ISSN 1870-4069

Trazabilidad de imágenes digitales usando Blockchain

José Antonio Jiménez Miramontes[1], Rocío Aldeco-Pérez[2]

[1] Universidad Nacional Autonoma de México,
IIMAS, Posgrado en Ciencias e Ingeniería de la Computación,
México

[2] Universidad Nacional Autonoma de México,
Facultad de Ingeniería,
México

ja.jimenez.mi@gmail.com
raldeco@unam.mx

Resumen. El control que la gente tiene sobre sus activos digitales una vez que están en Internet es debatible. Si nos enfocamos en imágenes que se comparten en Internet vemos casos en donde imágenes públicas son editadas sin el consentimiento del dueño o imágenes compartidas de manera confidencial que son compartidas a terceros sin el consentimiento del dueño generando daños personales o siendo parte del fenómeno de *fake news*. Una solución a esta problemática es el tener la historia de lo que ha sucedido a dicha imagen como evidencia de las operaciones realizadas y de ahí tomar acciones que pueden ir desde lo personal hasta lo legal. Esta historia o "linaje electrónico" de la imagen debe ser confiable, de ahí proponemos el uso de Blockchain. Sin embargo, las Blockchain públicas más usadas son lentas y consumen altas cantidades de energía, además de no ser eficientes cuando de almacenamiento de imágenes se trata. En vista de esto se propone el uso de una Blockchain llamada Solana junto con el protocolo IPFS para el almacenamiento de imágenes. Como consecuencia se obtendrá un esquema distribuido, eficiente y confiable que posteriormente puede ser usado para dar trazabilidad a diversos activos digitales.

Palabras clave: Blockchain, trazabilidad, cifrado simétrico.

Digital Image Traceability Using Blockchain

Abstract. The control that people have over their digital assets once they are on the Internet is debatable. When focusing on images shared online, we can observe cases where public images are edited without the owner's consent, or confidentially shared images are distributed to third parties without authorization, causing personal harm or contributing to the spread of fake news. A potential solution to this problem is to maintain a history of what has happened to an image as evidence of the

operations performed, enabling actions that may range from personal to legal measures. This history or "electronic lineage" of the image must be trustworthy; therefore, we propose the use of Blockchain. However, the most widely used public blockchains are slow, consume large amounts of energy, and are inefficient for image storage. In view of this, we propose the use of the Solana blockchain together with the IPFS protocol for image storage. As a result, a distributed, efficient, and reliable scheme can be achieved, which could later be applied to the traceability of various digital assets.

Keywords: Blockchain, raceability, symmetric encryption.

1. Introducción

El control que la gente tiene sobre sus activos digitales una vez que están en internet es debatible, estos activos pueden ser duplicados con facilidad o caer en manos de un tercero que no necesariamente es de confianza. Con esto surge la necesidad de controlar y saber qué sucede con dichos activos digitales.

Siendo las imágenes un ejemplo de activo digital, existen casos en donde imágenes públicas son editadas sin el consentimiento del dueño [16,19] o imágenes compartidas de manera confidencial son compartidas a terceros sin el consentimiento del dueño [22] generando daños personales o siendo parte del fenómeno de *fake news*.

Dada la naturaleza abierta del internet, borrar un activo digital se vuelve sumamente retador. Una solución alternativa es tener la historia de lo que ha sucedido con dicho activo como evidencia de las operaciones realizadas y de ahí tomar acciones que pueden ir desde lo personal hasta lo legal. Esta historia se denomina linaje electrónico. El "linaje electrónico" nos permite conocer el origen y los procesos por los que pasa un activo digital generando la propiedad de trazabilidad [14]. Este registro de procesos da la posibilidad al dueño del activo de saber qué es lo que ocurre con su propiedad. Para lograr esto, las aplicaciones deben generar documentación de los procesos que le están ocurriendo a los activos digitales y almacenarlos de forma segura para que no sean alterados y sean confiables.

La primeras propuestas para crear y almacenar el linaje electrónico involucran un sistema centralizado donde se encuentran todos los registros de procesos [14]. Esta situación obliga a tener un tercero de confianza sin garantía alguna de su comportamiento. Para resolver esta problemática se propone el uso esquemas distribuidos y descentralizados, una opción es Blockchain. Blockchain es una base de datos distribuida cuya tecnología engloba el uso de criptografía, algoritmos de consenso y modelos económicos. Blockchain combina redes punto a punto y algoritmos de consenso distribuidos para resolver problemas de sincronización que aparecen en bases de datos distribuidas tradicionales [9]. Sin embargo, las Blockchain públicas más usadas son lentas y consumen altas cantidades de electricidad, además de no ser eficientes en el almacenamiento de archivos [9]. Esto es consecuencia del tipo de protocolo de consenso usado.

Existen propuestas de almacenar el linaje electrónico usando Blockchain. Azaria et al. [5] proponen un esquema descentralizado para el manejo de registros médicos con el uso de Blockchain. Por otro lado, Sifah et al. [17], presentan el uso de Blockchain para el linaje electrónico archivos en la nube. Finalmente, Khatal et al. [10] proponen de igual forma un marco de referencia para el linaje electrónico de archivos con el uso de Blockchain y de IPFS.

La ventaja de estas propuestas es que ofrecen integridad del linaje electrónico gracias al uso de Blockchain pero tienen como desventaja el hecho de que el almacenamiento de los archivos es centralizado ya que el primero pretende que la información médica se mantenga en las bases de datos de los proveedores mientras que el segundo trabajo es específicamente diseñado para información almacenada en la nube, lo cual significa que un proveedor de servicio de la nube se encarga de administrar los archivos. La tercer propuesta sí maneja descentralización para el almacenamiento de los archivos pero únicamente se concentra en archivos de texto, mientras que este trabajo busca proporcionar una solución para imágenes.

Otro aspecto que vale la pena mencionar es que todas las propuestas anteriores utilizan la red de Ethereum, haciendo de estas soluciones lentas y costosas. El uso de otras redes que implementan protocolos de consenso más eficientes y con un menor costo en las transacciones genera una solución más factible de implementar.

En vista de esto se propone el uso de la Blockchain llamada Solana, basada en un protocolo de consenso eficiente y sustentable que se describe en el trabajo de Yakovenko [21]. Esta Blockchain almacenará el linaje electrónico de imágenes, es decir, las operaciones que se realizarán sobre ellas. Para el caso del almacenamiento de imágenes, se propone el uso de otro protocolo diseñado para este fin llamado IPFS [6].

Como consecuencia se obtendrá una solución distribuida, eficiente y confiable debido a las propiedades ofrecidas por estas tecnologías. Esto se formalizara en un marco de referencia que permitirá verificar su correcto funcionamiento, además de su uso en otros activos digitales que no sean necesariamente imágenes.

2. Linaje electrónico y blockchain

La palabra “linaje” puede definirse como la derivación desde un origen particular hasta un estado específico de un elemento. Esta idea puede aplicarse al mundo digital dando origen al linaje electrónico que no es más que el proceso que conduce a un dato [14]. El linaje electrónico apoya la información y la integridad del proceso documentando las entidades, sistemas y procesos que operan y contribuyen a los datos de interés, sirviendo como un registro histórico inalterable de la duración de los datos y sus orígenes [12]. Sus usos son diversos e incluyen el evaluar la calidad de la información [2], atribuir el origen de un resultado computacional, reproducir ejecuciones previas de aplicaciones o como evidencia en auditorías electrónicas como se menciona en [3]. Muchas de estas

aplicaciones se crearon en entornos centralizados asumiendo que existían las medidas necesarias para garantizar la integridad de esta información.

Por otro lado existe Blockchain que es una lista creciente de registros llamados bloques o transacciones, unidos entre sí a través del uso de funciones hash criptográficas. Una función hash se utiliza para mapear información digital de cualquier longitud a datos digitales de tamaño fijo. Los valores devueltos por este tipo de función se denominan digesto, valores resumen, o simplemente valores hash. Este digesto es único para un valor de entrada dado, así cualquier cambio en los datos de entrada cambia de manera significativa el dato de salida [1], siendo este una herramienta para verificar la integridad de datos. Mediante el uso de estas funciones, las redes punto a punto y los protocolos de consenso [7], Blockchain hace que la historia de activos digitales sea inalterable, inmutable y transparente [18]. Esto se puede ver en aplicaciones como criptomonedas, registros de propiedad, instrumentos financieros, servicios de identidad, cadenas de suministros, IoT, sistemas de votación, entre otros [11].

Existen tres tipos de blockchain: públicas, privadas y consorcio. En las blockchain públicas cualquiera en la red puede validar transacciones y formar parte del proceso para lograr consenso y así tener acceso al historial completo de las transacciones dentro de esta blockchain. Las transacciones además de contener la información de dicha transacción también contiene la firma digital de su creador. Esto quiere decir que cada participante deberá tener un par de llaves pública - privada por cada transacción. La llave privada es usada para firmar digitalmente la transacción y la llave pública para identificar al origen de dicha transacción [1]. Por otro lado, en las blockchains privadas, los nodos deben ser autenticados exitosamente para participar en el proceso de verificación y validación de transacciones y tener acceso a la información dentro del blockchain. Como consecuencia, en este tipo de blockchain, los nodos son identificados a través de un proceso de autenticación. Finalmente, una blockchain de tipo consorcio es una combinación de las dos anteriores donde un conjunto de nodos predeterminados que cuentan con la infraestructura necesaria (usualmente parte de una empresa u organización) validan transacciones y mantienen el blockchain, y los usuarios autenticados envían transacciones y consultas de la información contenida en dicha blockchain. Este esquema permite tener además de autenticación, control de acceso sobre la información [9].

Tanto las Blockhains privadas como las de tipo consorcio no son completamente descentralizadas, por lo que para esquemas descentralizados es conveniente hacer uso de blockchains públicas. Ejemplos de estas blockchains son Bitcoin y Ethereum siendo las primeras en ser creadas y de las que se encuentran diversas herramientas y documentación. Sin embargo, tienen varias desventajas como que la verificación de transacciones es lenta y su gasto de energético es alto. Además, cada transacción tiene un costo, usualmente bastante elevado, esto debido a que hacen uso del protocolo de consenso de prueba de trabajo (o Proof of Work) [20].

Para solucionar este problema, han surgido alternativas que buscan ser más eficientes en costo, tiempos de transacción y uso energético. Esto lo logran usando

otros tipos de protocolos de consenso siendo el más popular Proof of Stake (PoS). Una de estas alternativas es la red Solana [21] que propone el uso del protocolo de consenso PoS, donde cada miembro de la red apuesta una cierta cantidad de tokens de la propia red para tener la posibilidad de generar un bloque. Adicionalmente incluye el uso de un segundo protocolo llamado Proof of History (PoH), en el que una secuencia de hashes es usada como un registro del tiempo creando un orden en las transacciones y proporcionando una rápida sincronización entre los nodos de la red [21]. Estos dos, hacen de Solana una red más eficiente que las tradicionales como puede verse en [15].

Un elemento importante para el desarrollo de aplicaciones sobre blockchain son los contratos inteligentes. Un contrato inteligente es un programa de computadora auto-verificable, auto-ejecutable y resistente a la manipulación que consiste en un conjunto de reglas que se ejecutan en la Blockchain. Este programa permite ejecutar código sin la necesidad de un tercero, tomando una transacción como entrada, ejecutando el código correspondiente y desencadenando los eventos de salida [13].

Una limitante de Blockchain, como consecuencia del tiempo que requiere el verificar una transacción, es la imposibilidad de almacenar archivos. La mayoría de las soluciones existentes no contemplan el almacenar archivos completos dentro de un bloque que pertenezca a una blockchain. Con el objetivo de mantener la descentralización y distribución de dichos archivos, se requiere del uso de los llamados "sistemas distribuidos de archivos". Un ejemplo de este tipo es el Sistema de Archivos Andrew o Andrew File System (AFS) [4]. Otros ejemplos son las aplicaciones punto a punto para compartir archivos de gran tamaño entre las que se encuentran Napster, KaZaA y Bit Torrent. Finalmente está IPFS. InterPlanetary File System o IPFS es un sistema distribuido de archivos que busca conectar todos los equipos de cómputo con el mismo sistema de archivos. A diferencia de los ejemplos anteriores, IPFS está diseñado como infraestructura para construir aplicaciones sobre él. IPFS proporciona un modelo de almacenamiento en bloques con dirección a contenido de alto rendimiento con hiperenlaces con dirección de contenido, formando un árbol de Merkle. Además, IPFS combina una tabla hash distribuida, un intercambio de bloques incentivado y un espacio de nombres auto-certificable [6], haciéndolo una buena solución cuando se desea descentralización y distribución de archivos.

Usando las tecnologías mencionadas, se creará un esquema de trazabilidad de imágenes distribuido, eficiente y confiable que posteriormente puede ser usado en dar trazabilidad a otros activos digitales que no necesariamente sean imágenes. A continuación se presenta la arquitectura de este esquema usando la blockchain de Solana e el sistema de archivos IPFS.

3. Arquitectura del esquema de trazabilidad de imágenes basado en blockchain

La arquitectura del esquema propuesto se muestra en la Figura 1. Esta figura muestra la manera en que un usuario se comunica con una interfaz

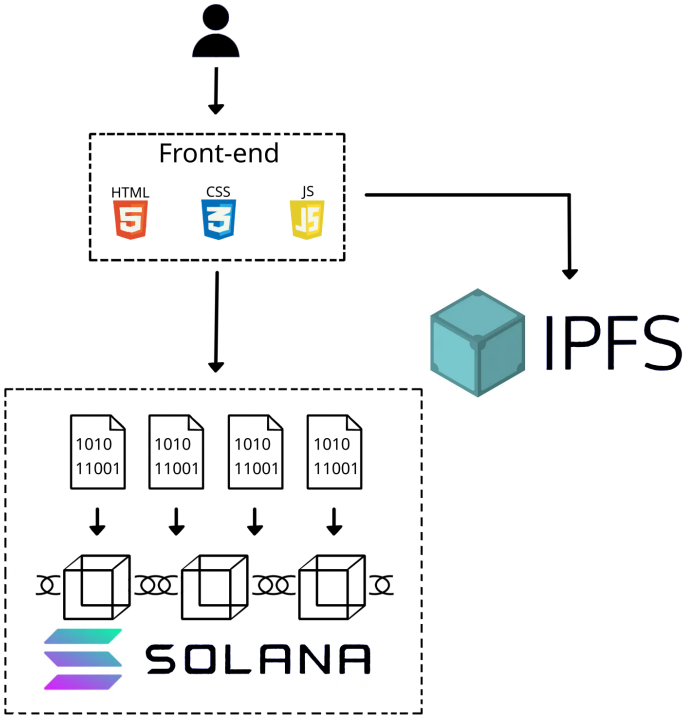

Fig. 1. Arquitectura general.

o *front-end* dentro de un navegador. Desde ahí, el usuario podrá solicitar el realizar funciones sobre las imágenes que desea compartir o descargar. A través de contratos inteligentes (mostrados en la figura como pequeños cuadros) que son ejecutados en la Blockchain se realizan las operaciones sobre las imágenes y a su vez se registran estas operaciones como transacciones en la misma Blockchain. Por último, la o las imágenes que sean parte de las operaciones realizadas son almacenadas en IPFS, desde donde podrán ser accedidas en cualquier momento.

Esta figura también muestra que nuestra propuesta hace uso de las tecnologías descritas en la Sección 2. A continuación se presentan los resultados de analizar que funciones sobre imágenes el usuario realizará.

3.1. Casos de uso

A partir de un análisis de lo que un usuario desea realizar con sus imágenes se definieron las operaciones que nuestro esquema debe soportar. Con esto, se determinó que información será almacenada dentro de la blockchain así como el diseño de los correspondientes contratos inteligentes. Así, se definieron las siguientes operaciones a realizar sobre las imágenes en nuestro esquema.

1. **Subir imagen.** Un usuario desea subir una imagen a la blockchain para compartirla con todos los usuarios de esta. Esta imagen puede subirse en claro, para que cualquier usuario pueda verla, o cifrada, para que sólo los usuarios con las llaves correspondientes pueden ver la imagen original.
2. **Modificar permisos de una imagen.** Cada imagen se subirá con ciertos permisos (público o editable). Estos permisos pueden ser modificados durante el ciclo de vida de la imagen.
3. **Descargar imagen.** Una imagen puede ser descargada de manera local si se tienen los permisos.
4. **Editar imagen.** Una imagen puede ser editada de manera local si se tienen los permisos.

A continuación se presenta una descripción a detalle del funcionamiento de estos 4 casos de uso.

Subir imagen La operación de subir una imagen, mostrada en la Figura 2, involucra un contrato inteligente que permitirá que el usuario agregue una imagen a IPFS y la evidencia de esta acción en la red Blockchain de Solana. Lo que hará dicho contrato inteligente es guardar una estructura de datos en la blockchain, la cual contiene la siguiente información de la imagen.

1. La llave púbica del usuario que sube la imagen.
2. El identificador de contenido (`cid`) de la imagen en donde `cid=hash(imagen)`. Este es un identificador único se utiliza IPFS para identificar la imagen de manera única.
3. El `cid` de la imagen padre de la cual proviene la imagen a subir. Esto aplica cuando la imagen a subir sea resultado de una edición.
4. Un indicador booleano para saber si la imagen es una edición.
5. Un indicador booleano para identificar si la imagen es la diferencia entre la imagen original y la edición.
6. Un indicador booleano para saber si la imagen es pública.
7. Un indicador booleano para establecer si la imagen es editable.

Al subir la imagen, el primer paso consiste en ejecutar el correspondiente contrato inteligente para almacenar la estructura previamente descrita dentro de la blockchain (paso 1). Una vez almacenada, habrá una respuesta exitosa por parte de la blockchain (paso 2) y entonces la imagen podrá subirse a IPFS donde permanecerá disponible, completando así el paso 3. De esta manera la imagen estará almacenada en IPFS y en la blockchain de Solana sólo se guardará la referencia única a dicha imagen.

Algo a tomar en cuenta es que el contenido en IPFS es público, esto quiere decir que cualquiera que tenga el correspondiente `cid` de la imagen podría acceder ella realizando una consulta. Por esta razón es necesario brindar la opción de cifrado para que el usuario tenga mayor control de su imagen. En caso de que el usuario quiera que su imagen sea privada, esta imagen se cifrará antes de ser subida a IPFS, almacenando en la blockchain el identificador de contenido de la

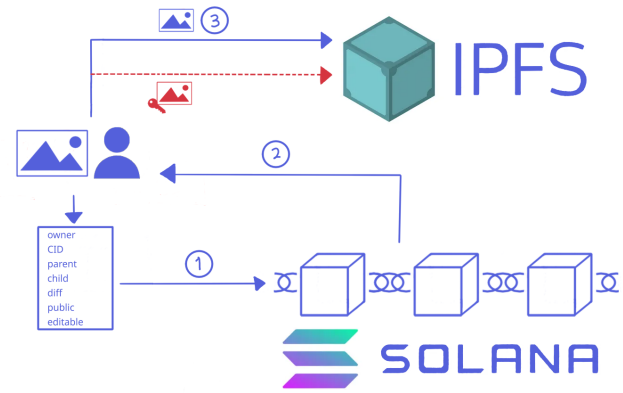

Fig. 2. Proceso de subir una imagen.

imagen ya cifrada. Este paso opcional puede observase en la figura 2 dentro del paso 3 en color rojo.

El usuario puede elegir al momento de subir una imagen si ésta es pública o privada y si es editable o no. Si la imagen es privada, será cifrada para que solo las personas permitidas tengan acceso a ésta. En ese caso, el dueño de la imagen creará una llave de cifrado simétrico con la que dicha imagen se cifrará.

La imagen 3 muestra este proceso, en donde el usuario que solicite la llave subirá a la blockchain una estructura de datos que contenga la llave pública de dicho usuario, una segunda llave pública RSA con la que se cifrará la llave simétrica, el `cid` de la imagen a la que desea acceder y un espacio en blanco donde la llave para descifrar la imagen será guardada. Posteriormente el dueño, de aceptar compartir la llave, subirá la llave simétrica cifrada con la llave pública RSA en el espacio vacío antes mencionado, el solicitante obtiene la llave y por último se elimina la información de la blockchain. De esta manera se comparte la llave de cifrado de manera segura sólo a la persona que tendrá permiso para descifrar esta imagen.

Modificar permisos Como se mencionó anteriormente, al momento de subir una imagen el usuario puede decidir si ésta es pública o privada y si es editable o no. A esto le llamamos permisos de imagen. Para modificar estos permisos se utilizará un contrato inteligente que, dependiendo de la elección del usuario, asignará uno de los permisos a la información de la imagen que se encuentra en blockchain asociada al `cid` de la imagen. Esto se presenta la figura 4. Los tipos de permisos son los siguientes.

– `Public.` Este es un campo booleano que cuando esta en 1 indica que el archivo será público, en este caso cualquier persona puede acceder a la imagen sin ninguna restricción. Si quisiéramos compartir la imagen con algún usuario

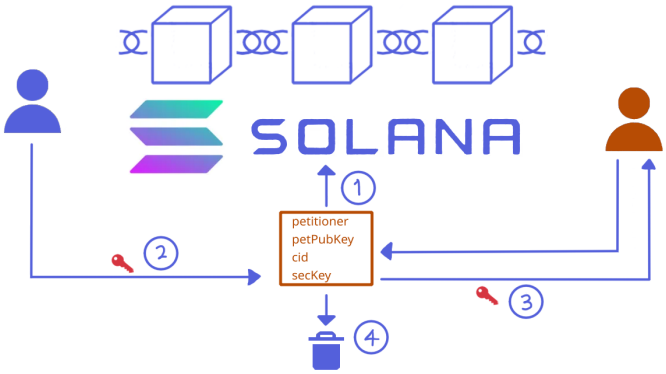

Fig. 3. Proceso de compartir llave.

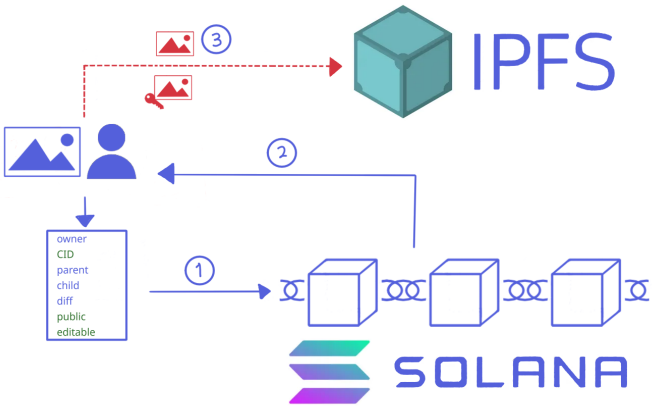

Fig. 4. Proceso de modificar permisos de una imagen.

en específico, el cual podrá consultar el archivo almacenado en IPFS, este campo estaría en 0. En esta caso la imagen estaría cifrada y se tendría que solicitar la llave simétrica correspondiente de descifrado.

– `Editable`. Este es un campo booleano que cuando esta en 1 indica que la imagen puede ser descargada directamente de IPFS para posteriormente ser editada. Un usuario con esta autorización, podrá descargar la imagen correspondiente y se generará un registro en blockchain de que esta imagen fue obtenida con el fin de realizar una modificación. Si este campo esta en 0 la imagen no podrá descargarse y como consecuencia no será editable, es decir, será de sólo lectura.

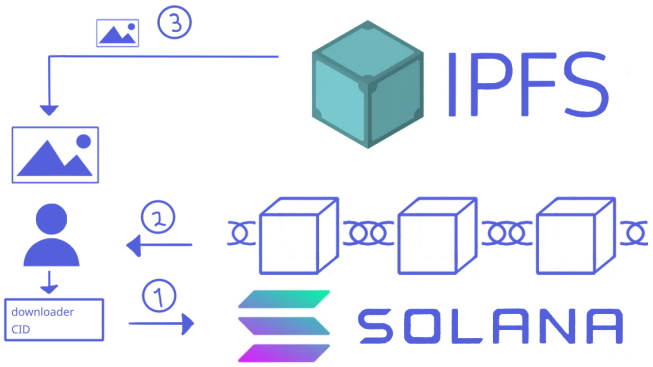

Fig. 5. Proceso de descargar una imagen.

Un usuario tiene la opción de modificar los permisos de una imagen que subió previamente. De ser así, los campos booleanos de público y editable así como el campo que contiene el `cid` de la imagen pueden ser modificados. Para modificar los permisos, el contrato inteligente actualizará los campos antes mencionados (resaltados con color verde en la figura 4) con los nuevos valores, guardando los cambios en la blockchain en el paso 1. Si el usuario decide cambiar su imagen de pública a privada o viceversa (paso 2), está deberá cifrarse o descifrarse dependiendo del cambio y por lo tanto tendrá que volverse a subir el nuevo archivo a IPFS y modificarse el `cid` que está guardado en la blockchain (paso 3).

Descargar imagen Si el usuario desea descargar una imagen, se crea una estructura de datos que incluye la llave pública del usuario que descarga dicha imagen y el `cid` de la imagen que se descarga. La figura 5 muestra este proceso. En el paso 1 se guarda un bloque de esta acción, esto es, indicando que un usuario descargó una imagen. Este bloque se guarda en la Blockchain. Si el usuario que solicita descarga no tiene permiso de edición no podrá realizar dicha descarga y requerirá que el dueño de la misma cambie los permisos. En caso de que sí exista el permiso (paso 2), la imagen puede ser descargada sin importar si es o no pública con la única consideración de que se deberá solicitar la llave correspondiente para descifrar la imagen en caso de que esta sea privada (paso 3).

Editar imagen La edición de una imagen es una combinación de las operaciones de descargar y subir imagen como puede observarse en la figura 6. Primero se ejecuta el contrato inteligente para descargar la imagen (Pasos del 1 al 4), posteriormente se ejecuta otro contrato inteligente para subir la imagen editada (Pasos del 5 al 6) y, por último, un tercer contrato inteligente es ejecutado para para subir otra imagen que muestra las diferencias entre la imagen original y la modificada (Pasos del 7 al 9). Ambas imágenes se suben a IPFS para alimentar el linaje electrónico de la imagen que las originó, mostrando los cambios por los que

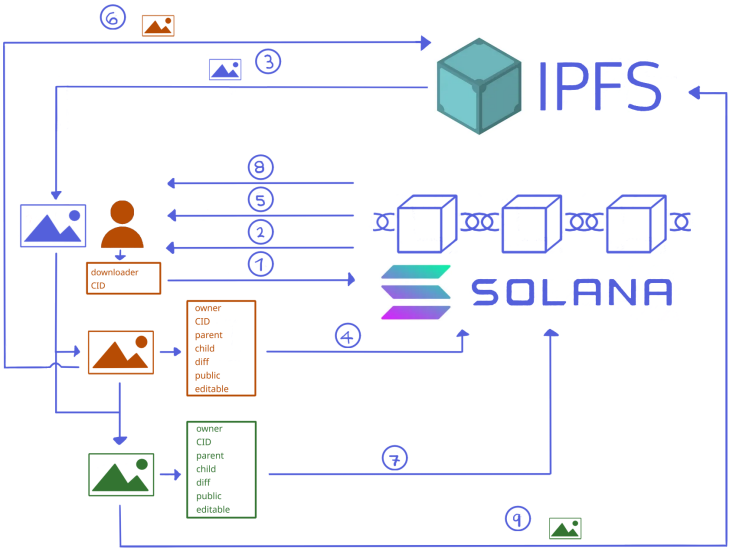

Fig. 6. Proceso de edición de una imagen.

pasó la misma. De esta manera generamos evidencia de cualquier modificación realizada a las imágenes dentro de nuestro sistema.

Profundizando un poco más en la comparación de imágenes se propone utilizar la biblioteca `jimp` (JavaScript Image Manipulation Program) de Javascript [8]. Esta implementa el método `diff` que obtiene las diferencias entre imágenes generando una nueva imagen que únicamente muestra las diferencias entre ambas imágenes. En la figura 7 puede observarse un ejemplo del proceso de comparación. Se muestra la imagen original, la imagen editada (en esta caso se cambió el color del fondo) y a la derecha la imagen resultante del proceso de comparación que muestra la modificación realizada. Estas tres imágenes son guardadas en blockchain mostrando cómo la nueva imagen editada es el resultado de la diferencia y la imagen original.

Para utilizar este método se deben tomar en consideración varios aspectos. El formato de las imágenes debe ser uno de preservación como el formato PNG. Otro aspecto a considerar es que las imágenes deben ser del mismo tamaño ya que la comparación generaría un empalme o diferencia en el resultado que no permitirá ver las modificaciones de forma clara.

Usando este análisis es posible realizar la implementación de los casos de uso descritos en forma de contratos inteligentes. En la siguiente sección se presenta dicha implementación.

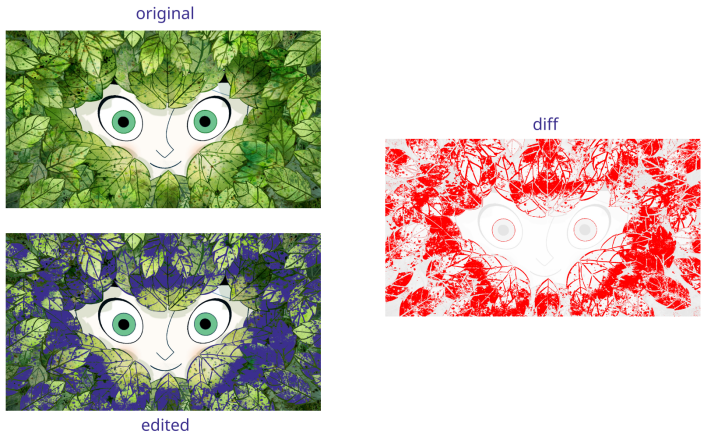

Fig. 7. Comparación de la imagen original y la modificada.

```
struct ImgData {
 owner: String,
 CID: String,
 parent: String,
 child: bool,
 diff: bool,
 public: bool,
 editable: bool
}
```

```
struct DwnldLog {
 downloader: Pubkey,
 CID: String
}
```

```
struct SecKeyRequest {
 petitioner: String,
 petitionerPublicKey: String,
 CID: String
 secretKey: Array,
}
```

Fig. 8. Estructuras de datos que se guardarán en la Blockchain.

4. Resultados

La implementación de los correspondientes contratos inteligentes se realizó utilizando los lenguajes de programación Rust y Typescript, siendo el primero necesario para los contratos inteligentes y el segundo para la interacción del usuario con la Blockchain y con IPFS. Para almacenar la información de una operación relacionada con una imagen se requiere crear una cuenta que incluye una llave privada y una pública. Además esa cuenta requiere de cierta cantidad de SOL (la criptomoneda con la que se pueden realizar operaciones en la Blockchain de Solana). En dicha cuenta se guardarán las estructuras mostradas en la Figura 8 ya descritas en la sección 3.1. Estas estructuras cambian de elementos dependiendo el caso de uso a ejecutarse. Además, en esta cuenta se almacenan y ejecutan los contratos inteligentes que implementan estos casos de uso. En esta sección nos enfocaremos en estos contratos.

Cada contrato inteligente en Solana consiste de los siguiente 6 archivos escritos en lenguaje de programación Rust.

1. `entrypoint.rs` Este es la entrada al programa y donde se enviará la información que se desea guardar en la Blockchain.

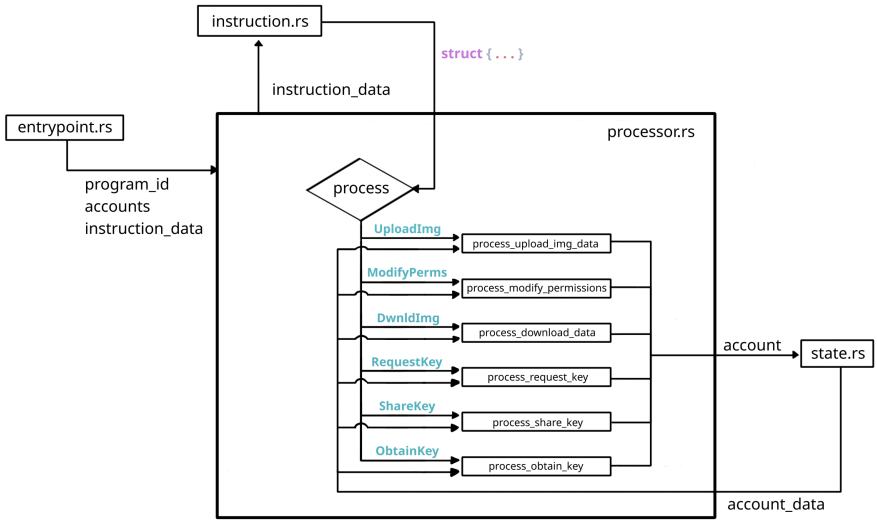

Fig. 9. Diagrama de flujo de la ejecución de un contrato inteligente.

2. `instruction.rs` En este se elige qué operación se va a realizar sobre la imagen de las descritas en la Sección 3.
3. `processor.rs` Aquí se ejecuta la operación seleccionada en `instruction.rs`.
4. `state.rs` Se define cómo se va a guardar la información en la Blockchain.

La figura 9 muestra un diagrama de flujo que describe la interacción entre los archivos de Rust. El archivo `instruction.rs` contiene las diferentes opciones que se pueden ejecutar en `processor.rs`, esto dependiendo del tipo de información de entrada en `entrypoint.rs`. Esta información es un arreglo de bytes donde el primer byte corresponde al número de instrucción (`process`) que se va a ejecutar y el resto es la información que se desea ingresar a la Blockchain. Existen 6 diferentes operaciones. Una vez elegida alguna de estas se realiza el llamado a la función correspondiente. En la imagen podemos ver que estas funciones tienen su equivalente a los Casos de Uso descritos previamente.

Por ejemplo, si se elige la opción 0 que corresponde a subir una imagen, la información dentro del arreglo de bytes que se ingresó al programa pasará a `processor.rs` donde se ejecuta la función $process\_upload\_img\_data()$. En esta función se agrega la información a la Blockchain dentro de una cuenta creada para ese fin. Esta información es la de la estructura mostrada en la figura 10. La figura 11 muestra el registro de dicha transacción en el explorador de Solana. Una vez que la transacción es aprobada en Blockchain es cuando la imagen se sube a IPFS. De esta manera garantizamos que sólo existan imágenes de transacciones validadas.

```
ImgData {
 is_initialized: 1,
 owner: '4NExjCL9Uc5Vq1z2hp8ZZHPJP3vahNsBBdnka6PB9w2S',
 cid: 'bafybeiadgjfkxogt6nn73n5qpudjjky2j6i7dp2jckbe3efxqgb2yawoqe',
 parent: '0000000000000000000000000000000000000000000000000000000000000000',
 child: 0,
 diff: 0,
 public: 1,
 editable: 1
}
```

Fig. 10. Información de la imagen disponible en la Blockchain.

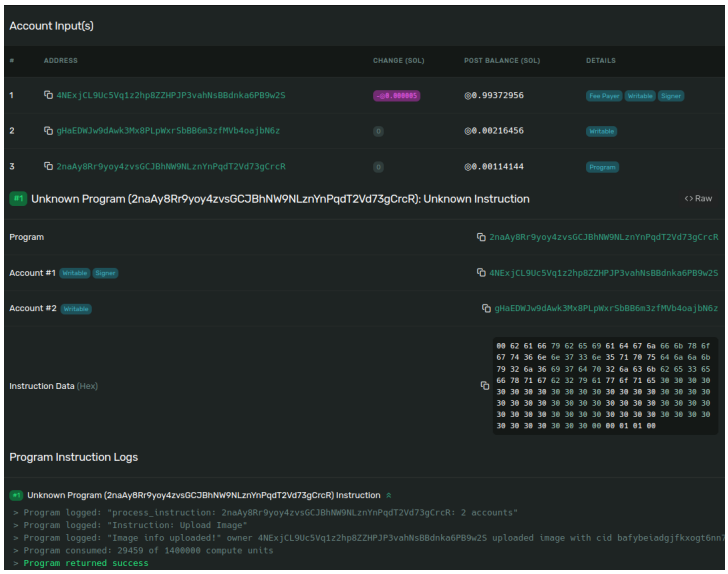

Fig. 11. Explorador de la Blockchain Solana.

Si se accede a la imagen desde IPFS mediante el identificador de contenido almacenado en la Blockchain (figura 12), se puede corroborar que el proceso tuvo éxito y efectivamente la imagen esta en IPFS.

4.1. Consulta del linaje electrónico de una imagen

Una vez que las operaciones realizadas sobre las imágenes han sido documentadas y almacenadas en la Blockchain de Solana e IPFS, se pueden realizar consultas que muestren el linaje electrónico de estas imágenes.

Estas consultas son realizadas haciendo uso de los mensajes de registro del contrato inteligente emitidos en cada transacción. Durante la ejecución del contrato inteligente se generan ciertos mensajes que indican qué imagen está involucrada en la operación (identificada por el `cid`), de qué operación se trata (de las mostradas en la figura 9) y el usuario que está ejecutando la operación

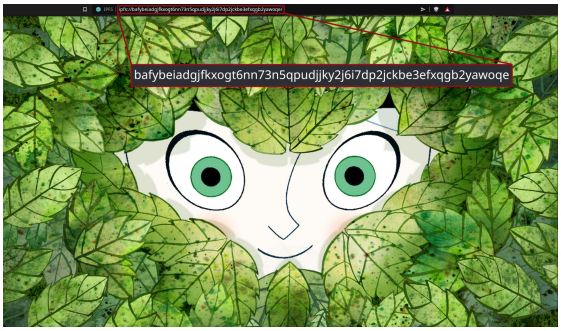

Fig. 12. Imagen vista desde IPFS.

Fig. 13. Imagen y su respectiva información.

(identificado por su llave pública), además de información adicional que depende de la misma operación.

De esta manera se realiza una búsqueda de las transacciones realizadas por un usuario donde el `cid` involucrado coincida con el `cid` de la imagen de interés. Al realizar esto, será posible observar los procesos que tuvieron lugar e ir construyendo el linaje de la imagen través de los registros en las transacciones dentro del Blockchain.

La figura 13 muestra una imagen almacenada en IPFS y su respectiva información almacenada en la Blockchain de Solana. Sabemos que dicha imagen es una edición de otra imagen, ya que su información muestra el indicador `child` con un valor de 1 indicando la presencia de una imagen padre.

A partir del `cid` de esta imagen padre se inicia la consulta del linaje electrónico (figura 14), observando la transacción que registró la edición de la imagen se tiene el `cid` (en color verde) que corresponde a la imagen mostrada en la figura 13 y el `cid` de la imagen padre (en color rojo).

Al consultar las transacciones de la imagen padre se obtienen 4 transacciones (Tx_i) mostradas en la figura 14. En esta imagen podemos observar que sólo la Tx_3 corresponde a la edición de una imagen, dicha imagen (con `cid` en color morado) resulta ser la imagen original. Al consultar el resto de las transacciones se observa un proceso de edición, un cambio de permisos y la subida de la imagen padre.

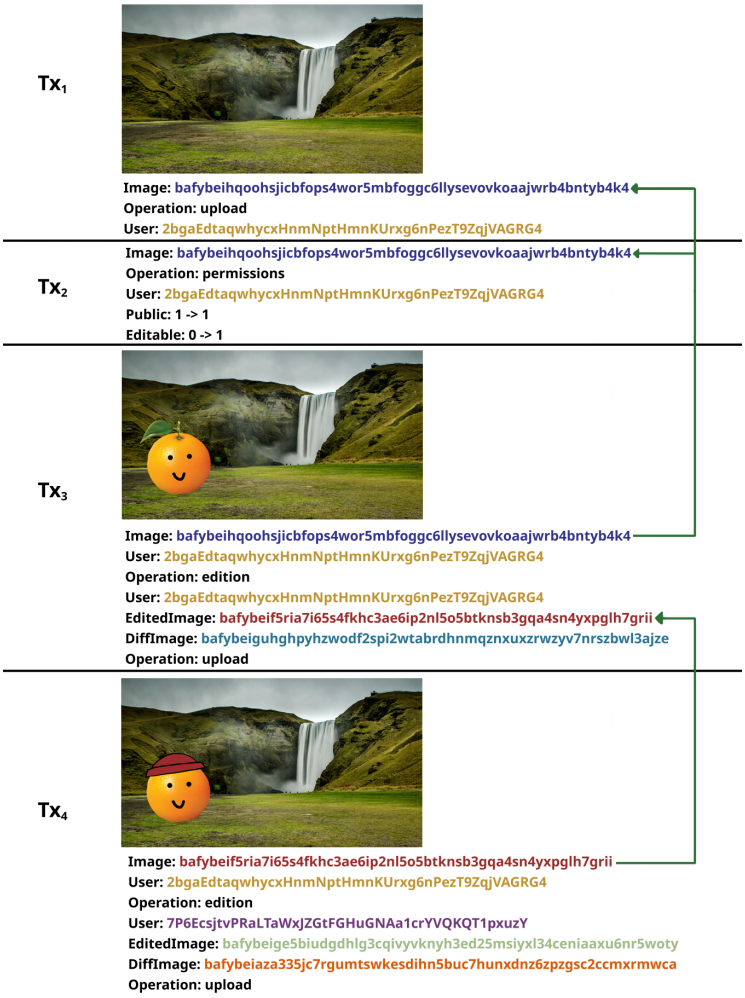

Fig. 14. Muestra de la consulta del linaje electrónico de la imagen.

Así, el linaje completo de este ejemplo puede describirse de la siguiente manera. La imagen con `cid` en color morado fue subida a IPFS siendo pública y de solo lectura (Tx_1), posteriormente el usuario decidió modificar uno de los permisos y hacer la imagen en editable (Tx_2). En algún momento el mismo usuario realizó una edición y la documentó (Tx_3). Finalmente, esta imagen resultado fue nuevamente editada pero ahora por el usuario con llave pública en color morado (Tx_4).

Todo esto se realiza utilizando los mensajes creados específicamente para documentar dentro de los registros de las transacciones las operaciones que se realizan, facilitando así la consulta del linaje electrónico.

5. Conclusiones

Es clara la necesidad que existe de que las personas tengan control sobre la información que comparten en Internet. La utilización de Blockchain para almacenar el linaje electrónico de imágenes brinda integridad y no repudio a la información referente a la historia de dicho activo. Sin embargo, esto requiere de una tecnología eficiente, sustentable y que permita el almacenamiento de archivos.

En vista de esto, el trabajo presentado incluye el uso del protocolo de consenso PoS con PoH y el protocolo de almacenamiento IPFS. Esto nos permitirá documentar los cambios que sufren las imágenes usando contratos inteligentes con tiempos de transacción cortos y almacenamiento distribuido. También, el construir este esquema sobre una arquitectura distribuida permite confiar en toda la información generada, garantizando que las medidas de control y trazabilidad implementadas serán cumplidas.

Se logró realizar una implementación del esquema propuesto haciendo uso de la Blockchain de Solana a través de los lenguajes de programación Rust y Typescript. Esta implementación logra registrar y ejecutar las operaciones de subir una imagen, modificar sus permisos, descargarla y editarla, además de proporcionar métodos para generar y compartir llaves para cifrar y descifrar simétricamente imágenes que son clasificadas como privadas. Finalmente, se implementó la consulta del linaje electrónico y su posterior análisis.

Como parte del trabajo futuro está el realizar una implementación completa que incluya el desarrollo de un *front-end* para que usuarios finales puedan ser parte de este sistema. A la vez, teniendo mucha más información relacionada al uso de imágenes, se podría desarrollar un contrato inteligente que permita la visualización de las consultas de la historia de las imágenes de forma más amigable para el usuario, de tal forma que cada dueño de una imagen tenga información de cómo están siendo usadas sus imágenes.

Acknowledgements. Se agradece a DGAPA por el proyecto PAPIIT TA101021 y a CONACyT por el apoyo a través de la beca de maestría 1085102.

Referencias

1. Aldeco-Pérez, R., Aguilar Torres, G., Cruz Cortés, N., Domínguez Perez, L. J., Escamilla Ambrosio, P. J., Gallegos García, G., León Chavez, M. A., Monroy Borja, R., Rodríguez Henríquez, L. M., Rodríguez Henríquez, F. J., Rodríguez Mota, A., Salinas Rosales, M., Silva Trujillo, A. G.: Introducción a la Ciberseguridad y sus aplicaciones en México. Academia Mexicana de Computación, A. C., 1 edn. (2020), http://amexcomp.mx/files/LibroCiber-ISBN-V2.pdf
2. Aldeco-Pérez, R., Leon Chavez, M.: Evaluar la Calidad de los Objetos de Aprendizaje Mediante Linaje Electrónico. In: El Desarrollo de los Recursos Digitales para la Educación en México, pp. 240. Benemérita Universidad Autónoma de Puebla, 1st edn. (2013), https://tinyurl.com/2p8akmch

3. Aldeco-Pérez, R., Moreau, L.: Securing provenance-based audits, vol. 6378 LNCS. Springer (2010)
4. Arpaci-Dusseau, R. H., Arpaci-Dusseau, A. C.: The Andrew File System (AFS). In: Arpaci-Dusseau Books, pp. 1—-14 (2014), https://pages.cs.wisc.edu/∼remzi/OSTEP/dist-afs.pdf
5. Azaria, A., Ekblaw, A., Vieira, T., Lippman, A.: Medrec: Using blockchain for medical data access and permission management. In: 2016 2nd International Conference on Open and Big Data (OBD). pp. 25–30 (2016) doi: 10.1109/OBD.2016.11
6. Benet, J.: Ipfs - content addressed, versioned, p2p file system (2014)
7. González-Ortega, A., de Asís López-Fuentes, F.: A web-based didactic tool for teaching of distributed consensus. Res. Comput. Sci., vol. 148, no. 5, pp. 25–32 (2019)
8. JIMP: `www.npmjs.com/package/jimp`
9. Joshi, A., Han, M., Wang, Y.: A survey on security and privacy issues of blockchain technology. Mathematical Foundations of Computing, vol. 1, pp. 121–147 (01 2018) doi: 10.3934/mfc.2018007
10. Khatal, S., Rane, J., Patel, D., Patel, P., Busnel, Y.: Fileshare: A blockchain and ipfs framework for secure file sharing and data provenance. In: Patnaik, S., Yang, X.-S., Sethi, I. K. (eds) Advances in Machine Learning and Computational Intelligence. pp. 825–833. Springer Singapore, Singapore (2021)
11. Mattila, J.: The blockchain phenomenon – the disruptive potential of distributed consensus architectures. ETLA Working Papers 38, Helsinki (2016), http://hdl.handle.net/10419/201253
12. McDaniel, P.: Data provenance and security. IEEE Security Privacy, vol. 9, no. 2, pp. 83–85 (2011) doi: 10.1109/MSP.2011.27
13. Mohanta, B. K., Panda, S. S., Jena, D.: An overview of smart contract and use cases in blockchain technology. In: 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT). pp. 1–4 (2018) doi: 10.1109/ICCCNT.2018.8494045
14. Moreau, L., Groth, P., Miles, S., Vázquez-Salceda, J., Ibbotson, J., Sheng, J., Munroe, S., Rana, O., Schreiber, A., Tan, V., Varga, L.: The provenance of electronic data. Commun. ACM, vol. 51, pp. 52–58 (04 2008) doi: 10.1145/1330311.1330323
15. Pierro, G. A., Tonelli, R.: Can solana be the solution to the blockchain scalability problem? In: 2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER). pp. 1219–1226 (2022) doi: 10.1109/SANER53432.2022.00144
16. Pixel, M.: México sigue teniendo problemas con las fake news: algunas de estas fotos son reales pero no son de Cancún (2017), https://www.xataka.com.mx/otros-1/mexico-sigue- teniendo-problemas-con-las-fake-news-algunas-de-estas-fotos-son-reales-pero-no-son-de-cancun
17. Sifah, E. B., Xia, Q., Agyekum, K. O.-B. O., Xia, H., Smahi, A., Gao, J.: A blockchain approach to ensuring provenance to outsourced cloud data in a sharing ecosystem. IEEE Systems Journal, pp. 1–12 (2021) doi: 10.1109/JSYST.2021.3068224
18. Sivleen, K., Sheetal, C., Aabha, S., Jayaprakash, K.: A research survey on applications of consensus protocols in blockchain. Security and Communication Networks, vol. 2021 (01 2021) doi: 10.1155/2021/6693731

19. Wen, T.: The hidden signs that can reveal a fake photo (2020), https://www.bbc.com/future/article/20170629-the-hidden-signs-that-can-reveal-if-a-photo-is-fake
20. Xiao, Y., Zhang, N., Lou, W., Hou, Y. T.: A Survey of Distributed Consensus Protocols for Blockchain Networks. IEEE Communications Surveys and Tutorials, vol. 22, no. 2, pp. 1432–1465 (apr 2019) doi: 10.1109/COMST.2020.2969706
21. Yakovenko, A.: Solana: A new architecture for a high performance blockchain v0.8.13. Tech. rep. (2020)
22. Yañez, B., Galván, M., Ramírez, S.: El ABC de la 'Ley Olimpia': sus alcances y retos (2022), https://politica.expansion.mx/sociedad/2022/21/25/el-abc-de-la-ley-olimpia-sus-alcances-y-retos