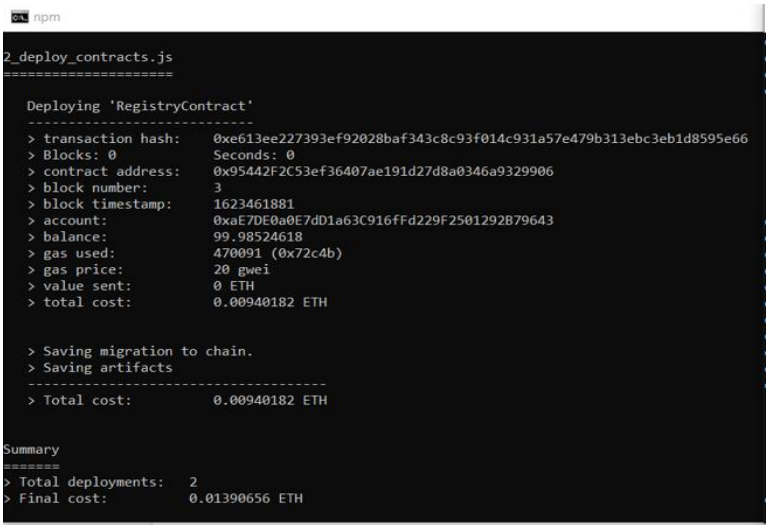


Figura 2-5. Despliegue del Contrato

Realizado por: Carrillo Carina, 2020

5.2.3. Registro de Usuarios

El registro de usuarios se realiza en dos factores, se realiza una separación de datos sensibles, como lo son usuario y contraseña, estos datos se envían al bloque en los cuales se realiza un proceso de encriptación del usuario y la contraseña, se le genera una llave privada única por cada usuario, después de este proceso se realiza un inserción a la base de datos con los datos no sensibles del usuario teléfono, nombre, apellidos, esto genera una respuesta con el id del usuario, que se anexa a los datos sensibles del usuario, luego de esto se genera un hash completo de estos 4 datos (Usuario,Contraseña,llave privada, id de usuario en base de datos) , se anexa al objeto que luego será insertado en un el bloque.

En la figura(3-5) se describe el flujo del proceso realizado por la aplicación para registrar un usuario:

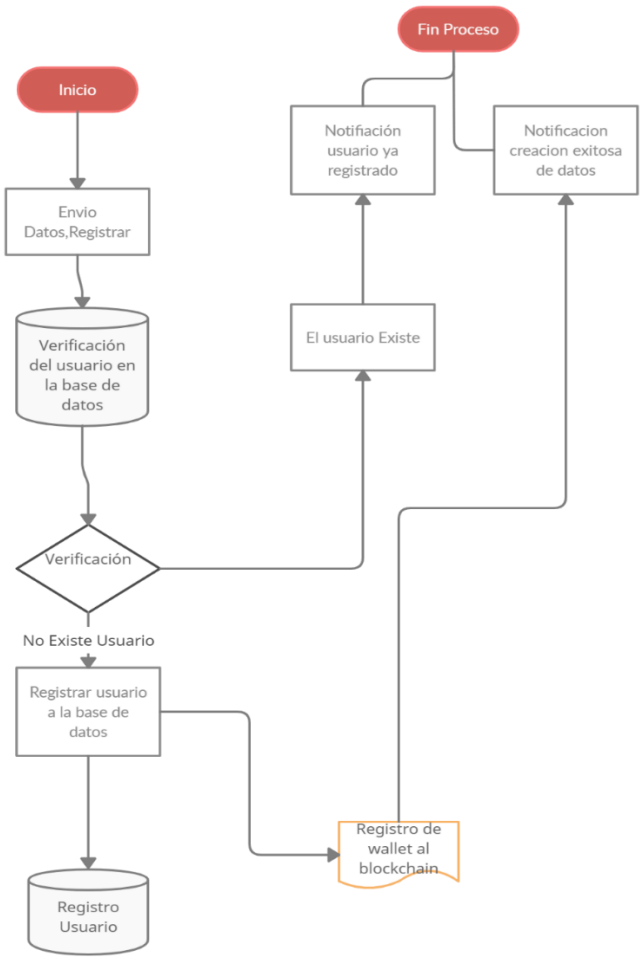


Figura 3-5. Registro de un usuario

Realizado por: Carrillo Carina, 2020

5.2.4. Validación del Usuarios

Se inicia realizando una búsqueda en toda la cadena de bloques buscando una coincidencia con el nombre de usuario (Correo electrónico), luego de esto se verifica que la contraseña coincida con la del usuario encontrado, después de este paso se verifica que el hash no haya sido modificado, realizando un proceso de hash con los datos ya nombrados, si todo el proceso es correcto se procede a generar una llave publica de identificación del usuario, para la validación de los accesos a rutas mediante este proceso, si en el proceso de validación un usuario ya se encuentra registrado, el sistema informara de error para asi no permitir duplicidad de datos.

En la figura(4-5) se describe el flujo del proceso realizado por la aplicación para registrar un usuario:

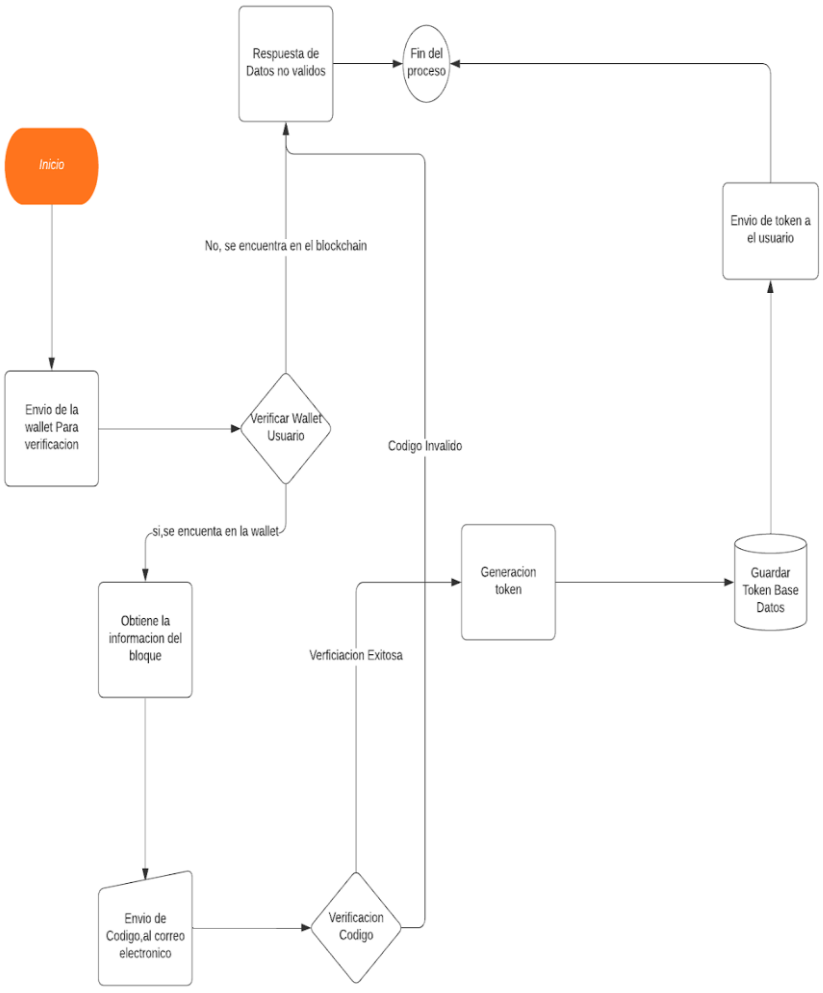


Figura 4-5. Validación de un usuario

Realizado por: Carrillo Carina, 2020

5.2.5. Acceso a un recurso

Para complementar la aplicación propuesta, se ha desarrollado un módulo que permita el ingreso a recursos, en este caso nos enfocamos a cuentas bancarias.

En la figura(5-5) se describe el flujo del proceso para solictar un recurso:

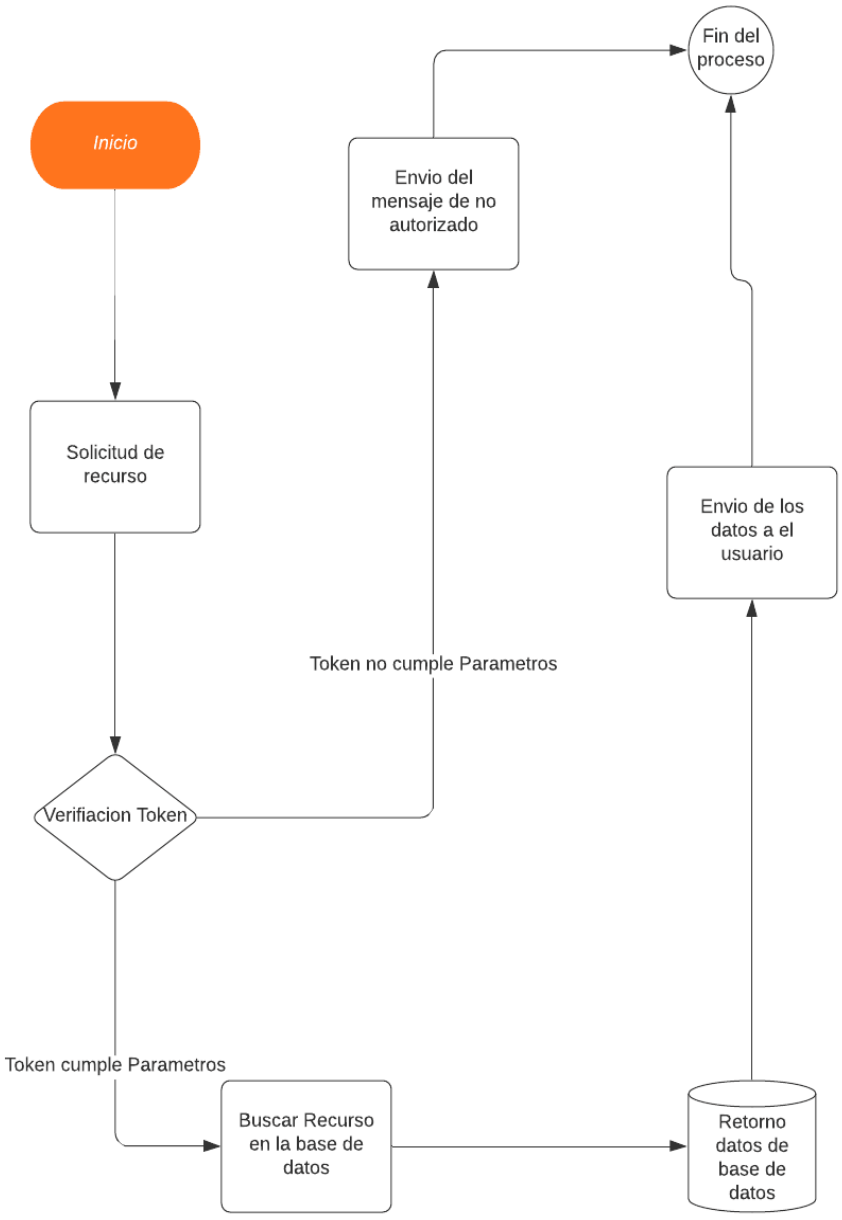


Figura 5-5. Solicitud de un recurso

Realizado por: Carrillo Carina, 2020

5.3. Implementación

Se implementa un prototipo funcional, publicado en un servidor con un dominio personalizado permitiendo el acceso a los usuarios mediante la web, este proyecto fue realizado bajo un contraro inteligente codificado en cuanto a la necesidad del proyecto, con posibilidad de ampliación de sus reglas de negocio en TypeScript un superconjunto de JavaScript usado para crear el servidor de el aplicativo.

Se creo un cliente en angular en cual le permite al cliente por medio una interfaz grafica crear una cuenta y mantener la privacidad de sus datos sensibles no expuestos, el cual le permite gestionar una identidad con su numero de cuenta bancaria, los cuales pueden ser verificados para establecer que no se le han realizado cambios en los datos, generado una alertar positiva o negativa del resultado en encontrado.

El cliente se conecta a través de la url de servidor, realizando una cambio de información con el servidor por medio de JSON, también cuenta con el envió de un jsonwebtoken usado como llave publica para identificar si el usuario cuenta con los permisos suficientes para acceder a un determinado recurso, esto para contrarrestar los posibles intentos de robo de información por parte de un externo.

5.4. Funcionamiento del Sistema Propuesto

El proceso de autenticación desarrollado presenta tres fases bien diferenciadas: el registro, la validación y el acceso.

Los siguientes pasos describen cómo funciona el sistema propuesto:

1. El usuario inicia sesión en la aplicación / sitio web con la combinación de nombre de usuario y contraseña.
2. La aplicación / sitio web solicita al usuario que envíe una transacción a su contrato 2FA. El usuario debe tener al menos un wallet en la Blockchain utilizada para poder enviar contratos y cifrar / descifrar cualquier mensaje. Se establece un tiempo de espera razonable para un evento autenticado, luego se rechaza el inicio de sesión si no se escucha ningún evento autenticado para la dirección de este usuario o si se agota el tiempo de espera.
3. El usuario envía una transacción al contrato 2FA.

4. Ethereum verifica la validez de la solicitud del usuario mediante la verificación de la integridad de su transacción con el sitio web / aplicación.
5. Si la solicitud es válida, el contrato 2FA genera OTP.
6. El contrato 2FA cifra la OTP mediante la clave pública del usuario y luego se la envía al usuario.
7. El contrato 2FA calcula el valor hash de OTP y lo envía al sitio web / aplicación (H1).
8. El usuario descifra la OTP y calcula su hash (H2), y envía el valor hash al sitio web / aplicación.
9. El sitio web / aplicación compara los valores hash recibidos (contrato inteligente del formulario H1 y usuario del formulario H2) para garantizar la integridad del usuario y que no haya ninguna alteración en la OTP. Si los dos valores son iguales y el proceso aún se encuentra dentro del tiempo de espera, el sitio web / aplicación proporciona el acceso al usuario.

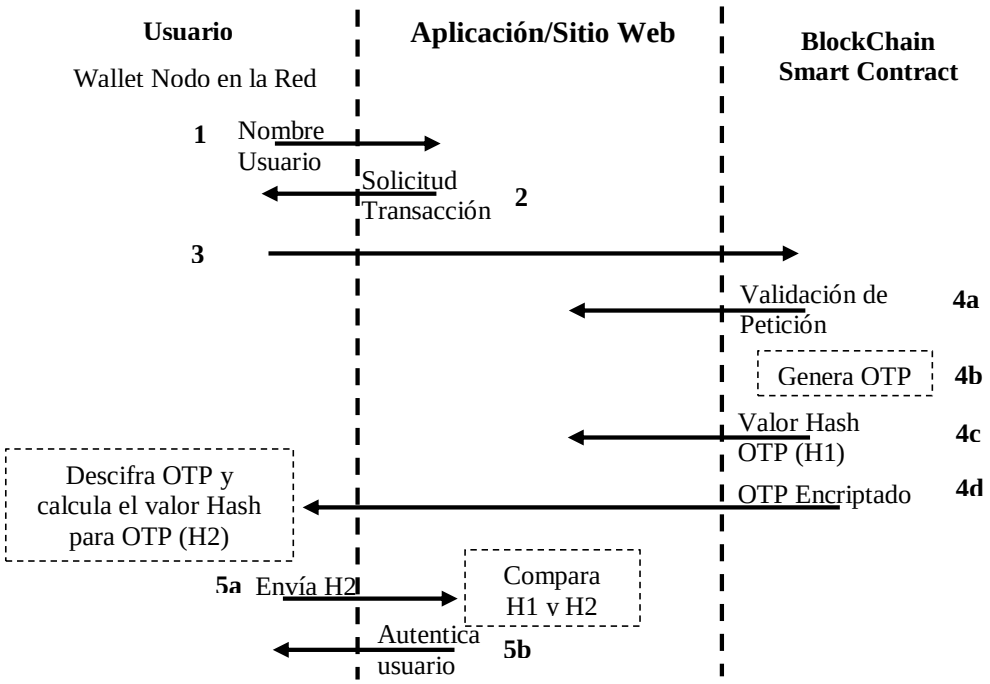


Figura 6-5. Funcionamiento del Sistema Propuesto

Realizado por: Carrillo Carina, 2020

CONCLUSIONES

- El estudio de blockchain llevado a cabo para el desarrollo de este proyecto no solo ha conducido a una solución que aprovecha cada una de las ventajas que proporciona esta tecnología, sino que ha permitido ver muchas de las posibilidades de futuro que tiene. Aunque cada día salen a la luz nuevas noticias sobre blockchain, y las criptomonedas, es aún una tecnología con mucho potencial por explotar, y que podría conllevar una revolución informática.

- El uso de la tecnología Blockchain y los contratos inteligentes permiten agregar una capa adicional de seguridad a una aplicación web para mitigar las vulnerabilidades en el proceso de inicio de sesión, y por ende brinda mayor protección a la información y activos de individuos y organizaciones. Cabe señalar que no existen entornos libres de vulnerabilidades. Por tanto, es imperativo la creación de políticas de seguridad que deben ser implementadas y monitoreadas constantemente.

- Al implementar el sistema de autenticación basado en la tecnología blockchain y el contrato inteligente, los ataques comunes como XSS, inyección SQL fallan porque no hay un servidor central donde se almacena la información de inicio de sesión, para lo cual se usó como herramientas de pentesting Vooki y Owasp Zap, obteniendo así una reducción del 100% en vulnerabilidades Altas y un 80% en las vulnerabilidades medias y bajas, obteniéndose un resultado de 83.33% de reducción total de vulnerabilidades frente al método tradicional de autenticación.

- Finalmente, la cadena de bloques es el núcleo del sistema que actúa como el backend del sistema de autenticación, asegurando toda la transacción entre los usuarios y la aplicación, permitiendo la validación y verificación de recursos, se crea el bloque de transacciones que se agrega a la cadena de bloques después de que la red alcanza el consenso. Tan pronto como se agrega el bloque a la blockchain, este cambio se vuelve permanente para la red y se difunde y se propaga a otros nodos. Este cambio no se puede deshacer ni duplicar. Por lo tanto, se asegura que los usuarios son legítimos y autorizados.

RECOMENDACIONES

- Los sistemas de autenticación actuales están constantemente mejorando, los esfuerzos de los programadores apuntan a incrementar la seguridad, pero un elemento a considerar en relación con seguridad de la información es el factor humano. Incluso con políticas estrictas de seguridad, el factor humano podría ser un punto menos seguro en un sistema de seguridad de la información, como humanos pueden verse afectado por técnicas de piratería social. En estos casos, personas con intenciones maliciosas intentan obtener acceso a información restringida en espacios físicos sin el debido permiso. Por tanto, es imperativo que los responsables de seguridad informática informen constantemente a los empleados de las organizaciones a través de la formación y la comunicación, dilucidando las razones para utilizar estos factores de autenticación adicionales.

- Al ser Blockchain una tecnología emergente aún no se han desarrollado marcos de referencias o estándares que normalicen el diseño de aplicaciones descentralizadas, por lo que se recomienda el asesoramiento de expertos al momento de implementarlo como una solución empresarial.

- No todos los escenarios pueden ser descentralizados por lo que se recomienda analizar el uso y los requistos que se desean implementar en el desarrollo, pero si se requiere el desarrollo de aplicaciones descentralizadas, Blockchain es la clave para garantizar la fiabilidad de la información sin necesidad de la intervención de terceros.

- Si se desea incrementar la seguridad en el acceso a otros aplicativos, se puede emplear el método de autenticación basado en tecnología blockchain y el contrato inteligente por los parámetros implementados como la inmutabilidad, el uso de hash criptográficos, la generación de tokens, las firmas digitales y la descentralización.

BIBLIOGRAFÍA

Ávila, J., & Toquica, L. (2018). Obtenido de Repositorio Institucional Universidad Distrital Francisco José de Caldas: http://hdl.handle.net/11349/14652

Balmaseda-Aranda, F. J. (2018). *Aseguramiento de Dispositivos IoT con Blockchain e Infraestructura de Clave Pública*. España: Re-UNIR.

Bashir, I. (2017). *Mastering Blockchain.* Birmingham: Packt Publishing Ltd.

Bitcoin. (2020). *bitcoindeveloper*. Obtenido de https://developer.bitcoin.org/reference/block_chain.html?highlight=header%20block

D. Johnson, A. M. (2001). The elliptic curve digital signature algorithm (ecdsa). *International Journal of Information Security*, 36-63.

Dolader, C. B. (2017). La blockchain : fundamentos, aplicaciones y relación con otras tecnologías disruptivas. *Nuevas tecnologías digitales*, págs. 33-40.

Fan Zhang, E. C. (2016). CCS '16: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. (págs. 270–282). ACM Digital Library.

Fernández, X. (2018). *Gestión de identidades descentralizadas con Blockchain.* Catalunya.

Google Inc. (2019). *Angular*. Obtenido de https://angular.io/

Maesa, D. D., Mori, P., & Ricci, L. (2018). Blockchain Based Access Control Services. *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)* (págs. 1379-1386). IEEE.

Maida, E. P. (2015). *Metodologías de desarrollo de software*. Obtenido de https://repositorio.uca.edu.ar/handle/123456789/522

Marshall, B. (2018). *Medium*. Obtenido de https://medium.com/working-lab-capital/schnorr-signatures-new-tokens-on-coinbase-and-blackrock-3c0aaf5cc502

Matas, A. (2018). *Diseño del formato de escalas tipo Likert: un estado de la cuestión.* Obtenido de http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1607-40412018000100038&lng=es&nrm=iso>. ISSN 1607-4041

Microsoft. (2019). *Visual Studio Code*. Obtenido de https://code.visualstudio.com/

Miethereum. (junio de 2020). *Miethereum*. Obtenido de https://www.miethereum.com/blockchain/

Miranda, V. (2018). *Explorando la Blockchain de Ethereum y el desarrollo de smart contracts.* Obtenido de http://hdl.handle.net/2117/127784

Nakamoto, S. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System.* Obtenido de https://bitcoin.org/bitcoin.pdf

OpenJS Foundation. (2019). Obtenido de https://nodejs.org/es

Pagán, A., & María, I. (Junio de 2019). *Sistema de autenticación robusto.* Obtenido de Repositorio institucional de la Universidad de Alicante: http://rua.ua.es/dspace/handle/10045/93270

Panicker, S., Patil, V., & Kulkarni, D. (2016). An Overview of Blockchain Architecture and it's Applications. *International Journal of Innovative Research in Science, Engineering and Technology*.

Park, W.-S., Hwang, D.-Y., & Kim, K.-H. (2018). A TOTP-Based Two Factor Authentication Scheme for Hyperledger Fabric Blockchain. *2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*, (págs. 817-819). Prague.

Postman, Inc. (2019). *Postman*. Obtenido de https://www.postman.com/

Rodriguez, N. (2019). *101blockchains*. Obtenido de https://101blockchains.com/es/como-funciona-blockchain/

Salesforce. (2019). *Heroku*. Obtenido de https://www.heroku.com/home

Sanchez Mora, D. C., & Jerez Vargas, F. G. (Marzo de 2019). *Repositorio Institucional Universidad Distrital - RIUD.* Obtenido de Sistema descentralizado para la verificación y autenticación de certificados académicos utilizando la tecnología Blockchain: http://hdl.handle.net/11349/22404

Sanchez, A. (2020). *Comprendiendo JWT y como implementar un JWT simple con Flask*. Obtenido de https://content.breatheco.de/es/lesson/what-is-JWT-and-how-to-implement-with-Flask

Savita Mohurle, M. P. (2017). A brief study of Wannacry Threat: Ransomware Attack 2017. *International Journal of Advanced Research in Computer Science*, 8.

Truffle. (2019). *Truffle Suite*. Recuperado el 2019, de https://www.trufflesuite.com/

Universidad Carlos III de Madrid-Departamento de Estadistica. (2019). Obtenido de http://www.est.uc3m.es/esp/nueva_docencia/getafe/ciencias_estadisticas/TecnicasInferenciaEstadistica/tablachicuadrado.pdf

Zikratov, I., Kuzmin, A., Akimenko, V., Niculichev, V., & Yalansky, L. (2017). Ensuring data integrity using blockchain technology. *2017 20th Conference of Open Innovations Association (FRUCT)*, (págs. 534-539). St. Petersburg.

ANEXOS

ANEXO A. Contrato en Solidy para el registro

En este anexo se detalla el contracto que se diseñó para el registro en la blockchain.

```
pragma solidity >=0.4.25 <0.6.0;

contract RegistryContract {
   struct Identity {
   bool isValue; // true when payload is stored
   bool isRevoked; // true when it is revoked
         }
  address public owner;
  // key: address, value: Identity struct
  mapping (address => Identity) public entities;

  event NewEntityAdded(address sender);
  event EntityRevoked(address sender);

  modifier entityMustExist(address entity) {
   require(entities[entity].isValue, "entity must exist");
   _;
  }

  modifier entityMustNotExist(address entity) {
   require(!entities[entity].isValue, "entity must not exist");
   _;
  }

  modifier entityMustNotRevoked(address entity) {
   require(!entities[entity].isRevoked, "entity must not revoked");
   _;  }
```

```
  constructor() public {
                owner = msg.sender;
        }

  function storeNewEntity() public
  entityMustNotExist(msg.sender) {
    Identity storage i = entities[msg.sender];
    i.isValue = true;
    i.isRevoked = false;

    emit NewEntityAdded(msg.sender);
  }

  function revokeEntity() public
  entityMustExist(msg.sender)
  entityMustNotRevoked(msg.sender) {
    entities[msg.sender].isRevoked = true;

    emit EntityRevoked(msg.sender);
  }

  function isEntityRevoked(address entity) public view
  returns (bool) {
    return (entities[entity].isRevoked == true);
  }

  function isEntityExist(address entity) public view
  returns (bool) {
    return (entities[entity].isValue == true);
  }
}
```

ANEXO B. Registro de una dirección blockchain y el proceso de firma a partir de un código (archivo eth_tools.js).

```
const path = require('path');
const prompts = require('prompts');
const chalk = require('chalk');
const EthCrypto = require('eth-crypto');
const fs = require('fs');
const Web3 = require('web3');
const web3 = new Web3(new Web3.providers.HttpProvider('http://localhost:8545'));

const contractABIPath = path.normalize('./build/contracts/RegistryContract.json');
const contractPath = path.normalize('./client/contract.json');
const keyPath = path.normalize('./client/user.json');

const questions = [{
  type: 'select',
  name: 'value',
  message: 'Escoge una opción?',
  choices: [{
    title: 'Registrar una dirección',
    value: 1
  }, {
    title: 'Firmar un código',
    value: 2
  }]
}, {
  type: prev => prev == 2 ? 'text' : null,
  name: 'message',
  message: 'Cuál es el código que deseas firmar?',
  validate: message => message == '' ? 'Ingresa tu código' : true
}];

(async () => {
  const response = await prompts(questions);
```

```
// registering address
if (response["value"] == 1) {

  let rawContractInfo = fs.readFileSync(contractPath);
  let info = JSON.parse(rawContractInfo);
  let rawContractABI = fs.readFileSync(contractABIPath);
  let abi = JSON.parse(rawContractABI);

  const RC = new web3.eth.Contract(abi.abi, info.address);

  let rawUser = fs.readFileSync(keyPath);
  let user = JSON.parse(rawUser);
  let userAddress = web3.utils.toChecksumAddress(user.address);

  let tx = await RC.methods.storeNewEntity().send({
    from: userAddress,
    gas: 1000000
  });

  if (typeof tx.events.NewEntityAdded !== 'undefined') {
    const event = tx.events.NewEntityAdded;
    console.log('We store this address in the blockchain');
    console.log(chalk.black.bgYellow(event.returnValues['sender']));
  } else {
    console.log(chalk.red('ERROR! No se puede almacenar la entidad en la blockchain!'));
  }

  // signing messages
} else {
  if (fs.existsSync(keyPath)) {
    let rawdata = fs.readFileSync(keyPath);
    let user = JSON.parse(rawdata);
    let message = response["message"];

    let messageHash = EthCrypto.hash.keccak256(message);
    let signature = EthCrypto.sign(user.privateKey, messageHash);
```

```
      console.log('Here is the signature of your message:');
      console.log(chalk.black.bgYellow(signature));
    } else {
      console.log(chalk.red('ERROR! Aún no ha creado una direccción'));
    }
  }
})();
```

ANEXO C. Generación del Token (archivo jwt.js)

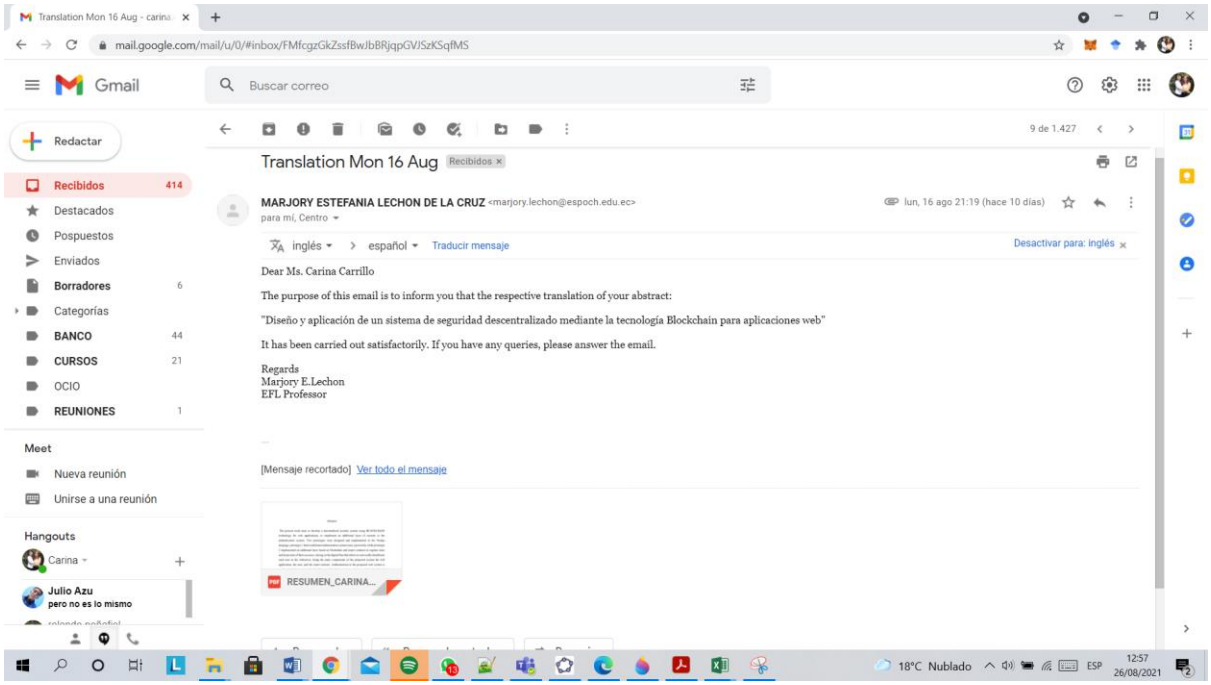
```
"use strict";
Object.defineProperty(exports, "__esModule", { value: true });
exports.JsonWebtoken = void 0;
const jwt = require("jsonwebtoken");
class JsonWebtoken {
    constructor() { }
    sign(data) {
        let secretKey = "Palabra Secreta";
        return jwt.sign({
            user: data,
        }, secretKey, {
            expiresIn: 24 * 60 * 60,
        });
    }
    verify(token) {
        let secretKey = process.env.SECRET;
        try {
            return jwt.verify(token, secretKey);
        }
        catch (error) {
            return "Token_not_valid";
            console.log(error);
        }
    }
}
exports.JsonWebtoken = JsonWebtoken;
//# sourceMappingURL=jwt.js.map
```

ANEXO D. Validación del Token (archivo validatetoken.js)

```
"use strict";
Object.defineProperty(exports, "__esModule", { value: true });
exports.tokenValidate = void 0;
const jwt = require("jsonwebtoken");
const tokenValidate = (req, res, next) => {
    const token = req.headers.authorization;
    if (!token)
        return res.status(401).send({ message: "No Autorizado" });
    const Btoken = token.replace("Bearer ", "");
    jwt.verify(Btoken, process.env.SECRET || "Palabra Secreta", function (err, decoded) {
        if (err)
            return res.status(401).send({ message: "No Autorizado" });
        if (decoded) {
            console.log(decoded.user.iduse);
            next();
        }
        else {
            return res.status(401).send({ message: "No Autorizado" });
        }
    });
};
exports.tokenValidate = tokenValidate;
//# sourceMappingURL=validatetoken.js.map
```

ANEXO E. Creación de la cadena de bloques (archivo hashblock.js)

```
"use strict";
Object.defineProperty(exports, "__esModule", { value: true });
exports.hashBlock = exports.HashBlock = void 0;
const crypto_js_1 = require("crypto-js");
class HashBlock {
    constructor() { }
    hashUser(data) {
        return crypto_js_1.SHA256(JSON.stringify(data)).toString();
    }
    validarHash(datobloque, datouser) {
        if (this.hashUser(datobloque) != this.hashUser(datouser)) {
            return false;
        }
        else {
            return true;
        }
    }
}
exports.HashBlock = HashBlock;
exports.hashBlock = new HashBlock();
//# sourceMappingURL=hashblock.js.map
```


Translation Mon 16 Aug - carina
mail.google.com/mail/u/0/#inbox/FMfcgzGkZssfBwJbBRjqpGVJSzKSqfMS
Gmail
Buscar correo
Redactar
Recibidos 414
Destacados
Pospuestos
Enviados
Borradores 6
Categorías
BANCO 44
CURSOS 21
OCIO
REUNIONES 1
Meet
Nueva reunión
Unirse a una reunión
Hangouts
Carina
Julio Azu
pero no es lo mismo
9 de 1.427
Translation Mon 16 Aug
Recibidos
MARJORY ESTEFANIA LECHON DE LA CRUZ <marjory.lechon@espoch.edu.ec>
para mi, Centro
lun, 16 ago 21:19 (hace 10 días)
inglés
español
Traducir mensaje
Desactivar para: inglés
Dear Ms. Carina Carrillo
The purpose of this email is to inform you that the respective translation of your abstract:
"Diseño y aplicación de un sistema de seguridad descentralizado mediante la tecnología Blockchain para aplicaciones web"
It has been carried out satisfactorily. If you have any queries, please answer the email.
Regards
Marjory E.Lechon
EFL Professor
[Mensaje recortado] Ver todo el mensaje
RESUMEN_CARINA...
18°C Nublado
ESP
12:57
26/08/2021

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

DIRECCIÓN DE BIBLIOTECAS Y RECURSOS DEL APRENDIZAJE

UNIDAD DE PROCESOS TÉCNICOS Y ANÁLISIS BIBLIOGRÁFICO Y DOCUMENTAL

REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: *20 / 08 / 2021*

INFORMACIÓN DEL AUTOR/A (S)
Nombres – Apellidos: *Carina Araceli Carrillo Villalva*
INFORMACIÓN INSTITUCIONAL
Instituto de Posgrado y Educación Continua
Título a optar: *Magister en Seguridad Telemática*
f. Analista de Biblioteca responsable*:* *Lic. Luis Caminos Vargas Mgs.*

LUIS ALBERTO CAMINOS VARGAS

Firmado digitalmente por LUIS ALBERTO CAMINOS VARGAS
Nombre de reconocimiento (DN): c=EC, l=RIOBAMBA, serialNumber=0602766974, cn=LUIS ALBERTO CAMINOS VARGAS
Fecha: 2021.08.20 08:37:27 -05'00'

0087-DBRAI-UPT-IPEC-2021