



# FLASH

GO FASTER

# WITHEPAPER

TECNOLOGÍA BLOCKCHAIN PARA COMUNIDADES

## Abstracto

FLASH es un Blockchain público que opera a beneficio de cada persona en el mundo. Es una plataforma hecha para clientes, comerciantes y desarrolladores para aprovechar esta poderosa tecnología en los momentos cotidianos

Hemos creado un campo de juego transparente y justo para todas las personas; limitado solo por tu imaginación. Flash Te invita a experimentar esta característica rica del Ecosistema de la próxima generación de tecnología Blockchain. Reuniendo nuestros valores centrales con los valores de la comunidad, Estamos orgullosos de presentar esta nueva fase en la evolución de FLASH.

# Contenidos

Abstract .....	1
Introducción .....	3
Bases Legales .....	4
La Filosofía y Aplicaciones FLASH.....	5
Descripción general de la arquitectura FLASH.....	7
Visión general y procesamiento de FLASH .....	11
FLASH Web Wallet, Cuenta Estructura + Clave Generación, almacenamiento y recuperación .....	29
FLASH Blockchain .....	33
Apéndice: API de Wallet Webservice .....	35

# Introducción

El principio fundamental de FLASH es que todo trabajo o contribución a la red debe ser valorado por la comunidad de manera objetiva . Permitir que funcione el proceso de libre mercado crea un mecanismo por el cual todas las formas de trabajo se pueden reducir a un denominador común: - FLASH.

Cualquier forma de trabajo , ya sea un tiempo valioso , el trabajo y la atención de un usuario , un conjunto de habilidades especiales como el desarrollo de herramientas , diversas formas de energía (es decir, procesamiento ) y liquidación , se pueden valorar en tiempo real en función de la oferta y demanda del mercado . trabajo / contribución específica.

Las formas de contribuir a la comunidad solo están limitadas por la propia imaginación y la valoración colectiva de la comunidad . En esencia, FLASH es fácil de entender, tiene un núcleo completamente transparente , distribuido sin reglas y sin condiciones complicadas con las que es difícil trabajar. Ponemos la confianza en la comunidad y la tecnología. Debido a que FLASH tiene un sistema de liquidación muy simple y rápido junto con muchas herramientas para los miembros de la comunidad, tendrá muchos casos de uso.

FLASH no tiene un valor definido ; puede ser intercambiado libremente por miembros de la comunidad que deciden independientemente la disposición de su propiedad privada.

---

# Bases Legales

FLASH, como Bitcoin, fue diseñado desde cero para cumplir con las leyes. Originalmente basada en una bifurcación de Litecoin, la tecnología fue desarrollada en Canadá por Flashnet Tech, Inc . con el apoyo de donantes que recibieron activos de contraparte llamados FLASHPRE y MEGAFLASH . No se permitió donantes de los Estados Unidos . Un miembro de la comunidad en Vietnam luego extrajo todo el suministro de FLASH y se lo dieron a un miembro anónimo de la comunidad para distribuirlo. Todo el suministro pre-extraído de monedas FLASH se lanzó en forma gratuita por medio de Airdrops a los miembros de la comunidad y las personas que probaron FLASH. La moneda no tenía valor cuando se distribuyó , ni hay ningún valor definido para FLASH.

Hasta la fecha, ~ 818 millones de monedas se han reclamado y circulado, mientras que ~ 82m FLASH permanecen sin reclamar a partir del 1 de mayo de 2018 . Tras completar la primera versión del software FLASH y la distribución de monedas FLASHPRE , MEGAFLASH y FLASH, Flashnet Tech se cerró y se creó la Fundación del Tercer Milenio para la Economía y la Cultura en Liechtenstein para apoyar las monedas comunitarias y promover la tecnología FLASH . FLASH es un protocolo mantenido por la comunidad. No es ningún tipo de entidad legal o compañía de ningún tipo , no tiene empleados , contratos y ningún tipo de posición. Es una idea en el dominio público.

# La filosofía de FLASH y sus aplicaciones

El blockchain de FLASH se ejecuta como un servicio público sin fines de lucro por parte de la comunidad FLASH. No se realiza ninguna actividad comercial en la operación de la cadena de bloques FLASH, ya que el objetivo del diseño era hacer que el uso de FLASH fuera casi gratuito. FLASH es ideal para usar en cualquier país, su diseño original era para países en desarrollo, donde se necesitan transacciones rápidas y baratas. La actividad comercial se lleva a cabo dentro de aplicaciones que utilizan blockchain, que puede tener funciones de monedero, o que acceden directamente al blockchain o al API. Los incentivos económicos pueden ser creados por aplicaciones que usan la cadena de bloques FLASH, generando tarifas directamente en las aplicaciones cuando se realizan transacciones u otros servicios.

Las aplicaciones que varios desarrolladores de aplicaciones están desarrollando con FLASH incluyen:

- Medios y juegos. FLASH es ideal para recompensar a los creadores de contenido y jugadores en lugares remotos.
- Intercambio personal (también conocido como ATM humano). Los países en desarrollo suelen no tener la infraestructura bancaria de los países desarrollados. Esto crea amplias oportunidades para el intercambio personal, una forma muy simple y rápida de intercambiar monedas de cifrado en persona o por Internet en cualquier lugar que sea legal.
- Remesas. Porque FLASH es una moneda de bajo costo, tan pronto como se crean intercambios en diferentes países, es posible hacer remesas rápidas y de bajo costo, ya sea a través de intercambios o intercambio personal.
- FLASH Web Wallet. La forma más rápida y conveniente de obtener comenzó con FLASH; no hay nada que instalar, solo un registro rápido vía correo electrónico. La billetera web FLASH pronto incluirá una función de envío social.

- FLASH Android Wallet . Android tiene un 75% de participación en el mercado mundial y lidera en los países en desarrollo por un amplio margen. Una billetera con simple envío y recepción.
- 25 idiomas y el conjunto de herramientas de intercambio personal que incluye detección de clientes, mapa interactivo (usando servicios de ubicación) y chat.
- (futuro ) Circunstancia . Esto utilizará los nodos FLASH para retransmitir transacciones, mensajes y archivos a través de una red distribuida. FLASH será el gas.
- (futuro) DEX Blockchain Marketplace . FLASH se puede utilizar como un plano de señalización para anunciar ofertas para comprar o vender criptomonedas de forma descentralizada.

# Descripción general de la arquitectura FLASH

FLASH ES PRE-MINADO, BLOCKCHAIN DECENTRALIZADO, BASADO EN LA BLOQUEO DE BITCOIN / LITECOIN ORIGINAL . HA SIDO OPTIMIZADO PARA VARIOS CASOS DE USO MUY ESPECÍFICOS:

1. Compatibilidad con BTC / LTC que hace que FLASH sea muy fácil de agregar a las nuevas centrales (solo unas pocas horas) y el puerto a las herramientas de software de código abierto , como la billetera de hardware Trezor.
2. Alto rendimiento . Las transacciones se pueden liquidar en menos de 2 segundos , pero generalmente menos de 5 segundos y las monedas se pueden volver a enviar inmediatamente a otro miembro de la comunidad . La liquidación permanente final ocurre en menos de 2 minutos.
3. Altamente seguro. Se han parcheado varios defectos de seguridad con la cadena de bloques bitcoin original . La plataforma se ha mejorado con el cifrado Curve 25519 Curva elíptica , que ahora es estándar en BTC / LTC.
4. Capacidad de producción a escala comercial . Delegar la minería permite una capacidad mucho mayor del sistema . Se presentarán pruebas reales del rendimiento real en vivo.
5. Bajo costo y respetuoso del medio ambiente . La eliminación de casi toda la costosa extracción elimina gran parte del costo que la mayoría de las criptomonedas de prueba de trabajo (PoW) suman a través de la inflación o altas tarifas de transacción.
6. Diseñada para comunidades . Es una moneda comunitaria , emitida y utilizada por una comunidad , no es una utilidad ni una ficha de seguridad. La cadena de bloques FLASH es casi gratuita, mientras que los usuarios comerciales pueden usar esta cadena de bloques para crear aplicaciones usando FLASH y cobrar tarifas , enviando FLASH directamente a las carteras de los desarrolladores o socios de la aplicación.
7. Una billetera web con funciones fáciles de usar que incluyen Enviar usando correo electrónico o dirección pública , historial de transacciones, solicitud de FLASH, soporte de códigos QR, contactos , Autenticación de 2 factores (2FA ), herramientas comerciales y recuperación de claves.

8. Se admiten billeteras de terceros , incluidas billeteras Qt, Coinomi , CoinPayments , ETHOS y Android . Se espera soporte para la billetera de hardware Trezor pronto.

Para alcanzar estas medidas críticas para el uso a gran escala de millones de miembros de la comunidad , la plataforma FLASH utiliza un modelo con núcleo delegado y minería mínima . La plataforma aprovecha la tecnología de base de datos distribuida existente con las características de la tecnología blockchain , control descentralizado , inmutabilidad y creación y movimiento de activos digitales.

El sistema de billetera web FLASH se basa en plataformas de aplicaciones de tres niveles . Como la mayoría de los sistemas estándar . tenemos niveles de Interfaz de usuario , Comunicaciones y Lógica empresarial / Almacenamiento.

La capa de interfaz de usuario permite al usuario o aplicación final interactuar con la cadena de bloques FLASH. La plataforma de billetera web FLASH aprovecha HTML5, CSS3 y JavaScript en los navegadores , sin extensiones , lo que permite una compatibilidad perfecta entre navegadores. Hemos desarrollado y adoptado todas las tecnologías que permiten a JavaScript hacer lo que C / C ++ y Java pueden hacer. Además, el sistema FLASH utiliza Twitter Bootstrap para proporcionar un marco web receptivo que funciona en cualquier dispositivo . El nivel de comunicación permite un túnel seguro entre el nivel de interfaz de usuario y el nivel de lógica de negocios / almacenamiento sin la necesidad de OpenSSL . Los protocolos de seguridad utilizados por la billetera web FLASH incluyen:

Nuestro trabajo en el área de la comunicación también significa que somos los primeros en adoptar nuevos protocolos seguros que ejecutan JavaScript mediante HTML5.

Business Logic / Storage Tier permite que todos los flujos de transacciones, la lógica comercial y los sistemas de red se almacenen y operen dentro de esta capa. Este nivel es responsable de ejecutar algoritmos patentados para almacenar en la base de datos distribuida , que alimenta FLASH.

- El Nodo clave de intercambio es un clúster horizontal de servidores que proporciona intercambio de claves o búsqueda de claves para cada transacción en el sistema FLASH, como un índice que se ejecuta junto a la cadena de bloques para buscar cosas más rápido . En el futuro cercano , estos nodos estarán descentralizados . Debido a la naturaleza de los pares de claves criptográficas en el sistema transaccional de Bitcoin, cada transacción requiere una búsqueda de dirección de billetera pública . El intercambio clave debe actuar como registrador público . El Nodo Clave también permite el intercambio de divisas , mensajes , plica pública y otro flujo de información entre billeteras de igual a igual . El motor de mensajes ha habilitado el registro y la notificación de todas las actividades en la comunicación de billetera mediante la notificación por correo electrónico de Sendgrid . El Nodo clave utiliza NodeJS , ZeroMQ , Redis y MySQL y está alojado por varios miembros de la comunidad para el beneficio de la comunidad

- La aplicación FLASH Web Wallet es una aplicación de servidor de escala horizontal que permite que todas las carteras FLASH se utilicen desde diferentes navegadores web. La aplicación FLASH Wallet tiene la mayoría de las funcionalidades Bitcoin Wallet y funciones de intercambio de claves. Es una billetera virtual en línea que utiliza el protocolo HTTP / Web Sockets. Además, las billeteras Qt con código fuente se han proporcionado sin extraer para Windows, Mac y Linux. Los intercambios son compatibles con FLASH.
- Los Servidores API de servicios web preprocesan, convierten e indexan todas las transacciones FLASH desde la Interfaz Web y las envían a la cadena de bloques. Esto acelera significativamente las transacciones ya que cada billetera no necesita sincronizar todos los bloques de la cadena de bloques del servidor local. Todas las carteras pueden compartir el blockchain en el servidor API de Blockchain . Esto mejora significativamente el rendimiento de la transacción debido a la latencia de red reducida para cada sincronización de monedero . Otro aspecto muy importante de la tecnología es la reducción significativa en la posibilidad de "doble gasto". Todas las transacciones de blockchain se indexan, preprocesan y validan a través del servidor de puerta de enlace de una manera similar a Bitcoin.
- El Blockchain FLASH es una bifurcación de la tecnología Litecoin . Se han realizado una serie de modificaciones significativas . La configuración ha sido modificada para acelerar transacciones a la cadena de bloques . Todas las monedas han sido minadas previamente y la extracción se ha restablecido justo por encima del factor de dificultad mínima . La comunidad FLASH en general seleccionará hasta 151 nodos de Delegado replanteados. Se requiere que cada delegado mantenga un mínimo de 1 millón de FLASH estacado en una billetera Qt. Estos delegados eligen hasta 25 mineros y determinan el gobierno de la cadena de bloques FLASH , las tarifas de transacción y otros asuntos . Los delegados y mineros comparten las tarifas de transacción y un conjunto de FLASH donados por la comunidad . Los mineros son aprobados por los delegados.
- El objetivo de FLASH Blockchain es reemplazar la base de datos transaccional tradicional con una base de datos de almacenamiento de red e incluir una estructura de datos de seguridad de extremo a extremo.
- Las tarifas de transacción están actualmente configuradas en 0.001 FLASH por transacción , estas tarifas pueden aumentarse o reducirse en el futuro , dependiendo de la votación de los nodos del Delegado elegidos.

# Descripción general del procesamiento de Flash

## Abstracto

FLASH implementará un nuevo modelo de consenso basado en delegados que depende de delegados de confianza, elegidos por la comunidad para llegar rápidamente a un consenso sobre el blockchain y garantizar su seguridad. Cada usuario podrá usar las monedas que controlan para emitir votos para los delegados. Los delegados elegidos votarán en asuntos relacionados con la red, tales como tarifas de transacción y selección de mineros. Los delegados eligen un pequeño conjunto de 25 mineros y estos mineros son los únicos que pueden crear nuevos bloques en la cadena de bloques. En lugar de usar PoS o PoW para asegurar la cadena, la red impone un conjunto de reglas para controlar el orden de los derechos de generación de bloques que se otorgan al conjunto de mineros confiables. Este nuevo modelo de delegado mantiene un alto grado de seguridad de la red y, al mismo tiempo, permite un rendimiento de transacciones muy elevado.

En realidad, es realmente simple: si  $> 50\%$  de los mineros electos "votan" que un bloqueo es válido, entonces ese bloqueo es permanente y nunca se puede deshacer. El "voto" es emitido creando un bloque en la cadena que viene después del bloque en cuestión. La cadena de bloque funciona como el bloque  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$ , etc. Entonces, si  $> 50\%$  de los Mineros construye un bloque en la cadena después del bloque 1, entonces sabemos que el bloque 1 es legítimo y permanente. Ningún nodo aceptará una cadena de bloques conflictiva que intente decir que el bloque 1 no es válido.

## Tipos de entidad de red

Hay tres tipos de entidades en la nueva red de FLASH:

1. Usuarios normales
2. Delegados
3. Mineros

### Usuarios Normales

Cualquier dirección FLASH que contenga Flashcoin se considera un usuario normal para el propósito de esta discusión. Al igual que en Bitcoin, no hay identificadores de usuario en la cadena, solo los UTXO están controlados por direcciones pseudo anónimas..

### Delegados

Los delegados son elegidos por usuarios normales. Los delegados deben postularse para elecciones eligiendo un mínimo de

1,000,000 de FLASH y proporcionar información sobre sí mismos, incluida una ID de delegado. Los usuarios usan la moneda que controlan para votar a los delegados en la oficina. Una vez elegidos, los delegados pueden votar sobre asuntos relacionados con la red FLASH. Cada delegado coloca los votos que se ponderan en función de:

1. La cantidad de FLASH que han replanteado
2. La cantidad de votos que han recibido
3. Cuánto tiempo han estado en el cargo (antigüedad)

Los delegados son elegidos por un período de 30 días a la vez. Los votos para los delegados se pueden emitir en cualquier momento, pero solo se evalúan una vez cada 30 días para cambiar de asiento. Se permitirá un máximo de 151 Delegados . Si hay más de 151 nodos que se postulan para elección como Delegado, se elegirán los 151 mejores según los criterios anteriores.

Los delegados deben apostar su moneda para comenzar a postularse para las elecciones , y su moneda debe seguir apostando durante todo el período electoral . Los terceros también pueden apostar monedas para cualquier Delegado dado , pero esas monedas apostadas se bloquean durante todo el período de elección y el término en el cargo de la misma manera que si el Delegado hubiera estacado las propias monedas . Los delegados pueden retirarse de la carrera para el siguiente ciclo de elección en cualquier momento, pero sus monedas deben seguir apostadas para el ciclo actual . Los delegados serán removidos automáticamente de sus oficinas si se vuelven inactivos.

Los delegados reciben una parte de las tarifas de transacción de red como compensación por sus funciones.

### **Delegados Permanentes**

De los 151 delegados , 50 son delegados permanentes que no requieren votación por medio de elecciones . Estos Delegados Permanentes son por lo demás los mismos que los Delegados elegidos; deben apostar el mínimo de 1,000,000 de FLASH, su peso de voto se basa en los mismos factores , pueden ser mineros , reciben la misma remuneración y tienen los mismos requisitos , excepto el requisito mínimo de estaca para 1,000 ,000 FLASH en lugar de 2,000 ,000 FLASH requerido por los delegados.

Los delegados permanentes no tienen que postularse para las elecciones . Los puestos permanentes de los delegados se llevan a cabo inicialmente por aquellos que han contribuido a la fortaleza y el bienestar del ecosistema FLASH ; estas posiciones son transferibles.

Si un Delegado Permanente no apuesta el mínimo requerido de 1,000,000 de FLASH , o se vuelven inactivos , ellos mantienen su posición como Delegado Permanente pero no pueden actuar como Delegados hasta que cumplan con sus obligaciones o hasta que se transfiera a alguien que cumpla con los requisitos para una buena posiciónq

### **Mineros**

Solo los delegados pueden convertirse en mineros, y lo hacen al señalar que desean ser mineros y al obtener el apoyo de sus colegas delegados en forma de votos y al apostar un mínimo de 2.000. 000 de FLASH. El mínimo de replanteo para los delegados permanentes se estableció inicialmente en 1,000,000 de FLASH . Los delegados son por lo tanto responsables de votar solo en mineros confiables y de alta calidad. El número de mineros se mantiene bajo, a un máximo de 25, con el fin de admitir altas tasas de bloqueo en la red y un alto rendimiento de las transacciones. Cuanto mayor sea el grupo de minería , menos eficiente se vuelve , y el conteo limitado de Mineros de FLASH es un equilibrio óptimo de redundancia , seguridad y rendimiento . Los mineros reciben una parte de las tarifas de transacción de red como compensación por sus deberes.

### Comisión de Transacción:

En el arranque de la red habrá una tarifa predeterminada de 0.001 FLASH por kilobyte , con una tarifa mínima de 0.33 FLASH . La tarifa de tarifa de transacción y el mínimo son parámetros que los delegados pueden votar para ajustar en el futuro. Los mineros no cobran las tarifas de transacción por bloque , sino que se cobran una vez por día y se distribuyen tanto a los mineros como a los delegados.

### Mecánica de replanteo:

Para postularse para una elección, una persona debe crear una dirección FLASH para representar su nodo. El candidato será identificado por esa dirección y las acciones que tomarán se firmarán con la clave privada asociada. Para participar en las elecciones, el candidato puede usar la billetera Qt para crear transmitir el formulario de registro de su nodo junto con su apuesta FLASH. Todos los registros y apuestas se envían a una dirección de elección especial desde la cual la red no permitirá el gasto, excepto para devolver las monedas replanteadas al remitente a petición del remitente . Se aplican estrictas reglas sobre la aceptación y devolución de fondos hacia y desde el domicilio de la elección especial.

Los delegados deben apostar un mínimo de 1,000,000 de FLASH para inscribirse en las elecciones. Cuantas más monedas apueste un delegado, más fácil será su elección, mayor será la influencia que tendrán en los votos futuros y mayor será la recompensa que recibirán . Se pueden hacer contribuciones de participación adicionales para un Delegado por cualquier persona en cualquier momento , sin embargo , todos los fondos apostados permanecerán bloqueados durante toda la campaña electoral , así como para el (los) siguiente (s) período (s) de elección si se elige a ese Delegado.

### Mecánica de votación

Cada transacción tiene la oportunidad de votar por un delegado anteponiendo un resultado de transacción de valor cero a esa transacción que usa el código de operación OP\_RETURN seguido de nuevos códigos de operación y datos relacionados con la votación. Las salidas relacionadas con la votación OP\_RETURN tx siempre deben ser la primera salida tx. El remitente de una transacción puede usar estos metadatos OP\_RETURN para especificar qué delegados obtienen la cantidad de poder de votación de la transacción.

La billetera Qt hará que la administración de los votos de un usuario sea lo más sencilla posible al mostrar qué monedas en su billetera votan para qué delegados y qué monedas no votan, y gestionar automáticamente la emisión de votos de acuerdo con la configuración especificada por el usuario. La votación no es necesaria en una transacción, pero se recomienda para la seguridad de la red.

Este método en el que el remitente establece los metadatos de votación tiene un desafortunado efecto secundario en la experiencia del usuario : cada vez que una persona recibe FLASH tendrá que devolvérselo junto con su voto si quieren que FLASH se use para votar. una manera diferente que el remitente especificado . La billetera Qt se puede configurar para que haga esto automáticamente , de modo que apenas sea perceptible para el usuario . Esto es una molestia y generará algunas transacciones redundantes a medida que los usuarios reenvíen fondos para volver a emitir los votos, pero este método permite una contabilidad y una escala dramáticamente más eficientes para el sistema de votación que otros métodos . La necesidad de volver a enviar las transacciones para refundir los votos también se puede aprovechar para combinar el polvo y otras transacciones para reducir el conjunto total de UTXO.

Las votaciones realizadas por los delegados en asuntos relacionados con la red se manejan de manera diferente que los votos de los usuarios. Cuando un Delegado vota envía una transacción a una dirección de votación especial con datos adjuntos que indican lo que están votando y cuál es su voto. No es necesario enviar Flashcoin para votar, sin embargo, se aplican tarifas de transacción de red normales . El voto de un delegado puede provenir de cualquier dirección , pero los datos de votación deben estar firmados por la clave privada de ese delegado.

### Mecanismos de Minería

Los mineros son elegidos por los Delegados , y solo los Delegados Elegidos pueden convertirse en Mineros . Los delegados votan por los mineros con su peso de voto que se deriva de la estaca de monedas, los votos de los usuarios y la antigüedad, y distribuyen su peso de voto sobre tantos o tan pocos mineros como quieran. Los primeros 25 mineros, calificados por el peso acumulado del voto en las boletas, se convierten en los mineros elegidos. Si un minero elegido es identificado como un mal actor, o como un nodo poco confiable, los delegados pueden actualizar sus votos y eliminar ese minero malo en 1 bloque.

Los mineros deben mantener un reloj del sistema preciso y tener una conexión de red de baja latencia y alto ancho de banda para poder participar de manera efectiva como mineros. El sistema está sintonizado para generar un bloque cada 5 segundos; sin embargo, cuando la red está inactiva, no es necesario emitir bloques. La inclinación del reloj en la marca de tiempo del bloque tendrá una tolerancia máxima de +4 segundos en el futuro, y ningún bloque puede tener una marca de tiempo antes que el bloque anterior . Cuando un minero crea un bloque , colocan su ID de delegado en la transacción de coinbase con fines de identificación , junto con una firma para demostrar que son ellos.

No existe una recompensa en bloque para que los mineros puedan reclamar , no hay tarifas de transacción por bloque para reclamar y, por lo tanto, no hay incentivos para minar los bloques lo más rápido posible . La dificultad de bloque se fija en un valor muy bajo para que no exista una carrera de armamentos con poder hash; cualquier CPU moderna puede bloquear un bloque en menos de un segundo.

Para asegurar el blockchain se aplican reglas estrictas para controlar el orden en el que los mineros elegidos pueden crear bloques . A cada minero elegido se le asigna una única ventana de 5 segundos en la que puede crear un bloque, si se necesita un bloque, y luego el siguiente minero de la lista obtiene su propia ventana de 5 segundos. Esta asignación ordenada del intervalo de tiempo previene los conflictos sobre quién va a extraer el siguiente bloque , reduciendo así la cantidad de ancho de banda y el procesamiento que generalmente se desperdicia en las redes con alto índice de bloqueo a medida que las cadenas en conflicto pasan y se evalúan. Esta asignación de intervalo de tiempo también garantiza que cada minero elegido tenga una oportunidad justa de crear un bloque.

Para asignar intervalos de tiempo a Miners , se usará una función similar a la siguiente función de JavaScript:

```
function canItMine(minerPosition, timestamp){ return ( (timestamp % 25) == minerPosition); }
```

Hay una lista de mineros elegidos actualmente que mantiene cada nodo, y se actualiza a medida que los mineros votan hacia adentro y hacia afuera. La "posición" de un minero es el índice de la matriz Elected Miner que corresponde a ese minero. Cuando un minero comprueba si puede extraer minas en este momento, pasa su propia posición a la función "canItMine" junto con la hora actual del sistema en formato Unix. Cuando un nodo evalúa la corrección de un bloque antes de aceptarlo, pasa la posición del Minero que extrajo ese bloque en esta función junto con la marca de tiempo de ese bloque.

Debido a que cada creación de bloque afirma que el creador cree que la cadena de bloques antes de que sea verdadera y precisa, cada bloque extraído es efectivamente un voto del Minero sobre cuál es la cadena actual correcta. En cualquier momento dado hay una Altura de consenso, una altura de bloque en la que se ha alcanzado el consenso en > 50% de los mineros electos, y esta altura de consenso aumenta a medida que se extraen nuevos bloques. Para cualquier bloque B dado, si > 50% de los mineros en el conjunto de mineros elegidos han creado un bloque en la cadena por encima de B, se ha logrado un consenso para B y se garantiza que todas las transacciones en y antes del bloque B serán definitivas. La Altura del consenso, entonces, está en B.

Todas las transacciones incluidas en o por debajo de la Altura del consenso se consideran definitivas, y todas las transacciones superiores a la Altura del consenso están en proceso de finalización. Las "confirmaciones" ya no son una medida de cuántos bloques se han construido encima del bloque de una transacción determinada, sino que "confirmaciones" es una medida de cuántos mineros electos han extraído un bloque encima del bloque de esa transacción, dividido por cuántos Los mineros son necesarios para lograr el consenso. Por lo tanto, las "confirmaciones" son efectivamente el porcentaje de consenso alcanzado por una transacción o bloque determinado. Dada una transacción T que se incluyó en el bloque B, si se construyeron 20 bloques encima de B por 10 mineros únicos del grupo de mineros electos de 25 mineros, entonces T y todas las demás transacciones en B tienen un porcentaje de consenso de:

$$\begin{aligned} 10 \text{ (miners' blocks)} / 25 \text{ (in pool)} / 0.5 \text{ (for 50\% consensus)} &= 0.8 \text{ consensus} \\ &= 80 \text{ 'confirmations'} \end{aligned}$$

Una vez que una transacción ha alcanzado o ha superado las 100 "confirmaciones", es definitiva y no hay riesgo de que se retrotraiga en un doble gasto . Todos los cálculos y lecturas de "confirmaciones" tendrán un límite de 100 para reducirla confusión.

Mientras un atacante no pueda comprometer más del 50% del conjunto de Miner Elegido , el doble gasto de una transacción dada es imposible una vez que la Altura de Consenso haya alcanzado o excedido el bloque de esa transacción . Si todo el conjunto de mineros elegidos permanece en línea y los bloques de minería , las transacciones se finalizarán en aproximadamente 65 segundos ( $> 50\% * 25 \text{ mineros} * 5 \text{ segundos bloques} = 65$ ). Todas las transacciones válidas se incluirán en bloques dentro de 10 segundos ( bloques de 5 segundos  $* 2 = 10$ ), y si todas las partes involucradas son confiables, dicha transacción puede ser procesable por el destinatario con confirmación por un solo Miner (5s promedio, máximo 10s) Para la finalidad garantizada al tratar con las partes que no son de confianza , el destinatario debe esperar 100 "confirmaciones" , lo que debería llevar unos 65 segundos. A modo de comparación, los sistemas puros de prueba de trabajo y prueba de estaca nunca proporcionan una finalidad de transacción garantizada sin puntos de control centralizados.

### **Resolución de bifurcación de cadena de bloques**

Como se describió anteriormente , hay una Altura de consenso (CH) que aumenta a medida que se extraen nuevos bloques . Ningún nodo acepta una horquilla blockchain que vaya más allá del CH actual ; el CH es efectivamente un punto de control dinámico votado por los Mineros . Cuando hay dos tenedores que compiten por encima del CH, gana el tenedor con la participación más única de los mineros . Si, por ejemplo , el tenedor F1 tiene 10 cuadras de largo, pero solo 2 mineros únicos hicieron esos 10 bloques , mientras que el tenedor F2 tiene 8 cuadras de largo con cada bloque siendo minados por un único minero (8 mineros únicos ) luego gana F2. Si ambas bifurcaciones tienen la misma cantidad de mineros únicos , entonces la bifurcación más larga gana . Si un minero elegido está evaluando dos tenedores y ese minero puede extraer un nuevo bloque en cualquiera de las horquillas, lo hará antes de comparar las horquillas . En la sincronización inicial de Blockchain , un nodo malicioso con una cadena de bloques no válida puede envenenar un nodo, y la regla que establece que reorgs no puede ser más profunda que el CH actual puede evitar que el nodo encuentre alguna vez la verdadera cadena de bloques . Si esto sucede, entonces el nodo tendrá que conectarse primero solo a un nodo en la cadena de bloques correcta , y luego de que encuentre un CH verdadero , entonces el nodo puede conectarse abiertamente a cualquier par. Los votos por y para los Delegados solo se cuentan una vez que la transacción que los emitió ha alcanzado 100 confirmaciones (consenso completo ), con la excepción de la selección del Minero . Cuando un Delegado cambia su voto para los Mineros , esos votos se cuentan instantáneamente y, si se cambia el grupo de Mineros Elegidos como resultado , ese cambio se aplica al siguiente bloque.

### **Red inactiva**

En los primeros años de cualquier red criptomonedas , puede haber largos períodos de actividad de transacción cero . A medida que crece la adopción de la capa 2, también disminuirá el nivel de actividad en la cadena requerida para las redes de soporte de la capa 2. Para evitar la creación sin sentido de bloques vacíos o inútiles, la nueva red FLASH admite el funcionamiento en vacío. Todos los incentivos para mineros para minar bloques lo más rápido posible se han eliminado , y nChainWork ya no existe, por lo que la red puede simplemente dejar de crear bloques cuando ya no los necesite . Una gran cantidad de ancho de banda y espacio de índice se guarda al ralentí , y también mantiene la sincronización de blockchain lo más eficiente posible . Para que la red permanezca inactiva sin afectar negativamente a los usuarios , los mineros deben continuar extrayendo bloques a la tasa normal de 5 segundos hasta que no haya transacciones por encima de la altura del consenso.

Si la red está inactiva y luego se transmite una sola transacción, los mineros comenzarán a extraer los bloques inmediatamente , y continuarán durante aproximadamente 13 bloques hasta que la transacción haya alcanzado el consenso total, y luego los mineros estarán inactivos nuevamente hasta la próxima vez que transacción válida es transmitida.

## Mecánica de Recompensas para Mineros y Delegados

Todos los días se iniciará con una transacción que distribuirá ganancias a Mineros y Delegados. Las ganancias se derivan de las tarifas de transacción de red: todas las tarifas del último día se suman, se dividen a la mitad, de modo que el 50% se destina a Miners y el 50% a Delegados , y luego se asignan como se describe a continuación.

El 50% para los Delegados se asigna en función del peso de voto de cada Delegado en el momento de la creación de la recompensa tx. Cuanto mayor es el peso del voto de un delegado, mayor es su parte de la recompensa.

El 50% para mineros se asigna por igual a mineros elegidos.

## Recompensas de la Fundación

La Fundación del Tercer Milenio se compromete a donar 32,000 FLASH a los Mineros y Delegados todos los días en forma de tarifas de transacción . Esto se hará mediante un proceso automatizado para enviar una transacción por día desde la billetera de la Fundación FLASH con una tarifa de transacción de 32,000. Estas donaciones continuarán hasta el SOME\_EXPIRATION\_DATE.

## Órdenes de votación relacionadas con la votación

El sistema de votación requiere el envío de metadatos de votación firmados en la cadena de bloques para que los vean todos los usuarios . Con el fin de mantener la compatibilidad con versiones anteriores de todas las herramientas creadas para Bitcoin, estos metadatos se envían utilizando el código de operación OP\_RETURN estándar de Bitcoin en el script de una salida de valor cero de tx. Las herramientas de Bitcoin no analizan los datos de OP\_RETURN, ni se almacenan en la memoria, solo en el disco. FLASH implementa varios códigos de operación de votación nuevos que solo se utilizan después de un opcode OP\_RETURN , y estos nuevos códigos de operación le dicen al analizador cómo interpretar los metadatos proporcionados. Lista de nuevos códigos de operación:

- **OP\_REG\_DG:** Delegar actualizaciones de registro e información
- **OP\_STAKE\_DG:** Proporcionar participación adicional para delegar.
- **OP\_VOTE\_DG:** Voto de usuario para delegar.
- **OP\_DG\_VOTE:** Delegar voto en un asunto relacionado con la red.
- **OP\_DG\_SIGN:** Usado en coinbase tx para probar la identidad del creador del bloque.

Estos nuevos códigos de operación deben ser utilizado en un script txout de valor cero inmediatamente después de un opcode OP\_RETURN , y este resultado de valor cero debe ser el primer txout (posición 0) en una transacción . Solo el primer txout se analiza para opcodes de votación, y solo si ese primer txout comienza con OP\_RETURN.

### Dirección especial de elección

Esta es una dirección FLASH válida que tiene reglas especiales para recibir y enviar transacciones.

Las monedas que se replantean para los Delegados se envían a esta dirección, al igual que la información de registro del Delegado. Todas las transacciones dentro y fuera de esta dirección deben pagar las tarifas de transacción de red normales. Nadie conocerá nunca la clave privada para esta dirección, e incluso si se derivara de alguna manera, la red solo permite que las monedas controladas por esta dirección se devuelvan a la dirección de devolución especificada o al remitente original, para que no puedan ser robados.

**Reglas de recepción:**

- Puede haber una o muchas entradas de tx. Si hay más de una entrada, se debe especificar una dirección de retorno en los metadatos de votación.
- El primer OP\_REG\_DG para un ID de delegado determinado debe enviar al menos 1,000,000 de FLASH a esta dirección . Cualquier participación adicional debe ser de al menos 100,000 FLASH.
- Si hay una salida tx, debe ser una operación OP\_REG\_DG que esté actualizando un Delegate existente • Si hay dos salidas tx, la segunda debe enviar las monedas replanteadas a la dirección especial de elección y la primera salida debe ser una operación OP\_STAKE\_DG agregar a la participación del delegado existente o una operación OP\_REG\_DG que está registrando un nuevo delegado. Los metadatos específicos de la operación se proporcionan después del código de operación.

**Reglas de envío:**

- Cualquiera puede gastar estas monedas sin la dirección especial de la elección 'privke'.
- Todas las transacciones recibidas por esta dirección tienen un ID de delegado para el que están siendo estacadas. Ese delegadoID no debe ser un delegado electo actualmente; las monedas apostadas para el Delegado solo pueden reclamarse si ese Delegado no ganó la elección o si se ha completado su retiro de la oficina.
- Para cada entrada de tx gastada, la dirección de salida debe coincidir con la dirección de retorno en los metadatos correspondientes de la entrada, o si no se proporciona, la dirección de salida debe coincidir con la dirección de envío original (devolver al remitente).

**Dirección especial de votación**

Esta es una dirección FLASH válida que tiene reglas especiales para recibir transacciones y rechaza todas las transacciones de envío . Cuando un Delegado emite un voto , envía una transacción a esta dirección con los datos de voto y pagan una tarifa de transacción por enviar el tx . Nunca se deposita ninguna moneda en esta dirección , simplemente se usa para facilitar la contabilización de los votos. El monitoreo de esta dirección monitoreará efectivamente todos los votos de los Delegados.

**Reglas de recepción:**

- Solo puede haber una salida tx que use el código de operación OP\_RETURN seguido del código de operación OP\_DG\_VOTE y los datos.
- Los datos OP\_DG\_VOTE deben validar con éxito.

**Reglas de envío:**

- No se pueden gastar transacciones enviadas a esta dirección.

El código de operación OP\_DG\_VOTE se explica en una sección posterior , al igual que los detalles relacionados con la forma en que un Delegado emite los votos y cómo se cuentan los votos.

**Formulario de registro de elección**

Para registrarse para elección como delegado debe enviarse una transacción con al menos 1,000,000 de FLASH a la dirección de elección especial, junto con un OP\_RETURN txout con el código de operación OP\_REG\_DG que incluye datos JSON serializados codificados en hexadecimal como:

```
{  
    delegatePubkey: [full pubkey],  
    infoVersion: [number],  
    displayName: [string],  
    enabled: [boolean],
```

```

mining: [boolean],
auditURL: [URL],
contacts: [ // optional
{
    type: [email/chat/IRC/URL/whatever],
    address: [address]
},
{
    type: [email/chat/IRC/URL/whatever],
    address: [address]
}
],
website: [URL] // optional,
registeredTime: [int], // timestamp of first ever registration
}

```

El documento JSON de registro debe ser inferior a 4 KB cuando está codificado. Un ejemplo de la secuencia de comandos en un txout [0] que está registrando un delegado:

```
OP_RETURN OP_REG_DG [regData] [delegateSig]
```

Un delegado puede actualizar este formulario en cualquier momento simplemente proporcionando un nuevo formulario de registro con los datos actualizados y un campo incrementado 'infoVersion'. No se requiere ninguna participación adicional para las actualizaciones de este formulario, siempre y cuando ya se haya estacado la cantidad suficiente para que este Delegado cumpla con el requisito mínimo de 1,000,000 de FLASH.

#### Explicación de los campos:

- **delegatePubkey:** esta es la publicación completa que identifica al delegado. Todas las acciones futuras tomadas por este Delegado se firmarán con la clave privada correspondiente de este pubkey.
- **infoVersion:** este es un número que se incrementa con cada actualización del formulario de registro de este Delegado para garantizar el estado correcto. Sin esto, las demoras en la replicación de tx o las reorgs de cadena pueden hacer que una versión anterior sobrescriba una versión más nueva.
- **displayName:** pretende ser una cadena legible para humanos que aparecerá en varias interfaces de usuario para identificar a este delegado, además del delegateID.
- **enabled:** si se establece en verdadero, este ID de delegado será elegible para elección. Si se establece en falso, entonces no será. Cuando un delegado electo desea retirarse de sus deberes, debe establecerlo en falso y esperar el final del término de elección actual, y luego puede reclamar su moneda apostada
- **mining:** cuando se establece en verdadero , este delegado indica que quiere ser minero , por lo que puede ser elegido como minero . Si se establece en falso , este Delegado no es elegible . Si este delegado ya es un minero elegido, puede establecer este campo en falso para retirarse de la minería. Si se registra como un Minero potencial, se debe replantear un mínimo de 2,000,000 FLASH.
- **auditURL:** esta es la URL que expone la interfaz de auditoría de este Delegado. Hay un nuevo comando de API que, cuando se consulta, devuelve información que se puede usar para probar que este nodo está activo y para medir ciertas características de rendimiento. Se recomienda utilizar un servicio como memcached al hospedar esta URL para proteger al nodo de consultas excesivas. La auditoría se explica con más detalle en una sección posterior de este documento.
- **contacts :** esta es una matriz JSON que contiene una lista de objetos de contacto . Cada objeto especifica el tipo de dirección de contacto que se proporciona y la dirección en sí. Esto tiene la intención de proporcionar una forma para que la comunidad se comunique con el propietario de este Delegado. Esto es opcional, sin embargo, es poco probable que la comunidad vote por un delegado que no saben nada.

- **website:** es una URL que enlaza con un sitio web que representa a este delegado o a la organización propietaria del delegado. Esto es opcional, sin embargo, es poco probable que la comunidad vote por un Delegado del que no saben nada.
- **delegateSig:** este formulario de registro se puede enviar desde cualquier dirección, no tiene que ser directamente del Delegado, sin embargo, el objeto JSON 'regData' debe estar firmado con la clave privada correspondiente al delegatePubkey, y esa firma debe colocarse en este campo. Cuando se reciben los datos de registro, se genera el delegateID convirtiendo el delegatePubkey en una dirección FLASH.

Cuando se reciben los datos de registro, se genera el delegateID convirtiendo el delegatePubkey en una dirección FLASH.

### Aposta por los delegados

Cualquier dirección puede contribuir con una participación a cualquier Delegado, sin embargo, no se pueden apostar monedas para ningún Delegado que no se haya registrado, y la primera inscripción debe incluir al menos una participación de 1,000,000 de FLASH. La primera participación de registro debe incluirse con la transacción de registro, pero se pueden proporcionar contribuciones adicionales enviando las monedas a la dirección especial de la elección con el código de operación OP\_STAKE\_DG. La transacción de replanteo debe tener dos salidas; la segunda debe enviarse a la dirección de elección especial (mínimo de 100,000 FLASH), y la primera salida debe tener valor cero y usar el opcode OP\_RETURN seguido por el opcode OP\_STAKE\_DG seguido del delegateID siendo apostado, y opcionalmente seguido por una dirección de devolución para los fondos replanteados. Si no se especifica una dirección de devolución, los fondos solo se pueden devolver al remitente.

Ejemplo:

```
OP_RETURN OP_STAKE_DG [delegateID] [returnAddress]
```

Debe tenerse en cuenta que las monedas apostadas no podrán devolverse hasta que el Delegado para el que están siendo apostadas se haya retirado de la oficina y haya terminado su período actual de elección. Colocar monedas para un Delegado es ceder el control de ellas durante el tiempo que ese Delegado se postule para las elecciones y cumplir sus mandatos.

### Votación para delegados

Los usuarios normales votan por delegados anteponiendo un valor de salida tx de valor cero a su transacción que utiliza el código de operación OP\_RETURN seguido del código de operación OP\_VOTE\_DG, seguido de la lista de ID de delegado para votar con cada salida de transacción. Los votos emitidos por las entradas de tx se destruyen y los votos se crean con las nuevas salidas de tx. El opcode OP\_VOTE\_DG espera una lista simple de delegateID's, uno para cada txout (sin contar el primer txout que contiene estos metadatos) y la totalidad de la cantidad de monedas de salida de tx se usa para votar por el Delegate descrito por este conjunto. Ejemplo:

```
OP_RETURN OP_VOTE_DG [delegateID1] [delegateID2] [delegateID3]
```

El primer ID delegado en la lista OP\_VOTE\_DG recibe el voto de txout [1], el segundo ID delegado en la lista recibe el voto de txout [2], y así sucesivamente. Si hay más txouts que delegateID, entonces los txouts que faltan un correspondiente delegateID simplemente no se cuentan en el voto. Si un delegateID proporcionado en la lista OP\_VOTE\_DG no es válido, entonces se rechaza toda la transacción.

### Votación por los delegados

Los delegados votan enviando una transacción a la dirección especial de votación. El uso de esta dirección facilita la explicación de cómo y cuándo votan los delegados, y esta dirección tiene reglas especiales, tal como se describe en la sección de Dirección de Votación Especial más arriba.

La transacción que emite el voto lo hace al tener una salida tx única seguida por el opcode OP\_DG\_VOTE y los datos que describen el voto.

Los votos se describen utilizando un objeto JSON que se pasa al código de operación OP\_DG\_VOTE. Cada clave en el objeto voteData representa un artículo de votación, y cada valor indica el voto en sí. Cada valor en el objeto voteData JSON tiene un límite de tamaño máximo de 2Kb y el objeto voteData completo tiene un límite de tamaño máximo de 100 Kb. Dado el suministro de monedas de 900M y la participación mínima de registro de delegados de 1M, nunca puede haber más de 900 Delegados en ejecución, y por lo tanto la variable mapDelegateState que contiene información de votación nunca puede ser de más de aproximadamente 90Mb. En la práctica, debería ser un orden de magnitud más pequeño.

El código de operación OP\_DG\_VOTE espera tres argumentos:

```
OP_RETURN OP_DG_VOTE [delegateID] [voteData] [voteSig]
```

El argumento 'voteSig' es una firma del argumento 'voteData', firmado con la clave privada que corresponde al delegateID. El argumento 'voteData' es un documento JSON serializado codificado hexadecimal de la siguiente manera:

```
{
    voteHeight: 12527,
    minTxFee: 30000000, // satoshis
    txFeePerByte: 110, // satoshis
    miners: [
        delegateID3: 50, // 50% of vote-weight
        delegateID14: 25, // 25% of vote-weight
        delegateID31: 25, // 25% of vote-weight
    ]
}
```

#### Explicación de las claves y valores:

- **key:** la clave en este objeto JSON representa el artículo de la boleta para la cual se está emitiendo este voto. Cada vez que se envía un tx para actualizar el VoteData, la actualización se considera incremental: se aplica sobre el VoteData existente para ese Delegado, sobrescribiendo cualquier valor existente para claves dadas. Al establecer el valor de una tecla en 'nulo', se eliminará la clave y el valor, y se retirará el voto.
- **value:** este es el valor del voto emitido. El valor no puede tener más de 2 KB de tamaño. Los nuevos valores para claves preexistentes sobrescribirán el valor anterior.

Cualquier transacción que use el opcode OP\_DG\_VOTE con un documento JSON que difiera del formato descrito anteriormente, o que viole cualquiera de las restricciones descritas, se considera no válido y se descarta. Las limitaciones en el tamaño de los datos de votación previenen los ataques de DoS y spam, y la estructura general del documento de votación permite una gran flexibilidad para futuras papeletas. Las transacciones OP\_DG\_VOTE deben pagar la misma tarifa de transacción por byte que cualquier otro tx en la red.

La clave especial 'voteHeight' en el voteData se incrementa cada vez que un delegado actualiza su voto. Esto permite que la red aplique estas actualizaciones de votos a mapDelegateState en el orden correcto. Si un Delegado intenta proporcionar una actualización con voteHeight 123, por ejemplo, sin proporcionar previamente la actualización con altura 122, entonces la actualización de 123 se rechaza. Si se recibe una segunda actualización de 123, se rechaza. Si un txid diferente que dice ser la actualización 123 se incluye en un bloque después una reorganización, luego, la

primera actualización 123 se invierte y se aplica la segunda actualización 123 ; los datos de mapDelegateState deben mantenerse sincronizados con los votos validados en blockchain..

### Contabilidad de votos e indexación

Este sistema de votación se ha diseñado cuidadosamente para que sea lo más eficiente posible, para agregar la menor sobrecarga posible a los requisitos de procesamiento de bloques y nodos, a fin de facilitar la mayor tasa de transacciones por segundo posible. Además de las consideraciones de rendimiento, es ideal para mantener una compatibilidad total con todas las herramientas relacionadas con Bitcoin, por lo que este sistema de votación ha sido diseñado para mantener esa compatibilidad hacia atrás.

Cada voto emitido, cada apuesta realizada y cada registro de delegado se realiza a través de una salida tx en la primera posición de salida que utiliza el código de operación OP\_RETURN seguido de un nuevo código de operación de votación que es exclusivo de FLASH, seguido de los metadatos de votación. Cualquier intérprete de guiones basado en Bitcoin no analizará los datos siguiendo el opcode OP\_RETURN, pero el analizador de guiones FLASH continuará leyendo el guión si hay un código de operación de votación inmediatamente después del opcode OP\_RETURN. Exigir que estos metadatos de votación estén en la primera posición de salida de tx (txout [0]) garantiza búsquedas eficientes al descubrir votos antiguos que están siendo destruidos por nuevas salidas de tx. También aumenta el análisis del script OP\_RETURN omitiendo el análisis de cualquier información OP\_RETURN que no esté en la posición txout [0].

Para facilitar la contabilidad y la indexación en el código central , se crearán nuevas variables y estructuras::

- ‘Delegate’ class: contiene información sobre un Delegado y métodos para sondear e interactuar con los Delegados.
- ‘mapDelegateState’: este es un mapa delegateID-indexed de objetos delegados
- ‘VoteState’: esto contiene el conjunto de reglas actual, el resultado de la votación , derivado de mapDelegateState.
- ‘PendingVote ’ struct: es una estructura que contiene información sobre un voto que está pendiente de un consenso total antes de ser contado . Los atributos incluyen blockheight , txid y datos de voto interpretados.
- ‘mapPendingVotes ’: Este es un mapa indexado txid de estructuras PendingVote , que se utiliza para acceder a PendingVotes por txid. Si se elimina un tx debido a la reorganización de la cadena , entonces ese tx se elimina de este mapa , así como del multimap mencionado a continuación.
- ‘mmPendingVotesByBlock ’: este es un multimap de vectores bloqueados de txid's. Cuando se actualiza la altura de consenso , este multimap se escanea para encontrar todos los votos pendientes que ahora están completamente confirmados y luego se cuentan sus votos . Una vez que se cuentan todas las transacciones en un vector descubierto , todos los txid correspondientes se eliminan de mapPendingVotes y luego el vector se elimina de este mmPendingVotesByBlock.

Dado un suministro máximo de monedas de 900M y una cantidad mínima de participación de registro delegada de 1M, nunca habrá más de 900 entradas en la variable mapDelegateState . Cuando un Delegado recupera su participación, su entrada en mapDelegateState será eliminada.

Al recibir una transacción válida de registro de delegado , cada nodo verificará si este nuevo delegadoID ya existe en mapDelegateState , y agregará una nueva estructura allí si no existe . Al recibir actualizaciones de registro válidas, cada nodo sobrescribirá la entrada del ID del delegado en mapDelegateState con la nueva estructura si la nueva estructura tiene un número de información de información más alto que la entrada existente.

A medida que se acepta cada bloque nuevo en la cadena de bloques, se comprueba cada transacción en el bloque para ver los metadatos en txout [0], y si está presente , se validará y anotará en mapPendingVotes y mmPendingVotesByBlock . Los votos no se cuentan hasta que la Altura de consenso haya alcanzado o superado la altura de bloqueo de esa transacción . Cada tx tiene entradas y salidas; cualquier voto emitido previamente por las entradas se debe sustraer de mapDelegateState y se debe agregar cualquier voto emitido por los resultados.

Cada vez que cambia la altura del consenso , lo que puede suceder con cada nuevo bloque , se reitera mmPendingVotesByBlock y cada entrada con una altura de bloque igual o inferior a la nueva altura de consenso se procesa y elimina tanto del multimap como del mapa. Procesar mmPendingVotesByBlock significa obtener cada transacción en el vector que se está procesando, analizar los metadatos de votación y actualizar mapDelegateState para contar los nuevos votos. Si se cambia mapDelegateState después de procesar el nuevo cambio de altura de consenso , todos los votos en mapDelegateState se vuelven a contar y la estructura global de VoteState se actualiza para indexar el conjunto de reglas activo actual. Los futuros bloques serán validados contra este conjunto de reglas actualizado. De esta forma, se mantiene un total acumulado para todos los votos en mapDelegateState , actualizado con cada cambio de altura de consenso, y el resultado de la votación se indexa en structVoteState para que todos los parámetros puedan buscarse de manera fácil y eficiente.

Nota: los cambios en el voto de un minero de un delegado se procesan instantáneamente cuando son recibidos por un nodo , pero todos los demás cambios deben esperar la finalización completa al permitir que la Altura de consenso alcance el bloqueo de ese voto.

### Cálculo de resultados del voto

Las claves del objeto voteData son arbitrarias ; sin embargo , hay varias claves que se utilizarán de manera predeterminada cuando se inicie la red FLASH. Otras llaves serán ignoradas. Cada clave en voteData puede usar un método diferente para determinar el resultado de la votación , y ese método está codificado en el código de gobierno de FLASH . Aquí hay explicaciones del método de votación para cada una de las claves voteData compatibles en el lanzamiento:

- **voteHeight**: Este es un número que se incrementa cada vez que un delegado actualiza su voto. La ordenación secuencial de las actualizaciones de votos garantiza que las actualizaciones se apliquen a mapDelegateState en el orden correcto.
- **minTxFee** : Esta es la tarifa de transacción mínima permitida para cualquier tx, denominada en satoshis . El resultado de la votación es la mediana ponderada de todos los votos de los delegados , ponderados con su voteWeight.
- **txFeePerB** : Es la tarifa de tx por byte del tamaño de la transacción , denominada en satoshis por byte . El resultado de la votación es la mediana ponderada de todos los votos de los delegados , ponderados con su voteWeight.
- **miners**: El resultado de esta votación es una selección de los 25 mejores delegados del Minero, ordenados por voto acumulativo de todos los delegados . La contribución del voto por voto de un delegado es el voto total del delegado multiplicado por el porcentaje que el delegado le otorgó a este minero.

### Firmas de transacciones y delegadas de Coinbase

Cada bloque extraído tendrá una transacción de coinbase con 0 entradas , similar a la mayoría de las criptomonedas derivadas de Bitcoin, sin embargo, la transacción de coinbase de FLASH tendrá un valor cero txout [ 0] que utiliza los códigos de operación OP\_RETURN y OP\_DG\_SIGN de la siguiente manera:

OP\_RETURN OP\_DG\_SIGN [delegateID] [delegateSigOfBlockhash]

Este txout [0] es prueba de que el delegateID dado fue el creador del bloque.

Una vez al día, la transacción de coinbase contendrá productos adicionales para distribuir las recompensas del día anterior a Mineros y Delegados. Estas recompensas se distribuirán como se describe en la sección anterior "Mecánica de recompensas de mineros y delegados". A medida que avanza el día y se pagan las tarifas de tx, esas tarifas se queman, se eliminan temporalmente del suministro de monedas . El primer bloque de cada día recrea esas monedas quemadas y las distribuye a los Mineros y Delegados.

## Interfaz de auditoría y URL

Habrá un nuevo comando API llamado **getauditpoint** que puede ser utilizado por la comunidad para verificar que este delegado esté en línea con una cadena de bloques que esté sincronizada. Este comando no toma argumentos. Cuando los usuarios eligen un Delegado (s) para votar desde una lista, esa lista debe mostrar el estado de sincronización de cada Delegado para que el usuario tenga la información que le permita votar. La respuesta de la llamada a la API de getauditpoint seguirá el formato API de Bitcoin estándar y podría verse así:

```
{
  error: null,
  result: {
    auditPoint: {
      delegateID: [pubkey],
      blockHeight: [chain-tip height],
      blockHash: [chain-tip blockhash],
      mempoolSz: [number],
      timestamp: [current time]
    },
    delegateSig: [base64 signature with privkey]
  },
  id: 0
}
```

Cada llamada de esta API generará un nuevo mensaje firmado con la información de fecha y hora actual del nodo y la información de la cadena. Los delegados que actualmente son elegidos o se postulan para una elección deben hacer que esta información sea públicamente accesible. Se recomienda utilizar un proxy de almacenamiento en caché por separado delante de esta llamada a la API para protegerse contra ataques DoS y de spam, y para ocultar la verdadera IP del delegado del público, si así lo desea.

## Interfaz de información del delegado

Habrá un nuevo comando API llamado **getdelegateinfo** que devuelve un objeto JSON que revela información sobre cada delegado, tanto elegido como en ejecución. Los resultados de este comando API se usarán para las pantallas del tablero en los sitios web de la comunidad y en la billetera Qt. Este comando tiene un argumento opcional: **delegateID**. Especificar un ID de delegado filtrará los resultados para devolver únicamente el estado de votación para el Delegado especificado; de lo contrario, se devolverán los datos de todos los Delegados. Los datos proporcionados en la respuesta de este comando API se derivan de la variable **mapDelegateState**. Esta información se puede utilizar para ver información sobre todos los delegados, incluso qué delegados votan en qué dirección y con cuánta influencia. Ejemplo:

```
{
  error: null,
  result: {
    delegateCount: 326,
    votesCast: 235486723.12345678,
    totalVotesPossible: 900000000.00000000,
    totalVoteWeight: 789437432323.12345678,
    delegates: {
      "Sojgf3w40FSj9fw92jgFSmbZAdqT2": { // key is delegateID
        infoVersion: 123,
        displayName: "This Delegate",
        enabled: true,
        mining: true,
        auditURL: "http://1.2.3.4:80/audit",
        contacts: [
          {
            type: "email",
            address: "contact@delegate1.com"
          }
        ],
        website: "http://delegate1.com",
        registeredTime: 1567234635883,
        lastDGVoteTime: 164264529832,
        votesRcvd: 1324523,
        stake: 2000000,
        timeInOffice: 1356542,
        voteWeight: 5673561,
        voteData: {
          minTxFee: 33000000, // satoshis
          txFeePerByte: 100, // satoshis per byte
          miners: [
            delegateID3: 50, // 50% of vote-weight
            delegateID14: 25, // 25% of vote-weight
            delegateID31: 25, // 25% of vote-weight
          ]
        }
      },
      "USgs39sdVnkdpA30SDmOP353zvc4": { // key is delegateID
        infoVersion: 234,
        displayName: "Another Delegate",
        enabled: true,
        mining: true,
        auditURL: "http://2.3.4.5:80/audit",
        contacts: [
          {
            type: "email",
            address: "contact@delegate2.com"
          }
        ],
        website: "http://delegate2.com",
        registeredTime: 165234635812,
      }
    }
  }
}
```

```

lastDGVoteTime: 164264523825,
votesRcvd: 526272,
stake: 1000000,
timeInOffice: 356548,
voteWeight: 2673564,
voteData: {
    minTxFee: 33000000, // satoshis
    txFeePerByte: 100, // satoshis per byte
    miners: [
        delegateID4: 20, // 20% of vote-weight
        delegateID63: 65, // 65% of vote-weight
        delegateID17: 15, // 15% of vote-weight
    ]
},
},
},
id: 0
}

```

### Interfaz del estado de votación

Habrá un nuevo comando API llamado getvotestate que devuelve un objeto JSON que revela el estado actual de todos los elementos de la votación contabilizados en la variable structVoteState. Esta es la ley para el bloque actual. Ejemplo:

```

{
    error: null,
    result: {
        minTxFee: 33000000,
        txFeePerByte: 100,
        miners: { // map of elected miners
            delegateID4: true,
            delegateID63: true,
            delegateID17: true
        }
    },
    id: 0
}

```

Este comando API y structVoteState están destinados a mostrar únicamente el resultado de todos los votos a partir del cierre del bloque anterior, lo que hace que estos resultados sean el conjunto de reglas que se aplicará al siguiente bloque generado. Si se necesitan más detalles acerca de los pesos y los delegados, se debe usar el comando API getdelegateinfo en su lugar.

## Interfaz de usuario de votación.

Tanto la billetera Qt como la CLI proporcionarán interfaces para votar. Los monederos tendrán parámetros de configuración que se pueden configurar para controlar la votación automatizada con monedas en la billetera, así como las interfaces que muestran el estado actual del poder de voto de esta billetera. Dado que los votos se emiten a medida que se gastan las monedas, no como se retienen, la votación automatizada permitirá al usuario establecer un nivel de umbral para la cantidad de monedas en la billetera que no está votando según lo especificado, y cuando se excede dicho umbral, la automatización enviará esa moneda nuevamente a la billetera con el conjunto de preferencias de votación correctas. De manera predeterminada, la automatización de votación combinará todas las transacciones en una sola salida al reenviarse a sí mismo, a menos que el usuario haya optado por dejar de combinar tx.

## Mineros y estadísticas delegadas

Los datos pueden recopilarse tanto de los mineros como de los delegados de sus auditURL publicadas para supervisar qué tan bien sincronizados están. Todos los delegados y mineros deben sincronizar rápidamente la punta de la cadena actual y mantener aproximadamente el mismo conteo de mempool tx. AuditURL proporciona un mensaje firmado que muestra esta información de cada delegado y minero. Se pueden recopilar estadísticas adicionales de Miner desde el blockchain mismo, mediante el análisis de la transacción de coinbase para identificar qué mineros están participando correctamente. Los mineros elegidos nunca deben perder su intervalo de tiempo de generación de bloques a menos que la red esté inactiva, y si un minero pierde un porcentaje demasiado alto de sus ranuras de tiempo, entonces deben ser reemplazados por un minero más confiable. Al reunir la información necesaria y hacerla accesible a todos los usuarios, es posible que la comunidad tome decisiones informadas sobre las elecciones de delegados y mineros, y las decisiones bien informadas conducirán a una red ópticamente eficiente.

# FLASH Web Wallet - Estructura de cuenta + Generación, almacenamiento y recuperación de claves

LAS CUENTAS EN LA BILLETERA WEB DE FLASH SE ALMACENAN EN UN SERVIDOR DE AUTENTICACIÓN CENTRAL (CAS)..



Cada cuenta  
CAS tiene los  
siguientes datos:

**id**: account ID.

**email**: Cuenta de correo.

**role**: Según sea la autorización. Ejem.:  
USER o ADMIN

**privateKey**: clave privada de cifrado (cifrada  
por la contraseña del usuario)

**publicKey**: EC criptografía de la clave  
pública.

**sc1**: Procedimiento de recuperación del  
usuario (cifrada por las respuestas de  
seguridad del usuario).

**sc2**: Parte del servidor utilizada para la  
recuperación.

**sc3**: Participación del administrador en caso  
de que sc1 no pueda recuperarse.

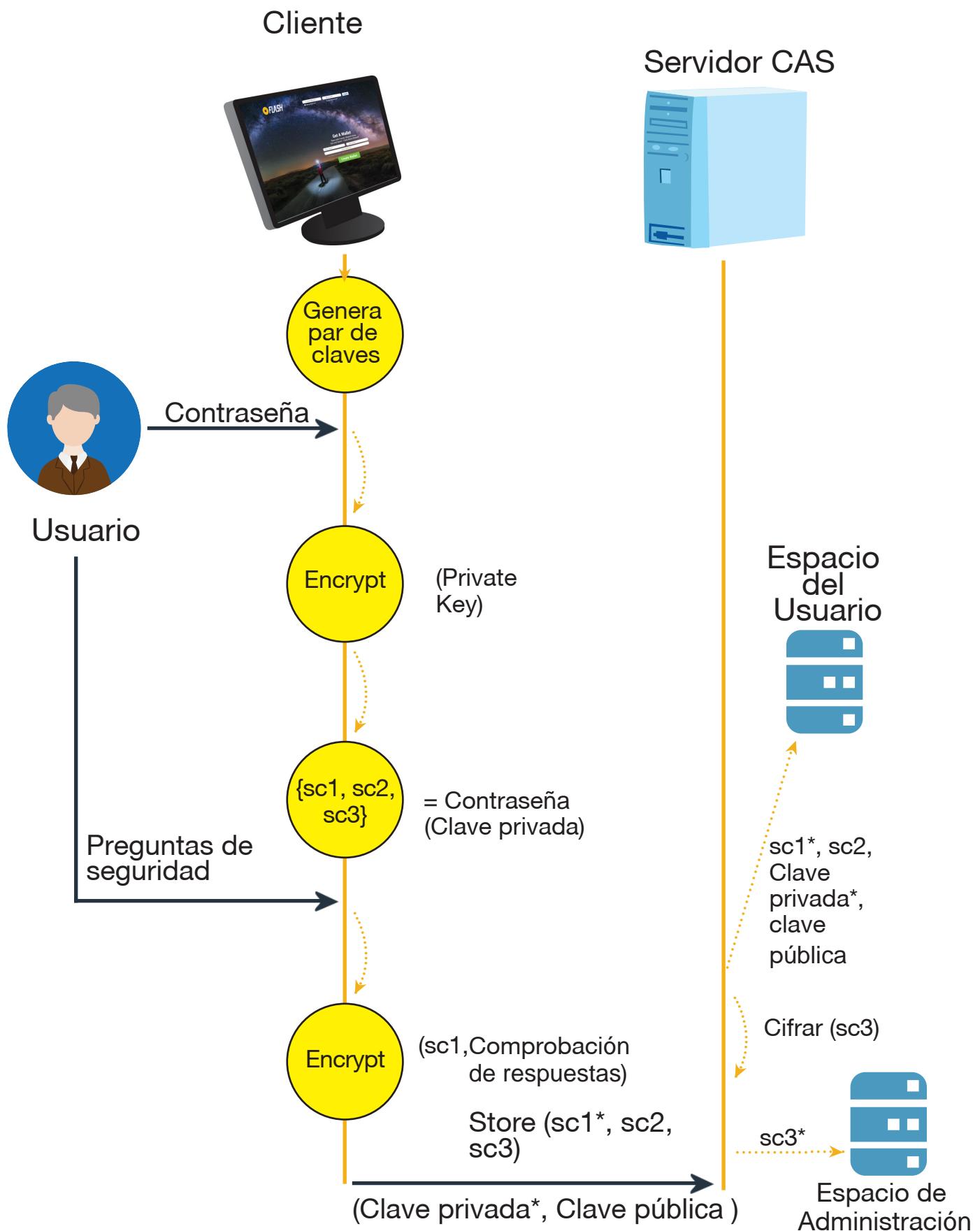
Además, los perfiles del usuario se almacenan en el  
servidor de claves Flashcoin. La información incluye:  
nombre para mostrar , avatar , país ... que varía de  
una aplicación a otra.

# FLASH Web Wallet

## Generación, almacenamiento y recuperación de llaves.

### Generación de claves al registrarse

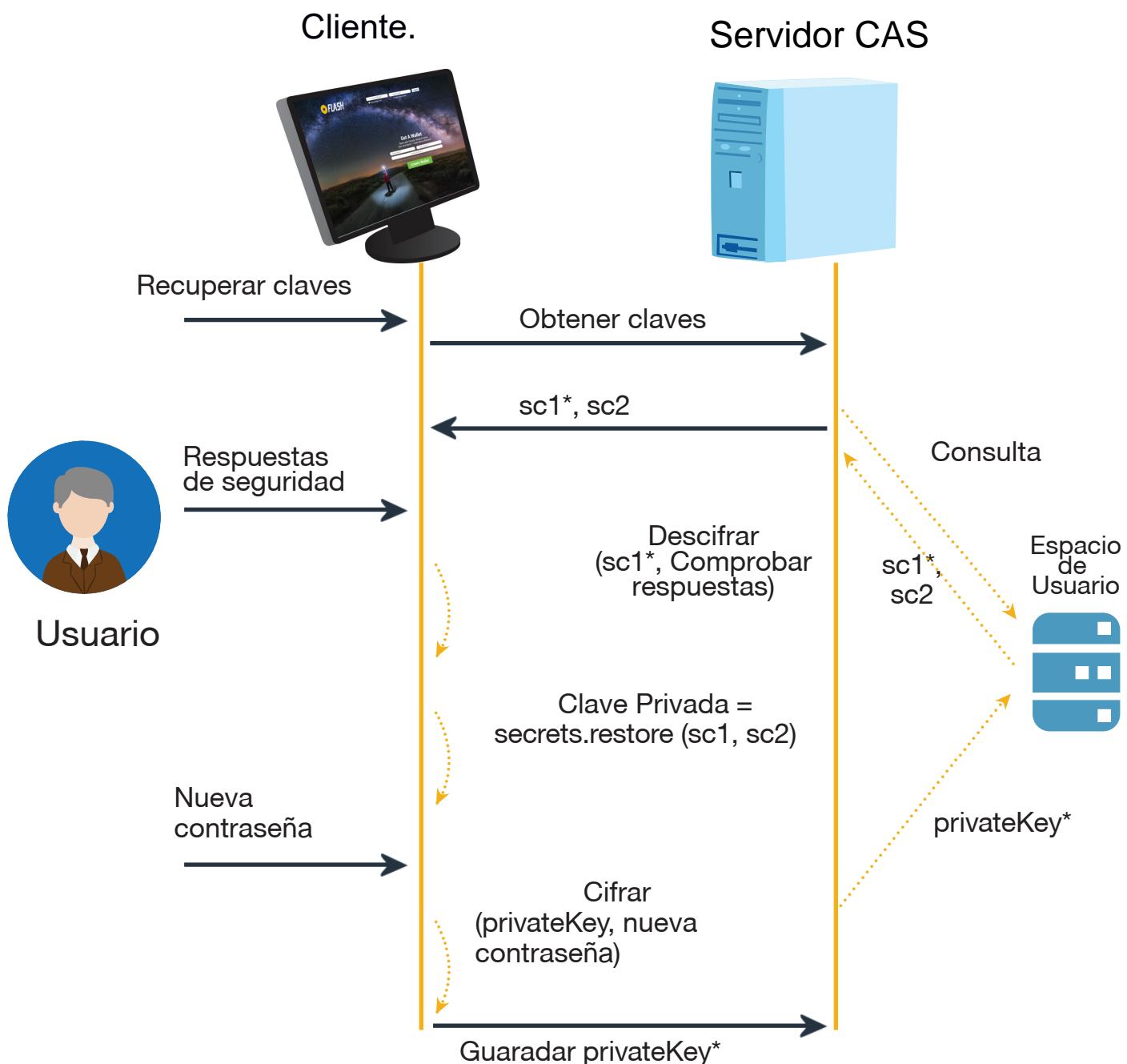
El par de claves de cifrado EC se genera en el lado del cliente cuando se registra . La clave privada luego se cifra con la contraseña del usuario. Las claves de recuperación también se generan a partir de la clave privada, que es una tupla (sc1, sc2 , sc3). Después de que el usuario responde las preguntas de seguridad , las respuestas se usan para cifrar sc1.



## Recuperación clave

El proceso de recuperación se activa automáticamente por el usuario . Después de que se verifica el correo electrónico , el cliente recibe sc1 encrypted y sc2 , preguntas de seguridad. Al responder correctamente a las preguntas de seguridad , la respuesta se utiliza para descifrar sc1. Desde sc1 y sc2, la clave privada se restaura. El usuario debe proporcionar una nueva contraseña e iniciar el proceso de protección y almacenamiento de las claves siguiendo el mismo procedimiento anterior

El usuario también puede elegir un modo súper seguro donde el servidor almacena un sc único. Cuando se activa el modo de recuperación , el usuario debe proporcionar su parte para combinar con el recurso compartido del servidor . Si el usuario pierde el sc1 (dado en el proceso de registro ), entonces nadie puede recuperar su contraseña . Por lo tanto , la participación obligatoria del usuario en el proceso de recuperación garantiza la seguridad del recurso compartido del usuario, así como la contraseña.



# FLASH Blockchain

FLASH HA MODIFICADO LA VERSIÓN LITECOIN DE BLOCKCHAIN PARA UTILIZAR COMO UN SISTEMA DE ALMACENAMIENTO DE RED DISTRIBUIDO. UN NÚMERO DE MODIFICACIONES SIGNIFICATIVAS AL CÓDIGO SE HAN HECHO PARA ENFRENTAR LAS DEBILIDADES DE LAS REDES TOTALMENTE DESCENTRALIZADAS:

- **Latencia de conexión** - La sincronización de bloques entre los nodos ralentiza drásticamente la velocidad de validación de la transacción (gasto doble). FLASH es una red distribuida con nodos de minería limitada por los Delegados que votan por los mineros. Debido a la cantidad delegada y limitada de nodos de minería, la latencia de la red debe ser inferior a 400 ms (ventana de tiempo de propagación).
- **Rendimiento** - Dos factores que determinan el rendimiento de blockchain incluyen la sincronización de bloques y la minería en bloque . FLASH utiliza servidores de caché e índice para sincronizar los nodos y restablecer el algoritmo de minería al menor factor de dificultad . Como tenemos la red confiable de nodos , no hay necesidad de seguir aumentando el grado de dificultad de la minería para el proceso de verificación de bloque . FLASH proporciona una solución única para garantizar la integridad de los datos de almacenamiento de red distribuida.
- **Riesgo de seguridad** - La minería en bloque es vulnerable con una red blockchain distribuida totalmente abierta a través del 51 % y otros ataques . El FLASH Blockchain no está abierto al público para extraer o manipular el blockchain por ventaja computacional . Esta tarea es mantenida por Mineros seleccionados por los Delegados Electos.

## Encriptado de fin a fin

Para garantizar que no exista ningún punto débil, el diseño del sistema de seguridad y de todas las funciones de cifrado deben realizarse desde el monedero web (nodo cliente). Como resultado, las transacciones se cifran con la clave pública del destinatario, que entonces son escritas en la cadena de bloques FLASH. Esta metodología protege de los intrusos que obtienen datos encriptados en la base de datos FLASH. Para que cualquier atacante o intruso pueda descifrar información, debe comprometer el sistema y descifrar los algoritmos de criptografía de curva elíptica (ECC) que se utilizan en cada tecla. Incluso si alguien tuviera una computadora cuántica y pudiera descifrar ECC, el costo de descifrar una transacción superaría con creces la posible ganancia. Para una computadora promedio, tomaría más de 100 mil millones de años en esfuerzo de computación, según la mayoría de los expertos. Por lo tanto, el costo de descifrar el ECC en cada transacción supera con creces el rendimiento potencial.

## Blockchain API

Un protocolo que permite a la aplicación web comunicarse con la red FLASH BlockChain. Todas las transacciones se han indexado en la capa API de la cadena de bloques para calcular previamente y acelerar las búsquedas de transacciones, como la verificación de doble gasto y los registros de transacciones.

# APÉNDICE

## Wallet Webservice API

### Crear cuenta

Name: create\_unverified\_account

Description: Create unverified account (need to verify via email)

Request params: name, email, ip, callbackLink, g\_recaptcha\_response  
(Google recaptcha response)

Response: {rc: Number}

### Establecer contraseña y verificar correo electrónico

Name: set\_password

Description:

Request params: password, privateKey (encrypted private key),  
publicKey, token

Response: {rc: Number}

### Obtener token de sesión (sso)

Name: get\_session\_token

Description:

Request params: idToken, resource

Response: { rc: Number, profile : Object { sessionToken: String } }

## Comprobación Token de sesión (sso)

Name: check\_session\_token

Description:

Request params: sessionToken, resource

Response: {rc: Number, profile: Object{username: String, email: String} }

## Login (sso)

Name:

Description:

Request params: email, password, ip, resource

Response:

Success: {rc: Number, profile: Object{email: String, display\_name: String, gender:

String, ...} }

## Actualizar cuenta

Name: update\_account

Description: Update user profile

Request params: display\_name, gender, profile\_pic\_url, about, timezone ...

Response: {rc: Number }

## Obtener Perfil

Name: get\_profile

Description: Get user profile

Request params: {}

Response: {rc: Number, profile: {username: String, email: String, display\_name: String, profile\_pic\_url: String ...} }

## Fijar PIN

Name: set\_pin

Description: Set PIN

Request params: pin

Response:

Success: {rc: Number}

## Check PIN

Name: check\_pin

Description: Check if PIN is correct

Request params: pin

Response:

Success: {rc: Number}

## Comprobar PIN

Name: change\_pin

Description: Change the PIN

Request params: old\_pin, new\_pin

Response:

Success: {rc: Number}

## Obtener detalles de contacto

Name: get\_contact\_detail\_by\_email

Description: Get contact details by email

Request params: contact\_email

Response:

Success: {rc: Number, profile: {username: String, email: String, display\_name: String, gender: String, profile\_pic\_url: String, ...} }

## Obtener Perfil

Name: get\_profile

Description: Get user profile

Request params: {}

Response: {rc: Number, profile: {username: String, email: String, display\_name: String, profile\_pic\_url: String ...} }

## Obtener usuarios

Name: get\_users\_by\_uid

Description: Get users information by user id

Request params: ['user1', 'user2', ...]

Response:

Success: {rc: Number, accounts: [account1, account2, ...] }

## Obtener lista

Name: ros\_get

Description: Get contact list of a user

Request params: {}

Response:

Success: {rc: Number, roster: {total\_subs: Number, subs: [], ...}}

## Operación de lista

Name: ros\_op

Description: operate roster, where operation could be REQUEST, APPROVE, REMOVE

Request params: op, from, to

Response:

Success: {rc: Number}

Notification: notify to related users

## Crear billetera

Name: create\_flash\_wallet

Description: Create a new wallet

Request params: idToken, wallet\_secret

Response:

Success: {rc: Number, wallet: {passphrase: String, wallet\_id: String, address: String }}

## Buscar Monedero

Name: search\_wallet

Description: Search for wallet by keyword, to send money to

Request params: start, size, term

Response:

Success: {rc: Number, criteria, wallets: [wallet1, wallet2, ..], total\_wallets: Number }

## Obtener mis carteras

Name: get\_my\_wallets

Description: Get my wallets (currently only support 1 wallet)

Request params: {}

Response:

Success: {rc: Number, my\_wallets: [], total\_wallets: Number}

## Agregar transacción

Name: add\_txn

Description: Push transaction to blockchain and add transaction log

Request params: receiver\_id, amount, currency\_type, receiver\_public\_address, transaction\_id, memo, request\_id, transaction\_hex (signed) Response:

Success: {rc: Number, id: String}

Notification: notify to the recipient about the new transaction

## Obtener registro de transacciones

Name: get\_txns

Description: Get transaction log of current user

Request params: date\_from, date\_to, order, start, size

Response:

Success: {rc: Number, txns: [tx1, tx2, ...], total\_txns: Number}

## Obtener registro de transacciones por Id

Name: get\_transaction\_by\_id

Description: Get transaction detail by id

Request params: transaction\_id

Response:

Success: {rc: Number, txn: {...} }

## Crear una transacción sin firmar

Name: create\_unsigned\_raw\_txn

Description: Create a unsigned transaction to be signed by the owner later

Request params: from\_address, to\_address, amount

Response:

Success {rc: Number, transaction: {...} }

## Obtener detalles de la transacción

Name: get\_transaction\_details

Description: Get transaction details from blockchain

Request params: transaction\_id

Response:

Success: {rc: Number, transaction: {...} }

## Obtener saldo

Name: get\_balance

Description: Get wallet balance from blockchain api

Request params: {}

Response:

Success {rc: Number, balance: Number}

## Agregar solicitud de dinero

Name: add\_money\_request

Description:

Request params: to, amount, note

Response:

Success: {rc: Number, id: String }

Notification: notify to the requested user

## Obtenga solicitudes de dinero

Name: get\_requests

Description:

Request params: date\_from, date\_to, status, start, size, type

Response:

Success: {rc: Number, money\_requests: [req1, req2, ...], total\_money\_reqs: Number}

## Marcar la solicitud de dinero como aceptada

Name: mark\_accepted\_money\_requests

Description:

Request params: receiver\_id, request\_id, note\_processing

Response:

Success {rc: Number}

## Marcar solicitud de dinero como rechazado

Name: mark\_rejected\_money\_requests

Description:

Request params: receiver\_id, request\_id, note\_processing

Response

Success {rc: Number}

## Marcar solicitud de dinero como cancelado

Name: mark\_cancelled\_money\_requests

Description:

Request params: sender\_id, request\_id, note\_processing

Response

Success {rc: Number}

## Marcar solicitud de dinero como leído

Name: mark\_read\_money\_requests

Description:

Request params: receiver\_id, request\_ids: Array<{request\_id, sender\_bare\_uid}> Response

Success {rc: Number}

# Blockchain APIs (en progreso)

## Empujar la transacción a la cadena de bloques

Name: push\_transaction

Description: push a transaction raw format (hexa encoding) to the blockchain

Request params: transaction hex

Response: {}

## Enviar token

Name: send\_token

Description: send token (coin) to a wallet identified by public address

Request params: to\_public\_address, amount, message

Response: {}