

UNIVERSIDAD AUTÓNOMA DE OCCIDENTE

*Informe de laboratorio proyecto final Telecomunicaciones 3*

*proyecto final*

*Jose Manuel Velasquez Cabello*

[*jose.velasquez@uao.edu.co*](mailto:jose.velasquez@uao.edu.co)

*Christian Tenorio Velasco*

[*christian.tenorio@uao.edu.co*](mailto:christian.tenorio@uao.edu.co)

*Juan Esteban Agredo*

[*juan\_esteban.agredo@uao.edu.co*](mailto:juan_esteban.agredo@uao.edu.co)

*Andres Luna*

[*andrns.luna@uao.edu.co*](mailto:andrns.luna@uao.edu.co)

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

1. **Introducción:**

Este Proyecto final tiene como objetivo implementar una red peer-to-peer (P2P) basada en blockchain para gestionar operaciones durante desastres naturales. Este proyecto busca proveer la infraestructura necesaria para facilitar la gestión de desastres entre diferentes partes, como el gobierno, organismos de socorro y ciudadanos. La red P2P permitirá un flujo eficiente y transparente de información y ayudas, asegurando que todas las partes involucradas puedan coordinarse y responder de manera efectiva ante emergencias. Utilizando blockchain, se garantiza la integridad y veracidad de los datos compartidos, mejorando la capacidad de respuesta y recuperación durante y después de un desastre.

1. **Contexto:**

El uso de tecnologías avanzadas es crucial en la gestión de operaciones durante desastres naturales, donde la comunicación eficiente entre el gobierno, organismos de socorro y ciudadanos es esencial. Las redes tradicionales pueden fallar debido a daños en la infraestructura, lo que dificulta la coordinación y distribución de recursos.

El Peer-to-Peer (P2P) es una red descentralizada en la que dispositivos (nodos o pares) se conectan directamente entre sí para compartir recursos sin depender de un servidor central. En el contexto de blockchain, el P2P permite la transferencia de información entre usuarios sin intermediarios, utilizando un registro compartido y verificado por todos los participantes, lo que refuerza la seguridad al detectar y rechazar actividades fraudulentas.

Este proyecto implementa una red P2P basada en blockchain para facilitar la gestión de operaciones durante un desastre natural. La red permitirá compartir información de manera segura y transparente, garantizando la veracidad y disponibilidad de los datos críticos antes, durante y después del desastre. Con blockchain, se puede construir una fuente común de información confiable para coordinar una respuesta efectiva, facilitando el flujo de información y ayudas de manera eficiente y transparente, beneficiando a autoridades, personas y comunidades.

1. **Alternativas de Solución:**

Como posibles soluciones a la problemática planteada en la red per to per basada en blockchain obtuvimos 3 posibles soluciones las cuales fueron: realizar un seguimiento de kits de emergencia a través de blockchain el cual se basaba en realizar una trazabilidad certera conocer la ubicacion especifica de los kits de emergencia entregados a las familias

enviar mensajes a un dispositivo móvil con mensajes encriptados que evitan las corruptibilidad de los mismo para dar un aviso real a una persona y saber si esta bien o en que estado de salud se encuentra a la hora de una catástrofe natural ya sea un terremoto o una inundación

por último y la solución escogida fue la implementación de una infraestructura certera que permitiera la realización de transacciones economicas inmodificables entre los entes involucrados los cuales eran gobierno, organismos de socorro y ciudadano para dee sta manera poder llevar un control certero y eficaz en el manejo de los dineros destinados a cada ente ya que con blockchain per to per no es posible modificar los bloques algoritmos de los archivos una vez realizadas las transacciones o la creación del mismo.

1. **Diseño solución**

Al momento de diseñar una red peer-to-peer basada en blockchain, utilizamos Ethereum y el cliente Geth para configurar tres nodos. Esta red permitirá transferencias de dinero y será la base para futuras expansiones, incluyendo la coordinación de envíos y otros aspectos cruciales en la gestión de desastres naturales.

Pasos de Implementación:

1. **Instalación de Geth en cada nodo:**

* Descargar e instalar Geth en tres dispositivos diferentes.
* Configurar cada nodo para que se conecte a la red Ethereum usando el algoritmo de consenso Proof of Work (PoW).

1. **Configuración de la Red P2P:**

* Iniciar Geth en cada nodo con los comandos necesarios para conectarse y sincronizarse con la red.
* Configurar los nodos para que se reconozcan y se comuniquen directamente, formando una red descentralizada.

1. **Transferencias de Dinero:**

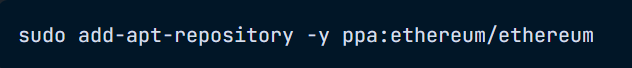
* Crear y gestionar cuentas en cada nodo para realizar transacciones.
* Usar comandos de Geth para transferir fondos entre las cuentas, asegurando que todas las transacciones sean registradas y verificadas en la blockchain.

1. **Gestión de Desastres:**

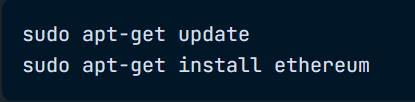
* Esta red P2P basada en blockchain no solo facilitará la transferencia de dinero de manera segura y eficiente, sino que también ofrecerá una infraestructura robusta para gestionar operaciones durante desastres naturales, garantizando transparencia, veracidad de la información y coordinación efectiva.

1. **Implementación:**

para la implementación de este proyecto nos basamos en la linea de comandos proporcionada por la pagina oficial de ethereum donde nos muestran como instalar, y arrancar nuestro servicio

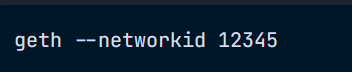


accedemos al repositorio de ethereum para acceder a los datos de instalacion de Go.

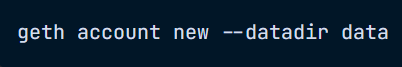


Actualizamos e instalamos ethereum del repositorio.

Una vez ya tenemos instalado ethereum, procedemos al montaje de nuestra red descentralizada.



Primero asignamos un “networkid” a nuestra aplicacion “geth”.



Creamos una cuenta y contraseña a nuestro nodo, en este caso llamado “data”.



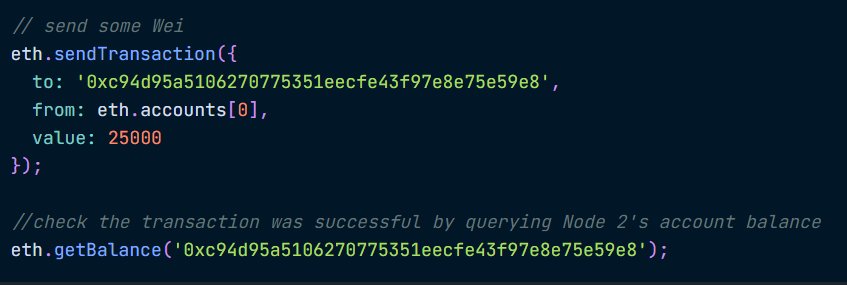
Ahora creamos un archivo “genesis.json”, en el cual entre otras cosas esenciales de geth, tambien se configura la información inicial de nuestro nodo, en este caso estamos agregando 2 nodos a los cuales les estamos ingresando un valor inicial de 300000 y 400000.



Enlazamos la informacion del genesis.json con el nodo llamado “data”.



arrancamos nuestro servicio p2p de nuestro nodo llamado “data”



Por ultimo, conectandonos al nodo podemos realizar transacciones entre nodos o incluso consultar el balance de uno de los nodos.

**Resultados**

El resultado de implementar Blockchain con Ethereum en un entorno Ubuntu para transacciones puede ser muy efectivo para diversas aplicaciones. Aquí hay algunas consideraciones y posibles resultados:

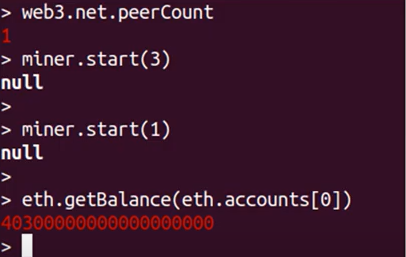
Seguridad mejorada:La implementación de Ethereum en una red blockchain en Ubuntu puede proporcionar un entorno seguro para las transacciones, protegiendo los datos y los activos digitales contra ataques maliciosos.

Transparencia y confiabilidad: La naturaleza transparente de la tecnología blockchain permite que todas las transacciones sean visibles y verificables por todas las partes involucradas. Esto puede aumentar la confianza entre los participantes y reducir la necesidad de intermediarios.

Eficiencia en las transacciones: Ethereum y la tecnología blockchain en general pueden mejorar la eficiencia en las transacciones al eliminar intermediarios y simplificar los procesos. Las transacciones pueden ser ejecutadas de manera más rápida y con costos reducidos en comparación con los sistemas tradicionales.

la implementación de Blockchain con Ethereum en un entorno Ubuntu para transacciones puede conducir a un aumento en la seguridad, la eficiencia y la confiabilidad de los sistemas de transacciones, así como abrir nuevas posibilidades para la innovación en el desarrollo de aplicaciones descentralizadas.

en la siguientes imágenes observamos los procedimientos pertenecientes a una transacción que se realizada con la blockchain de ethereum la cual nos permitirá evidenciar el incremento de una cantidad de monedas definida anteriormente en el archivo genesis.Json de nuestro proyecto



una vez se ha establecido la conectividad entre el el génesis y la cuenta creada podemos ver que con el comando eth.getBalance obtenemos un valor de 4030000000000000000000 eteres que es la base de dinero asignada en el archivo genesis.Json

para hacer esta transacción se requiere del siguiente comando

**“eth.sendTransaction({from: eth.accounts[0], to: "0xugkjbjbyh333467267397426487093f965", value: web3.toWei(1, "ether")})”**

Esta línea de código sería una transacción válida en Ethereum utilizando la consola de JavaScript de la biblioteca web3.js.

Ahora, desglosemos lo que está sucediendo aquí:

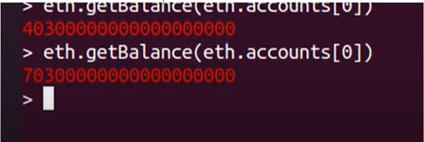
eth.sendTransaction: Este es un método de la API de Ethereum que se utiliza para enviar una transacción desde una cuenta a otra en la red Ethereum.

{from: eth.accounts[0]}: Aquí especificamos la cuenta de la cual se enviarán los fondos. eth.accounts[0] se refiere a la primera cuenta en la lista de cuentas disponibles en el nodo Ethereum que estás utilizando.

to: "0xugkjbjbyh333467267397426487093f965": Indica la dirección del destinatario de la transacción. Esta dirección debe estar en el formato correcto de una dirección Ethereum (una cadena de 40 caracteres que comienza con "0x").

value: web3.toWei(1, "ether"): Aquí especificamos la cantidad de Ether que queremos enviar en la transacción. web3.toWei(1, "ether") convierte 1 Ether en su equivalente en wei, la unidad más pequeña de Ether en Ethereum. En este caso, 1 Ether equivale a 1e18 wei.

Por lo tanto, en resumen, la línea de código envía 3 Ether desde la primera cuenta disponible en el nodo Ethereum a la dirección especificada.



Observamos que posterior a la ejecucion de la linea de comando anterio podemos observar un incremento de el eth.getBalance el cual se inicio con 403000000000000000000 y posteriormente se incremento a 7030000000000000000

1. **Conclusiones:**

Las redes peer-to-peer (P2P) en combinación con la tecnología blockchain pueden ser extremadamente útiles para la gestión de procesos durante desastres naturales. Su arquitectura descentralizada garantiza la disponibilidad y la integridad de los datos, permitiendo una rápida recuperación y continuidad de las operaciones críticas sin depender de un servidor central, lo que es esencial en situaciones de emergencia.

La implementación de la red blockchain no se pudo completar satisfactoriamente debido a problemas en la transferencia de datos entre nodos. La plataforma no funcionó correctamente con nuestra base de datos existente, lo que impidió realizar transacciones de manera efectiva y aseguró la integridad de la red.

1. **REFERENCIAS**

[1]Cryptopolitan, "Peer-to-Peer en Blockchain: Cómo Funciona," [En línea]. Disponible: https://www.cryptopolitan.com/es/peer-to-peer-en-blockchain-como-funciona/. [Accedido: 15-May-2024].

[2]EBSCOhost, "Understanding Blockchain Technology," [En línea]. Disponible: https://edsp.proxyuao.elogim.com/eds/pdfviewer/pdfviewer?vid=10&sid=db36b8c7-3273-42cc-8077-83e951889344%40redis. [Accedido: 15-May-2024].

[3]Blockgeeks, "Ethereum: Getting Started with Geth," YouTube, 22-Nov-2021. [Vídeo]. Disponible en: https://www.youtube.com/watch?v=k5LlT7PBOyk. [Accedido: 15-May-2024].

[4]E. Durán, "Geth Ethereum: Primeros Pasos," Medium, 20-Mar-2020. [En línea]. Disponible: https://medium.com/@edumar111/geth-ethereum-primeros-pasos-d71186cf9a70. [Accedido: 15-May-2024].

[5]Ivan on Tech, "How to Set Up a Private Ethereum Network," YouTube, 15-Apr-2022. [Vídeo]. Disponible en: https://youtu.be/KUdLT2sF6wY?si=46oaJljGnT-we6-s. [Accedido: 15-May-2024].

[6]Ethereum Foundation, "Installing Geth," [En línea]. Disponible: https://geth.ethereum.org/docs/getting-started/installing-geth. [Accedido: 15-May-2024].