

2019



#PrivacyMatters

TABLA DE CONTENIDOS

3 Resumen

4 Introducción

6 Privacidad de Zcash
7 Servicios de la red DAPP

8 El Protocolo

9 Billeteras privadas
10 UTXOs privadas
12 Leyenda del diagrama de flujo
13 Creación de UTXOs fungibles y no fungibles
15 Quema de UTXOs fungibles y no fungibles
17 Conectando los dos mundos
18 Transferencia de UTXOs fungibles y no fungibles
20 Creación de UTXOs de autentificación
21 Quema de UTXOs de autentificación
22 Depósitos y retiros privados

27 Transacciones privadas sin cuenta de EOS

30 El Proxy
32 El Minero
33 Conclusión
34 Depósitos y retiros privados con la cuenta proxy

38 zAssets

42 Colateral
43 Creación y quema de zAssets
44 Reequilibrio del colateral
45 Retrasos y medias móviles
46 Eficiencia del capital
47 Riesgo

48 El Token

50 Airdrop

51 Organización inteligente



RESUMEN

ZEOS permite realizar transacciones privadas de activos fungibles y no-fungibles en la blockchain de EOS aprovechando la tecnología de privacidad de Zcash y los servicios de la red DAPP de LiquidApps. No es necesario modificar los contratos inteligentes de los tokens existentes. Los usuarios pueden mover libremente sus activos entre cuentas transparentes de EOS y billeteras privadas de ZEOS. Es imposible rastrear la propiedad de los activos en las billeteras privadas de ZEOS, porque se mantienen bajo la custodia del contrato inteligente de ZEOS. La propiedad está representada por UTXOS¹ imposibles de rastrear que pueden transferirse de forma privada entre las billeteras de ZEOS. Las UTXOs se pueden canjear en cualquier momento para recuperar el activo subyacente en una cuenta de EOS. A parte de las transferencias privadas, ZEOS ofrecerá una interfaz fácil de implementar para todos los contratos inteligentes existentes en EOS para hacer posible depósitos y retiros privados. Esto permite que todas las aplicaciones en EOS se conviertan en privadas por defecto, protegiendo la privacidad de sus usuarios y siendo totalmente transparente a nivel de contrato inteligente para facilitar la auditoría. Para desvincular completamente las transacciones privadas de las cuentas personales de EOS que los usuarios tengan, se introduce el concepto de cuenta proxy. Esta cuenta tiene un permiso especial "público" que puede ser utilizado por cualquier persona para llevar a cabo transacciones privadas mediante el pago de una cuota de transacción denominada en tokens ZEOS. Los llamados "mineros" compiten por esas tasas alimentando la cuenta proxy con recursos de EOS. Por último, el token ZEOS impulsa un protocolo DeFi de capital eficiente para los "zAssets" sintéticos. Todo el ecosistema de ZEOS se rige por una organización inteligente².

1 Unspent transaction output = Salida de la transacción no gastada

2 [Explicación](#) de Daniel Larimer

INTRODUCCIÓN

El mercado de las finanzas descentralizadas sin permisos, las llamadas "DeFi", está creciendo de forma masiva. Sólo en los últimos dos años, el valor total bloqueado en los protocolos DeFi pasó de unos 600 millones de dólares en 2020 a más de 230.000 millones en 2022. Los cripto-entusiastas son muy conscientes de que la DeFi acabará sustituyendo a la mayoría de los elementos del sistema financiero tradicional.

Sin embargo, lo que actualmente casi todas las blockchains y aplicaciones descentralizadas existentes carecen por completo es uno de los deseos humanos más básicos cuando hablamos de asuntos financieros: la protección de la privacidad del usuario. Esto es sobre todo una realidad para las blockchains de contratos inteligentes como EOS.

Si bien es necesario que los contratos inteligentes de casi todas las aplicaciones funcionen con total transparencia incluyendo todos los activos que entran y salen, al menos la identidad de los usuarios que interactúan con ellos debe ser anónima. Cuando hablamos de transacciones entre usuarios, ningún dato sensible de la transacción, es decir, las direcciones del remitente o del destinatario, el tipo de activo o la cantidad, para nada deben exponerse públicamente.

El estado actual de DeFi es una absoluta pesadilla desde el punto de vista de la privacidad. Sin embargo, la blockchain y DeFi tienen el potencial de ofrecer una privacidad aún mayor para sus participantes que la que podría ofrecer el sistema financiero tradicional. Después de todo, en el mundo financiero actual nadie es realmente privado: Incluso sin la vigilancia del gobierno a través de la regulación, siempre hay un broker, un "intermediario" que facilita el acceso de los individuos a los mercados financieros. En definitiva, esto significa que siempre hay alguien que "vigila".

Pero con las criptomonedas, los particulares pueden acceder a los mercados directamente, sin intermediarios. Esto permite por primera vez a los usuarios permanecer en el anonimato cuando actúan en los mercados financieros. Esta es otra de las ventajas potenciales de DeFi sobre las finanzas tradicionales, que aún no ha sido reconocida por la comunidad cripto. La verdadera privacidad de los usuarios cambiaría las reglas del juego de DeFi y supondría un importante logro en términos de madurez para toda la industria.

La privacidad del usuario no sólo es deseable, sino absolutamente crucial para que DeFi tenga éxito.

PRIVACIDAD DE ZCASH

Cuando hablamos de privacidad en la blockchain, las verificaciones de conocimiento cero han demostrado ser una herramienta poderosa, en particular las "zk-SNARK" (Zero Knowledge Succinct Non Interactive Argument of Knowledge). El equipo de desarrollo de Zcash lidera esta innovación creando y manteniendo la base del código abierto más avanzado y optimizado para zk-SNARK. Su código se utiliza ampliamente en muchos proyectos de criptomonedas y es revisado por muchos equipos de desarrolladores. ZEOS utiliza exactamente la misma base de código, específicamente el sistema de verificación zk-SNARK para permitir las transacciones privadas en EOS.



Justo antes de que se escribiera este whitepaper, Zcash tuvo su quinta mayor actualización de red llamada "Orchard". Con ella se introdujo un nuevo sistema de verificación SNARK llamado "Halo 2". Entre otras mejoras, el nuevo sistema de verificación ya no requiere la llamada "trusted setup". La trusted setup es un acto en el que los miembros de confianza de la comunidad realizan un cálculo multipartito para generar de forma segura un conjunto aleatorio de parámetros para cada circuito aritmético de SNARK. A partir de estos parámetros se obtienen las claves de comprobación y verificación. Estos son los pares de claves que se requieren para posteriormente generar y verificar las verificaciones

de conocimiento cero.

A pesar de que si se realiza correctamente una "trusted setup" puede considerarse muy segura, ya que sólo uno de los participantes tiene que ser honesto, siempre ha sido una fuente de FUD (miedo, incertidumbre y duda). Sin embargo, con el nuevo sistema de verificación se utiliza un método de aritmética diferente para crear los pares de claves de comprobación/verificación, eliminando la necesidad de generar parámetros fiables y, por tanto, todo el FUD relacionado con la "trusted setup".

ZEOS adapta el nuevo sistema de verificación de Zcash "Halo 2" proporcionando una tecnología vanguardista para todas las aplicaciones del ecosistema EOS.



ZcashSapling

Nota: El "verificador SNARK" de la primera temporada fue desarrollado y desplegado en la red de pruebas Kylin para la aplicación de demostración en ZEOS (prueba del concepto) antes de que se escribiera este whitepaper. Todavía utiliza el sistema de verificación 'Groth16' de Zcash 'Sapling' y será actualizado al nuevo sistema de verificación 'Halo 2' antes de ser desplegado en la red principal de EOS.

Halo 2

SERVICIOS DE RED DAPP

Cuando hablamos de desarrollar dApps en EOS, los servicios de red DAPP son una gran adición al ecosistema que permiten

escalabilidad ilimitada y los menores costes de transacción posibles para el usuario final. Se utilizan los siguientes servicios:



VRAM

Para construir un modelo de transacción privada UTXO basado en zk-SNARKs dentro de un contrato inteligente EOS, es necesario mantener tres estructuras de datos crecientes en la blockchain. Estas estructuras de datos crecen con todos y cada uno de los elementos de "sólo-lectura" para el contrato inteligente. Por desgracia, la memoria RAM de la blockchain de EOS es costosa y poco adecuada para estos grandes conjuntos de datos y en su mayoría de sólo-lectura. Por otro lado, la VRAM es perfectamente adecuada para esta tarea. Es una gran solución para un almacenamiento barato e infinito que es directamente accesible para los contratos inteligentes en EOS. Dado que todos los fondos mantenidos de forma privada utilizando ZEOS se almacenan básicamente en VRAM, cabe destacar que todo el estado de la VRAM siempre se puede restaurar volviendo a poner la blockchain de EOS. Además, es imposible manipular los datos en la VRAM, ya que esto sería detectable en la blockchain. Estas características hacen que la VRAM sea realmente fiable.



VCPU

Para la verificación de la prueba de conocimiento cero (ZKP), ZEOS utiliza el servicio VCPU como parte de 'LiquidHarmony'. Esto podría ser sólo temporal, ya que la verificación de la prueba debería ocurrir idealmente en la blockchain. Sin embargo, en la preparación de este documento, la verificación ZKP en blockchain basada en "Groth16" ha sido probada y evaluada en la red de pruebas Kylin: Una verificación en blockchain de una simple prueba de conocimiento de una semilla hash de 32 bytes tarda alrededor de 150ms usando una implementación en C++ del verificador "Saplings" de Zcash vinculado a un contrato inteligente EOS. Desafortunadamente, esto excede el límite de ejecución de 30ms de las transacciones de EOS por multitudes, sin mencionar los costes resultantes de un tiempo de ejecución tan largo. Sin embargo, la verificación total o al menos parcial en la blockchain podría ser posible en el futuro. Se podrían añadir ciertos intrínsecos al EOSVM para mejorar la aritmética de la curva elíptica de la verificación ZKP, o quizás incluso una función intrínseca para la verificación ZKP completa. Esto podría eliminar la dependencia de la VCPU en futuras versiones del protocolo. Aunque la utilización del VCPU hace que las transacciones privadas sean relativamente baratas: El coste de una transacción privada completa es sólo de unos 2ms de CPU (Kylin testnet). Sin el VCPU, ZEOS no podría ser desarrollado en este momento.



Oráculos

Para extraer datos de los precios del mundo real desde fuera del ecosistema EOS se utiliza el servicio de oráculos de la red DAPP como parte de "LiquidHarmony". Los oráculos de precios son necesarios para crear y quemar zAssets.

Este documento sigue siendo un "trabajo en curso" y no un whitepaper definitivo. Habrá más iteraciones. Los comentarios de la comunidad son bienvenidos y agradecemos las contribuciones que puedan surgir.

EL PROTOCOLO

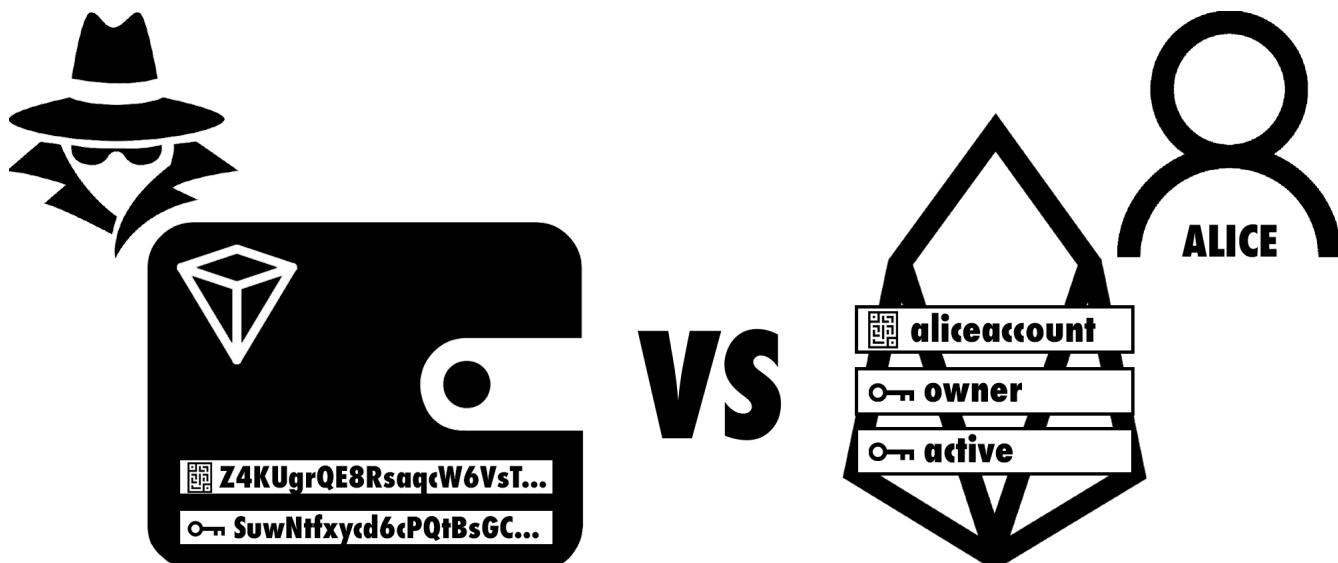
ZEOS se inspira en el protocolo Nightfall para las transferencias privadas de activos fungibles y no fungibles en la blockchain de Ethereum. Aunque en su versión original el protocolo es muy limitado. ZEOS amplía el concepto de Nightfall añadiendo algunas características útiles de Zcash. Un ejemplo es la "distribución secreta en banda" de Zcash, que elimina la dependencia de Nightfall de un canal lateral seguro para la comunicación privada. Esto es necesario para compartir los datos secretos de las transacciones con los receptores de las mismas. ZEOS elimina la dependencia de un canal de comunicación adicional y se basa únicamente en la blockchain pública de EOS y en los servicios de red DAPP.

BILLETERAS PRIVADAS

ZEOS introduce un nuevo tipo de billetera en el que los activos de EOS pueden mantenerse con total privacidad. Las billeteras privadas de ZEOS existen en paralelo a las cuentas transparentes de EOS y los activos pueden moverse libremente entre ellas.

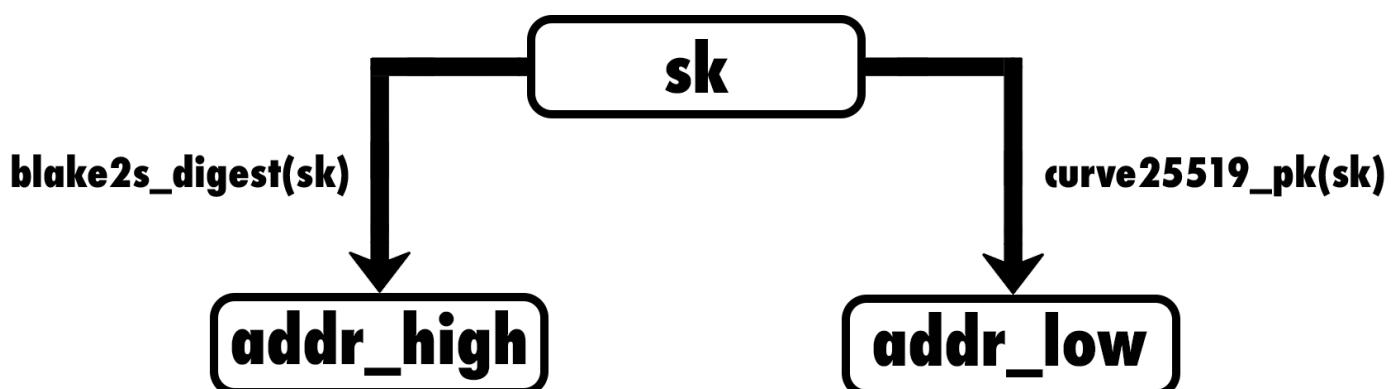
Las billeteras privadas de ZEOS son muy similares a las de las criptomonedas tradicionales como Bitcoin. Cada billetera se

define mediante una clave de gasto secreta (también conocida como clave privada). La dirección de recepción de una billetera se deriva de esa clave privada. Lo mismo ocurre con las billeteras de ZEOS. Crear una billetera significa básicamente coger un número aleatorio de 32bytes que representa la clave de gasto secreta, igual que en Bitcoin.



En el diseño actual, la dirección de ZEOS consta de dos partes: La clave pública correspondiente a la clave secreta de gasto (basada en curve25519) concatenando con

el valor hash de la clave secreta (blake2s digest). Esto resulta en direcciones ZEOS de 64 bytes de longitud.



Cada transacción privada de ZEOS viene con una clave de visualización única de 64 bytes que revela todos los detalles sobre la parte receptora de la transacción. Las claves de visualización pueden ser utilizad-

as por los remitentes para demostrar que se ha ejecutado una determinada transacción y que ciertos activos han sido recibidos por una billetera determinada.

UTXOS PRIVADAS

La propiedad de los activos ocultos en las billeteras de ZEOS se representa a través de la propiedad de UTXOs privadas. A parte de los tokens fungibles y no-fungibles, ZEOS introduce un tercer tipo de token que es intransferible y representa un *permiso privado*.

Este tipo de token de autentificación permite realizar depósitos y retiros privados de activos fungibles y no-fungibles desde y hacia cualquier contrato inteligente de terceros en el ecosistema de EOS.

Tipos de UTXO privados:



**TOKEN
FUNGIBLE**



**TOKEN
NO-FUNGIBLE**



**TOKEN DE
AUTENTICACIÓN**

Representa un activo fungible en EOS. Se caracteriza por la cantidad y el símbolo.

Representa un activo no fungible en EOS. Se caracteriza por un identificador único.

Representa un permiso para acceder a los activos en custodia de un determinado contrato inteligente. Se caracteriza por un hash asignado.

Todos los tipos de UTXO comparten la misma estructura de datos:

header: 64 bit

El campo del header contiene meta-information como el tipo de UTXO y un número que especifica bajo qué versión de protocolo ha sido creado.

d0: 64 bit

El primer campo de datos contiene el importe del activo concreto (token fungible) o los 64 bits inferiores de un identificador (token no fungible). En el caso del token de autenticación, este campo no está definido.

d1: 64 bit

El segundo campo de datos contiene el código del símbolo del activo concreto (token fungible) o los 64 bits superiores de un identificador (token no fungible). En el caso del token de autenticación, este campo no está definido.

contract: 64 bit

En el caso de los tokens fungibles y no fungibles, este campo contiene el nombre de la cuenta EOS del contrato inteligente del token emisor. En el caso del token de autenticación, este campo contiene el nombre de la cuenta EOS del contrato inteligente que emitió el permiso.

cool_down: 64 bit

Determina cuándo se puede "quemar" esta UTXO. En concreto, este valor es una altura de bloque EOS a partir de la cual este UTXO se convierte en "quemable". Un valor de enfriamiento permite varias características interesantes como los tiempos de espera de quema para los activos sintéticos o los permisos privados que sólo se vuelven válidos después de un punto particular en el tiempo (es decir, una altura de blockchain particular).

rho: 256 bit

Un número aleatorio de 32 bytes (también conocido como "salt") para garantizar valores hash únicos a las resoluciones de la UTXO (token fungible y no fungible). En el caso de los tokens de autenticación, este valor es como una clave secreta que da acceso a lo que el permiso fue creado.

La dirección de la billetera no se almacena por UTXO sino por cada transacción que crea la UTXO. Todas las transacciones privadas también incluyen un campo memo de 256-bytes para mensajes privados.

LEYENDA DIAGRAMA DE FLUJO



WORLD Ilustra lo que ocurre en la blockchain de EOS cuando una transacción privada es ejecutada. Muestra exactamente qué información sensible está expuesta y qué podría ver un "observador".



WORLD Ilustra el equivalente desde la perspectiva del mundo virtual de ZEOS. Muestra exactamente qué información sensible está expuesta y qué podría detectar un observador.



EOSACTION Una acción EOS iniciada por un determinado actor y/o ejecutado por una determinada cuenta y permiso de EOS.



Firma de un permiso de una cuenta EOS



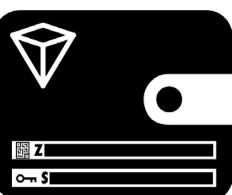
Zeos TX Data Datos de transacciones privadas de ZEOS que siempre contienen una prueba de conocimiento cero (ZKP) que demuestra la corrección de las entradas privadas (ocultas). Todas las entradas públicas están expuestas y son accesibles al público.



Firma de datos de las transacciones privadas, que es básicamente la ZKP misma: Si la prueba es válida, implica automáticamente que la transacción privada está firmada por la clave secreta de gasto correcta.



Una cuenta de EOS con un determinado nombre (dirección) y ciertos permisos. Si está representada con un "</>" en la parte superior, hay un contrato intelectual desplegado en esta cuenta. Si no hay un "</>", se trata de una cuenta personal.



Una billetera ZEOS con una dirección determinada y una clave de gasto secreta. El protocolo nunca expone información sobre las billeteras, por lo que siempre es privada.



Activos estándar de EOS fungibles o no-fungibles.



La salida de la transacción no gastada (UTXO) representa un activo fungible o no fungible en EOS, o un token de autenticación ZEOS.



Firmar o llamar a una acción EOS.



Movimiento de activos/UTXOs entre cuentas EOS/billeteras ZEOS



Creación de UTXO



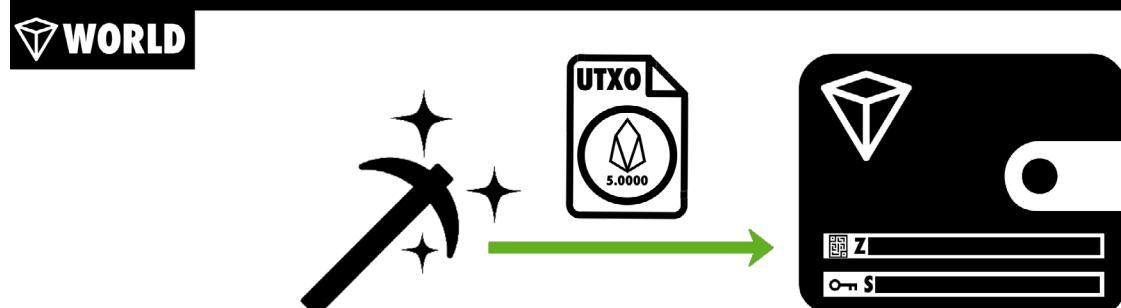
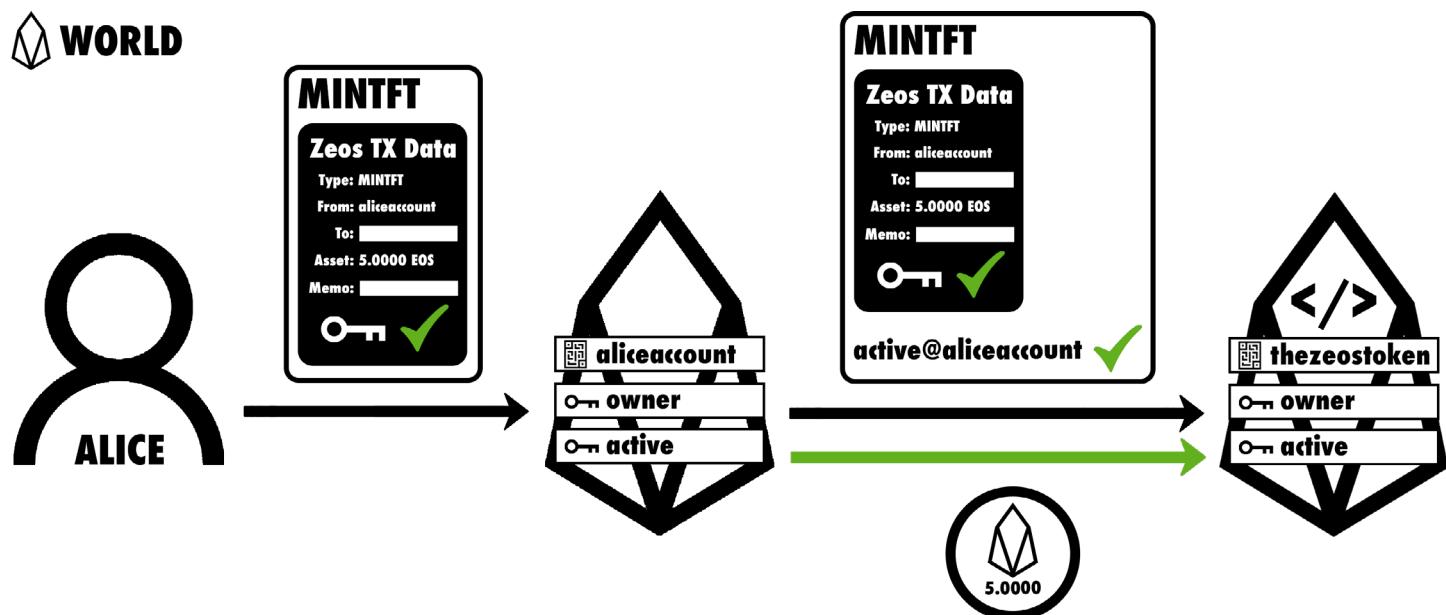
Quema de UTXO.

CREACIÓN DE UTXOS FUNGIBLES Y NO-FUNGIBLES

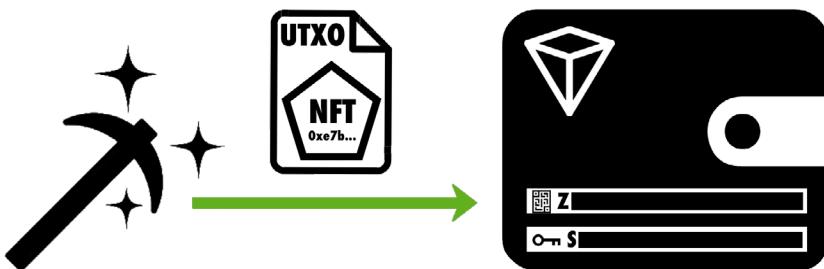
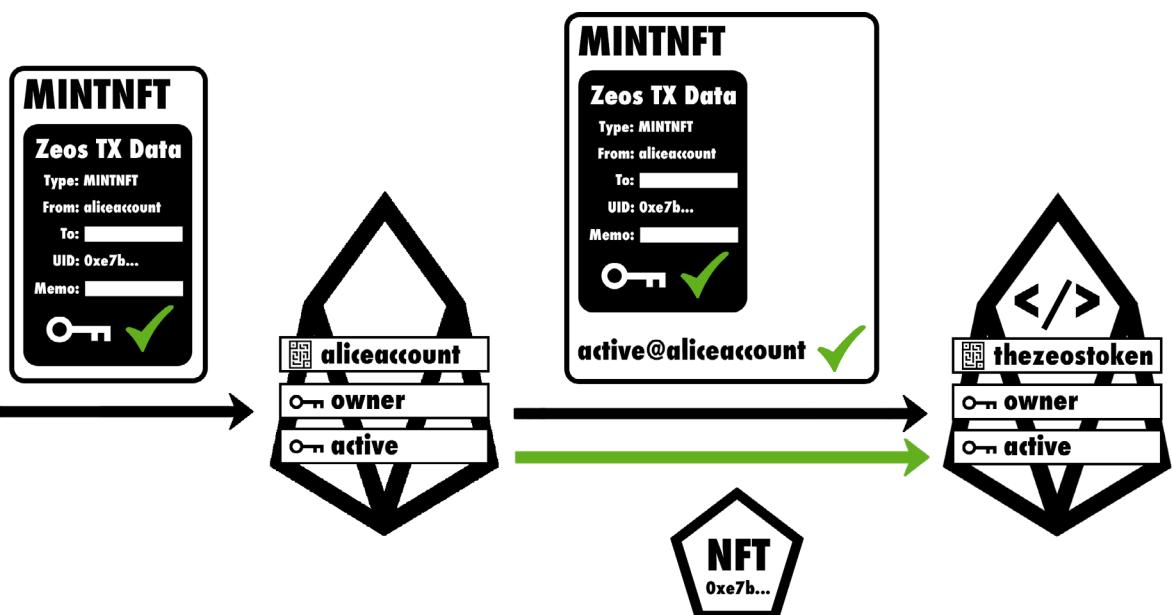
Para mover un activo de una cuenta transparente de EOS a una billetera privada de ZEOS se llama a la acción "crear"(mint). Para mover un activo desde una billetera ZEOS privada a una cuenta EOS transparente se llama a la acción "quemar"(burn). Para las transferencias de tokens totalmente privadas y no rastreables se llama a la acción "ztransfer". Estas tres acciones existen tanto para los tokens fungibles como para los no fungibles:

- 'mintft' y 'mintnft'
- 'burnft' y 'burnnnft'
- 'ztransferft' y 'ztransfernft'

Cuando se llama a "crear" en un token fungible o no-fungible, el activo indicado pasa de la cuenta EOS a ser custodiado por el contrato³ del token ZEOS. Al mismo tiempo, se crea una UTXO privada que representa el activo en la dirección de la billetera de ZEOS indicada. La información de la dirección de la billetera en la que se está creando la UTXO permanece privada. Sólo la cantidad y el símbolo del activo son públicamente rastreables cuando se crean UTXOs. De hecho, el protocolo no expondrá públicamente ninguna dirección de ZEOS.



MINTFT: ALICE MUEVE 5 TOKENS EOS DE SU CUENTA PÚBLICA EOS A UNA BILLETERA PRIVADA DE ZEOS



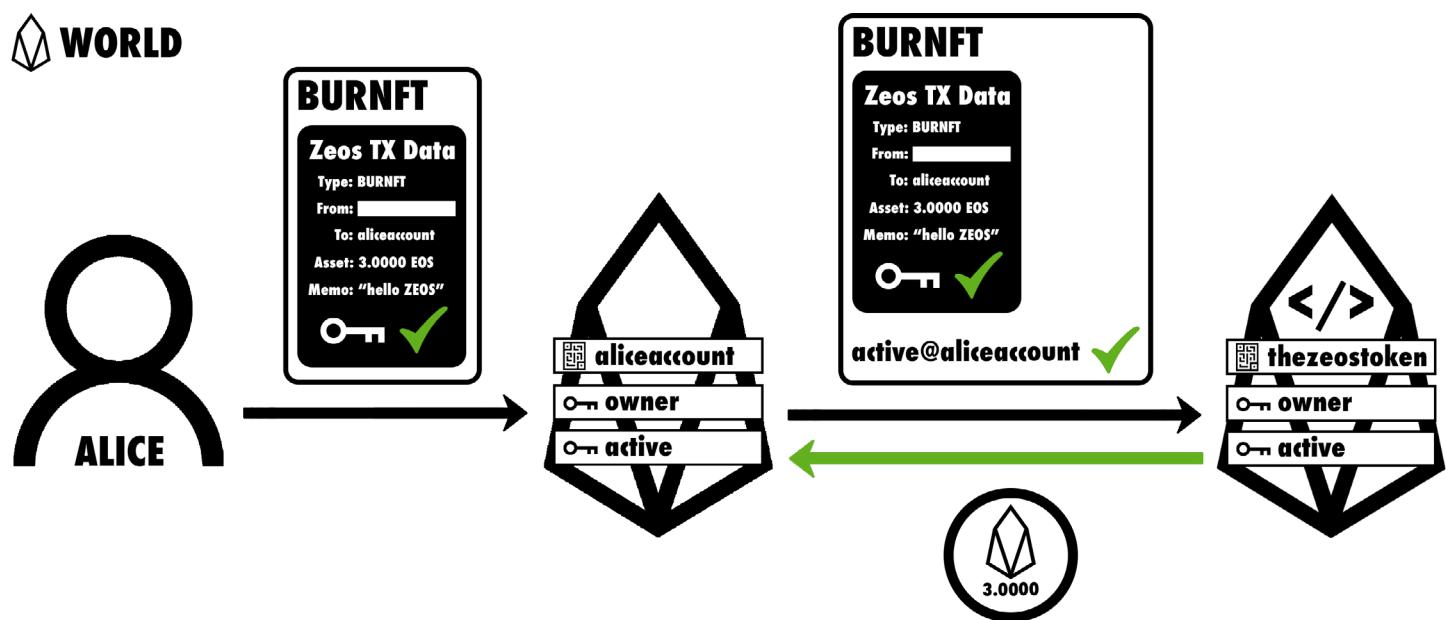
MINTNFT: ALICE MUEVE UN NFT DE SU CUENTA PÚBLICA EOS A UNA BILLETERA PRIVADA DE ZEOS

³ Cada llamada a 'mint'(crear) debe ir estrictamente precedida de una llamada a 'transfer' para mover el activo fungible o no fungible indicado de la cuenta EOS del usuario a la cuenta de 'thezeostoken' según el concepto de 'depósito/retirada' de los contratos inteligentes de EOSIO(Llamado ahora Antelope). Sin embargo, en aras de la simplicidad, esta acción de transferencia anterior se ignora en este documento.

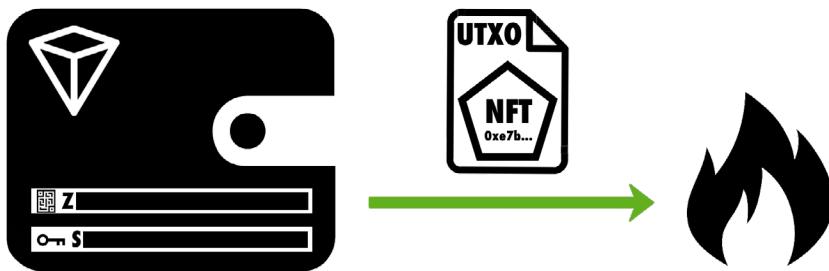
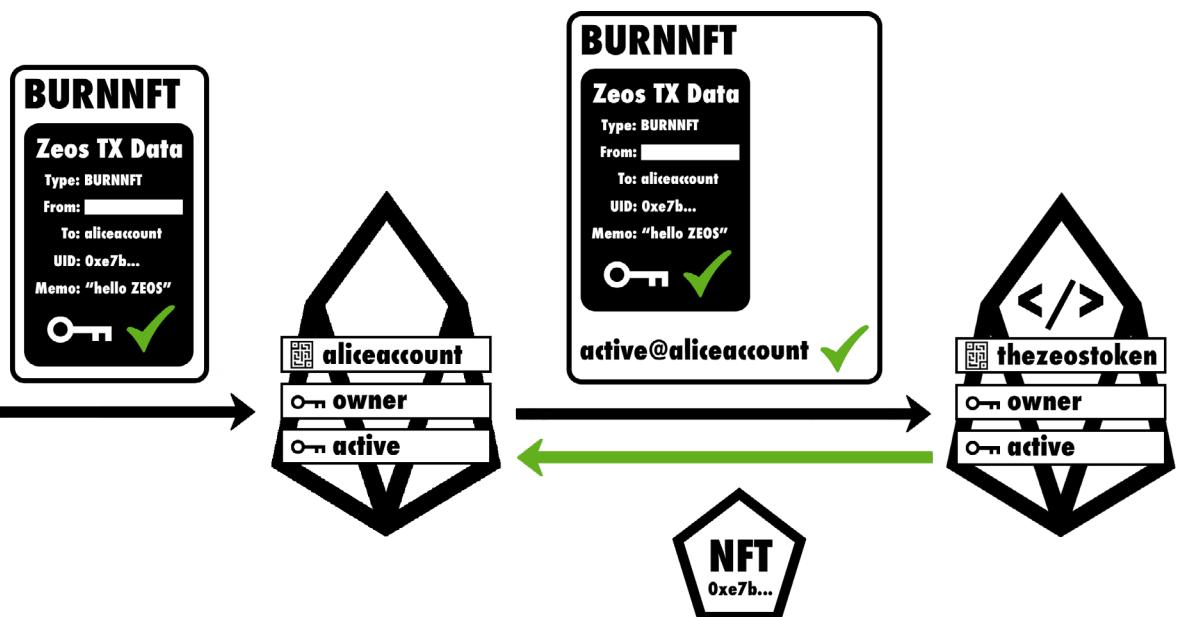
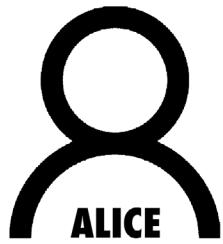
QUEMA DE UTXOS FUNGIBLES Y NO FUNGIBLES

Las UTXOs son canjeables en cualquier momento por el (mismo importe del) activo subyacente que representan. Para ello, la UTXO se quema utilizando la acción de "burn"(quema) correspondiente. Cuando esto ocurre, el activo subyacente se libera de la custodia del contrato del token ZEOS y se transfiere a la cuenta de EOS indicada por el usuario. La única limitación en cuanto a la quema podría ser un potencial valor de

enfriamiento(cool down) que pueda tener cualquier UTXO privada. En este caso, la quema fallará hasta que se alcance la altura del bloque EOS en particular. La única información que se revela públicamente al quemar UTXOs es de nuevo sólo la cantidad y el símbolo del activo (o el identificador único en caso de NFTs). La dirección de la billetera desde la que se quema el activo sigue siendo privada.



BURNFT: ALICE MUEVE 3 TOKENS EOS DE UNA CARTERA PRIVADA DE ZEOS A SU CUENTA PÚBLICA DE EOS

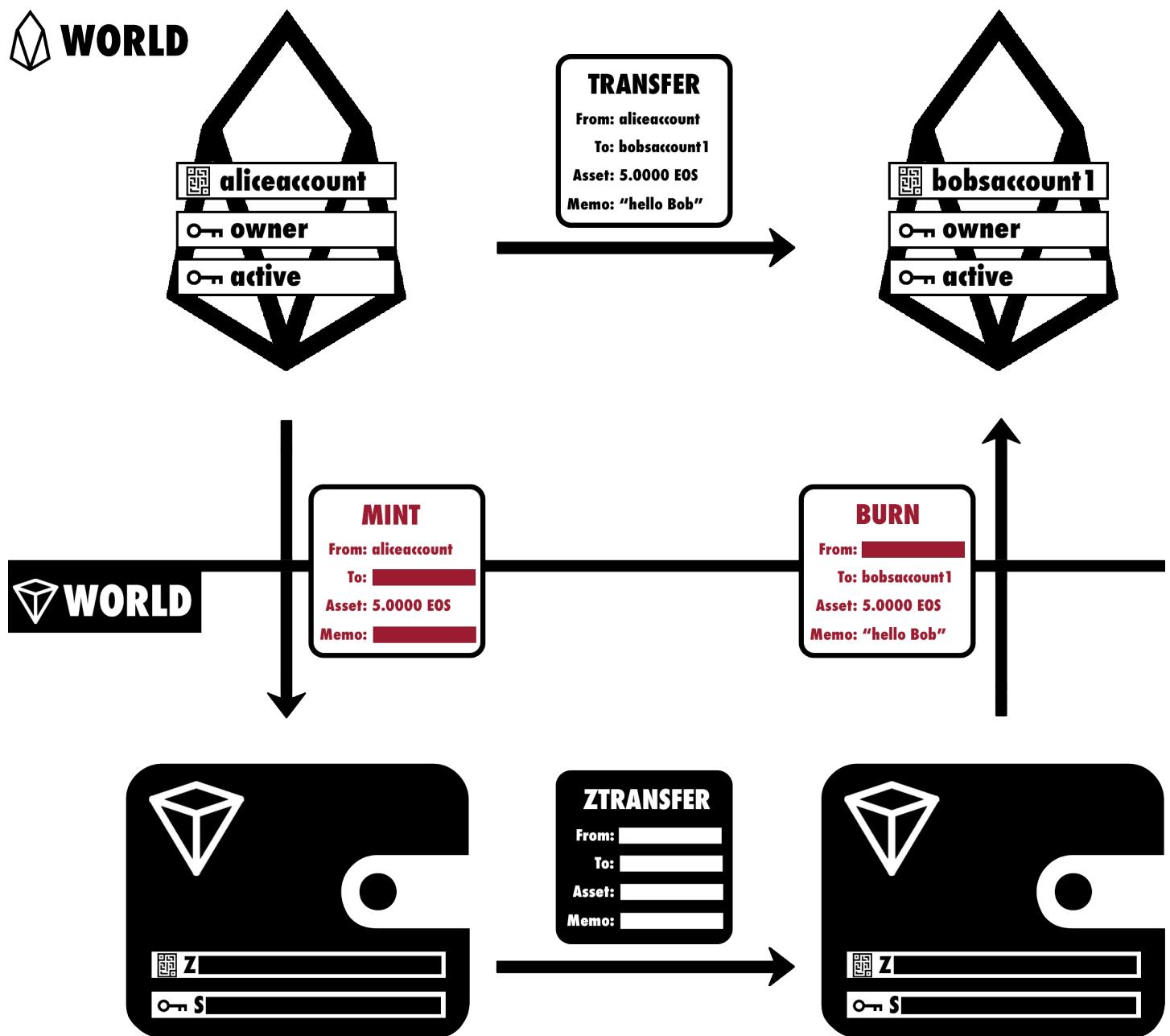


BURNNFT: ALICE MUEVE UN NFT DE UNA BILLETERA ZEOS PRIVADA A SU CUENTA EOS PÚBLICA

CONECTANDO LOS DOS MUNDOS

Las acciones de "crear"(mint) y "quemar"- (burn) para los activos fungibles y no fungibles pueden considerarse como la puerta de entrada que conecta los dos mundos: la única forma en la que los activos pueden pasar del mundo EOS transparente al mundo ZEOS privado es llamando a la(s)acción(es) "crear". Y la única forma en que los activos pueden pasar del mundo privado

de ZEOS al mundo transparente de EOS es llamando a la(s)acción(es) de "quemar"- (burn). Para las transferencias privadas p2p se utiliza la acción 'ztransfer' para tokens fungibles o no fungibles en analogía con la acción 'transfer' de tokens estándar de EOS.

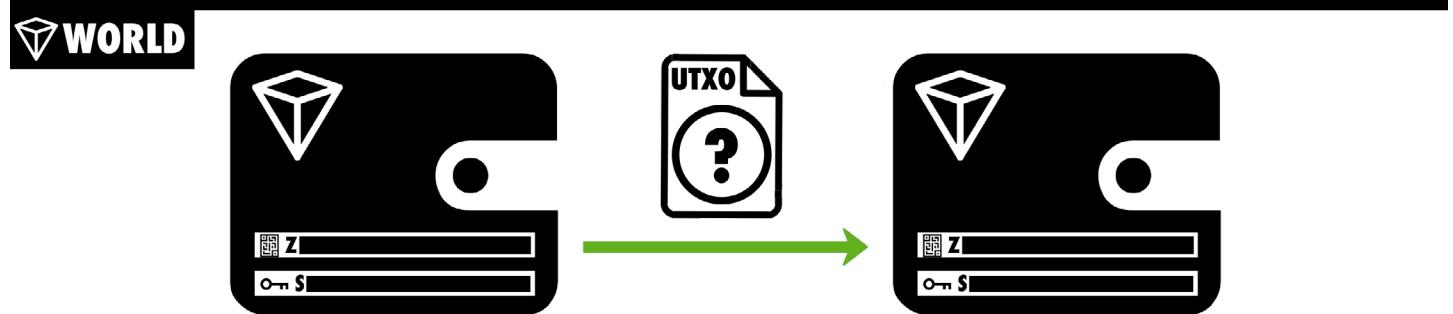
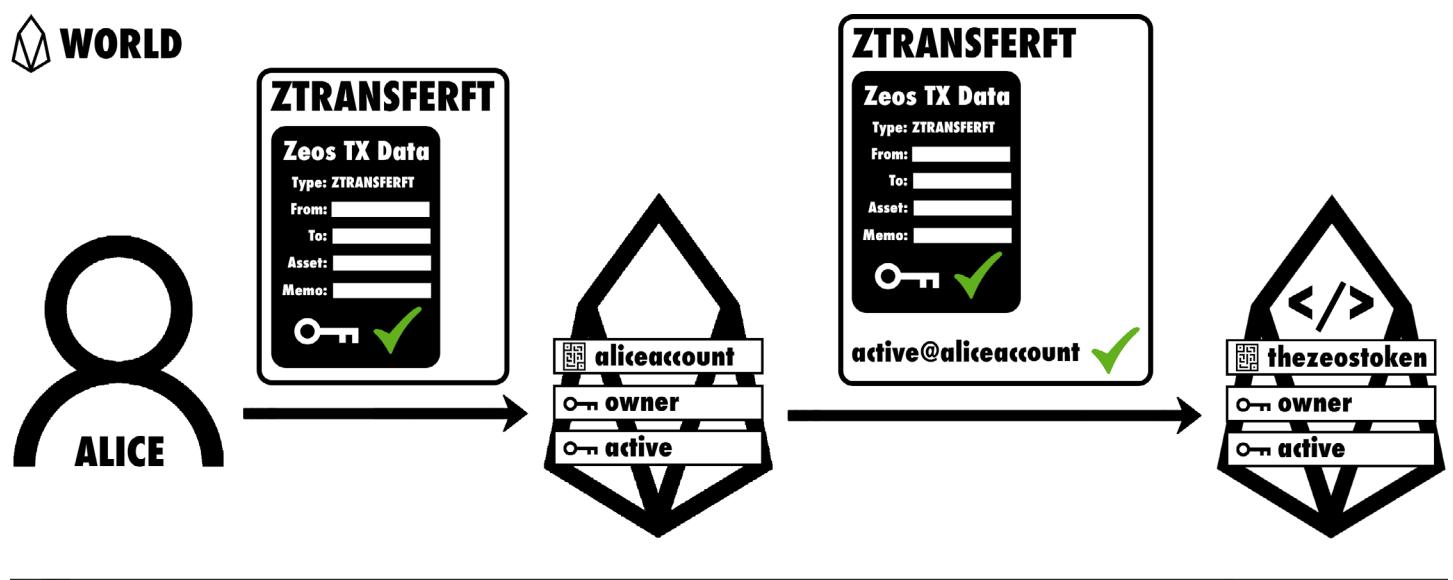


MOVIENDO ACTIVOS ENTRE LOS "DOS MUNDOS" LLAMANDO A "CREAR"(MINT) Y "QUEMAR"(BURN) EN TOKENS FUNGIBLES O NO FUNGIBLES Y SUS IMPLICACIONES PARA LA PRIVACIDAD

TRANSFERENCIA DE UTXOS FUNGIBLES Y NO-FUNGIBLES

Las UTXOs que representan tokens fungibles o no fungibles son totalmente transferibles de forma privada entre las billeteras de ZEOS. Si una UTXO representa un token fungible, es divisible hasta exactamente el

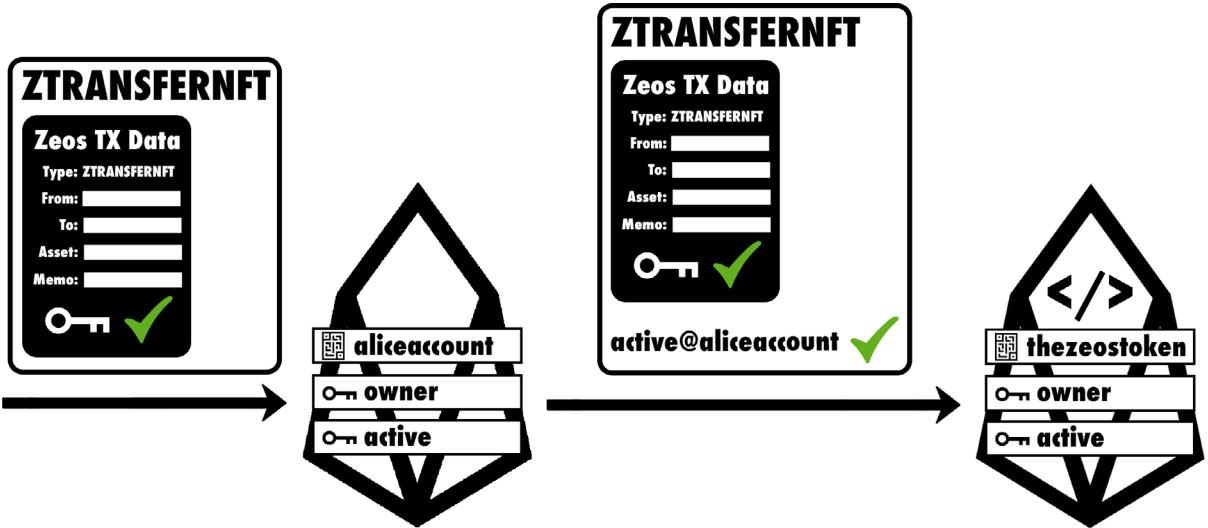
mismo grado que el propio activo subyacente. Las UTXOs de tokens no fungibles no son divisibles.



ZTRANSFERFT: ALICE REALIZA UNA TRANSFERENCIA PRIVADA DE TOKENS FUNGIBLES



ALICE



ZTRANSFERNFT: ALICE REALIZA UNA TRANSFERENCIA PRIVADA DE UN TOKEN NO FUNGIBLE

CREACIÓN DE UTXOS DE AUTENTICACIÓN

La creación y quema de tokens de autenticación no mueve ningún activo directamente. En su lugar, un token de autenticación representa simplemente un permiso

que permite al titular acceder a un contrato inteligente concreto. Para crear un token de autenticación el usuario crea un compromiso hash h tal que:

$$h \stackrel{\text{def}}{=} \text{hash}(\rho | \text{addr})$$

donde:

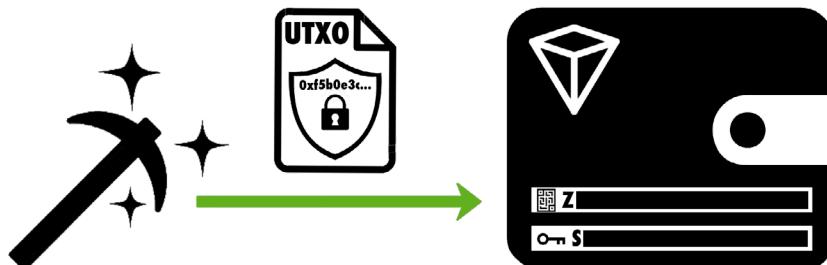
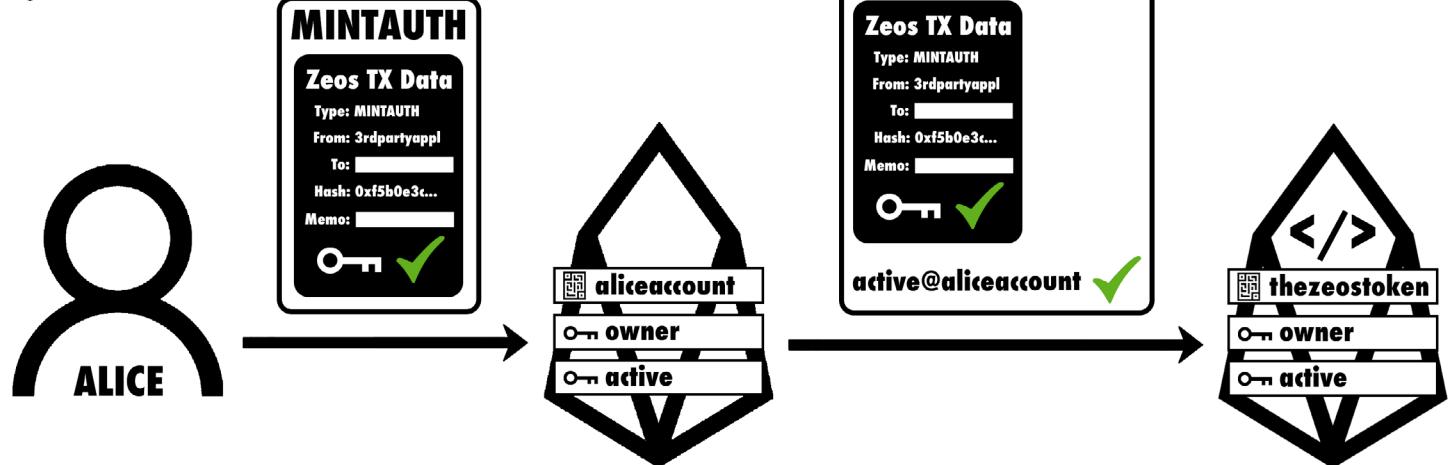
$\text{hash}()$ es una función hash resistente a colisiones

ρ es un número aleatorio de 32 bytes

addr es la dirección de la billetera donde se creó el token

La creación de tokens de autenticación mediante la llamada a "mintauth" se realiza normalmente como una acción en línea por un contrato inteligente de terceros que emite el permiso. El compromiso hash es almacenado por el contrato inteligente ha-

sta que el token de autenticación correspondiente es quemado por el usuario, es decir, es utilizado para la autenticación.

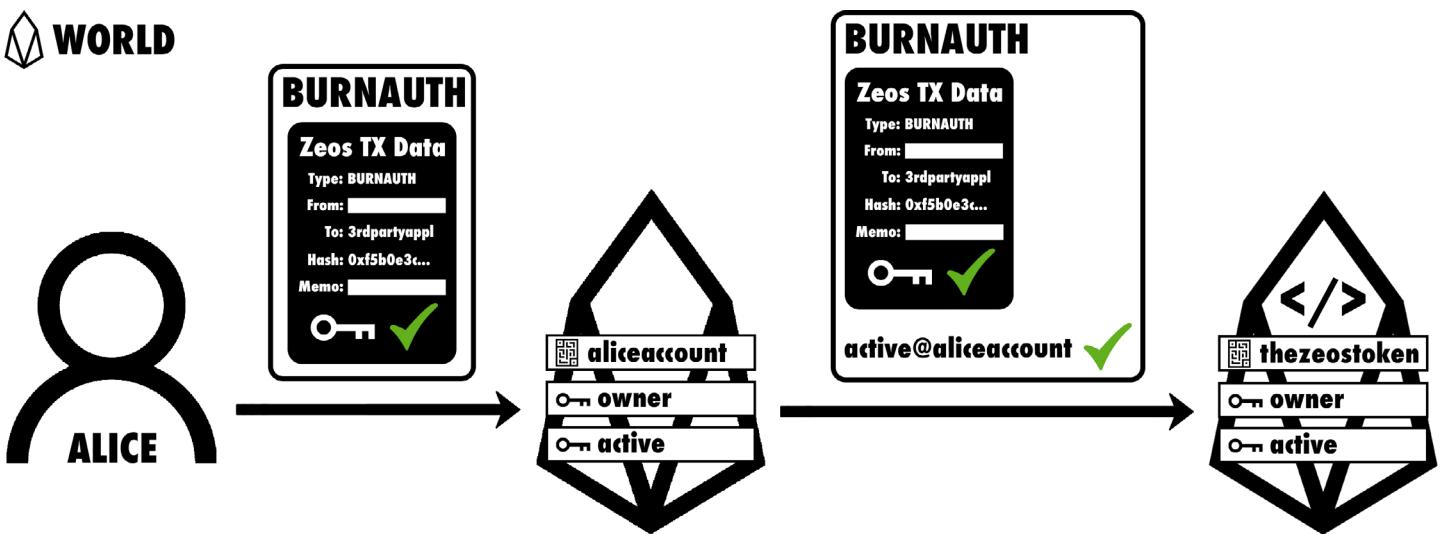


MINTAUTH: ALICE CREA UN TOKEN DE AUTENTICACIÓN EN UNA DIRECCIÓN PRIVADA DE ZEOS

QUEMA DE UTXOS DE AUTENTICACIÓN

Para obtener acceso al contrato inteligente de terceros, se quema el permiso representado públicamente por el compromiso h y representado privadamente por el token de autenticación dentro de la billetera de ZEOS.

Por lo que se genera una verificación de conocimiento cero que prueba el control público del valor secreto ρ y el conocimiento de la clave secreta a donde se encuentra el token de autenticación.



BURNAUTH: ALICE QUEMA UN TOKEN DE AUTENTICACIÓN EN UNA DIRECCIÓN PRIVADA DE ZEOS

DEPÓSITOS Y RETIROS PRIVADOS

Utilizando las acciones de "crear" y "quemar" de arriba, los depósitos y retiros privados de tokens pueden ser implementados por cualquier contrato inteligente de terceros en el ecosistema de EOS. Esto podría lograrse

utilizando las acciones en línea de EOSIO-Ahora Antelope). Por ejemplo, una acción de depósito privado a un contrato inteligente personalizado de terceros podría tener este aspecto:

3RDPARTYAPPL::PRIVDEPOSIT(DPST_PARAMS, AUTH_PARAMS)

```
// pseudo code for deposit of fungible tokens. It works analogously
// for non-fungible tokens: instead of 'quantity' save the NFT's UID.

// 'burn' the quantity from the user's private wallet into the 3rd party
// application's EOS account. NOTE: In your deposit notification handler
// make sure to ignore incoming transfers from 'thezeostoken'!
```

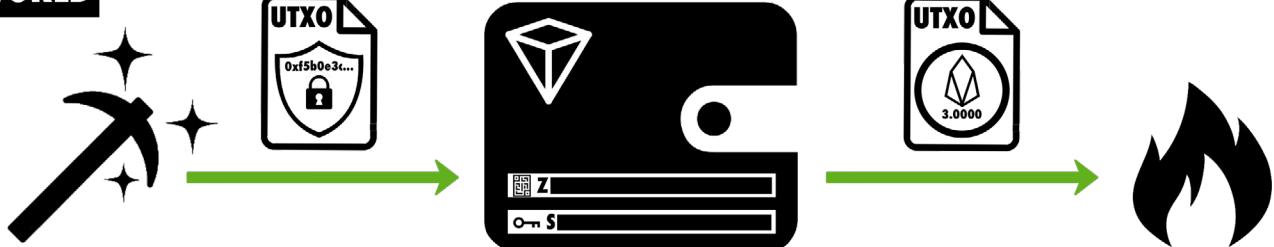
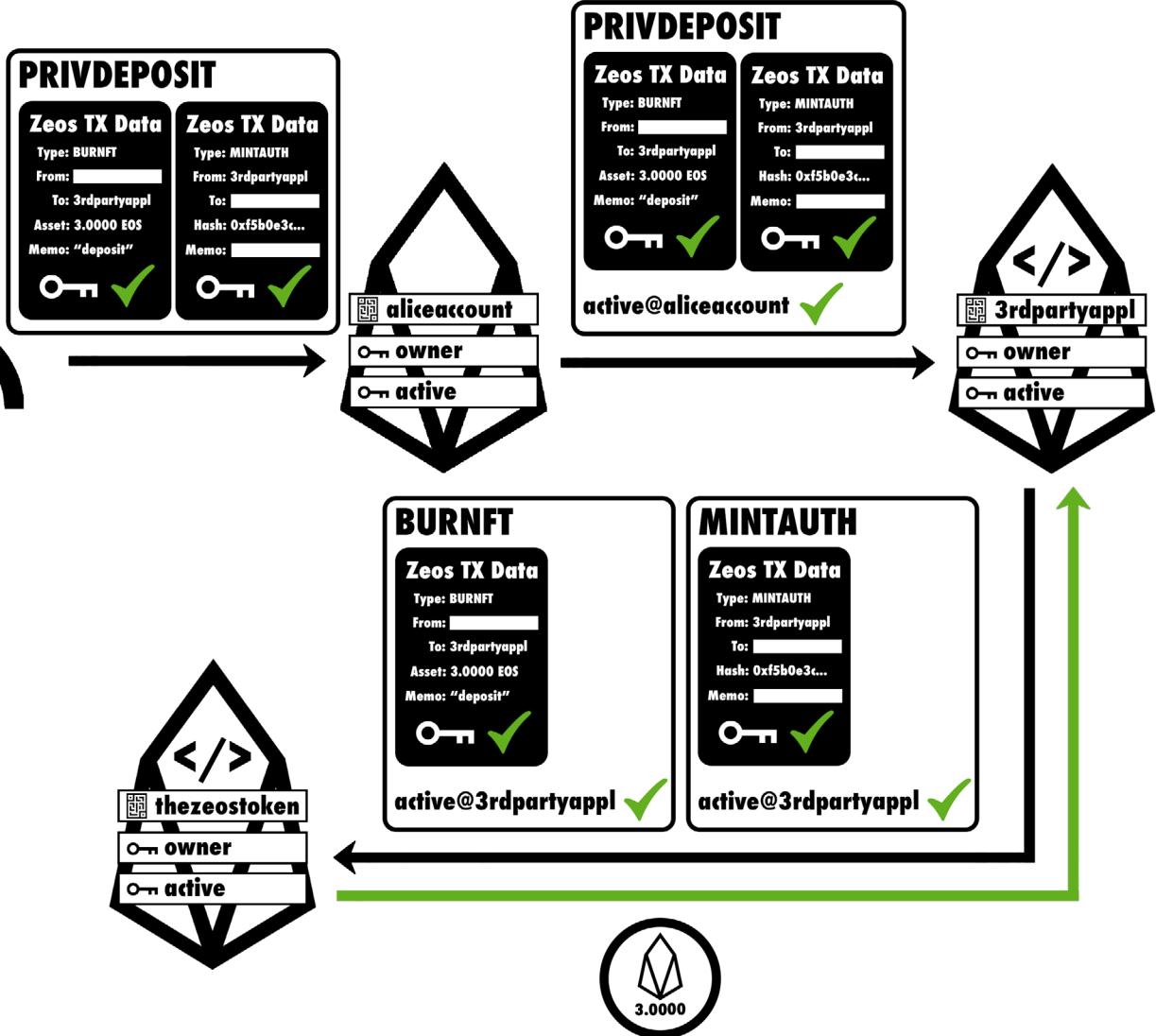
THEZEOSTOKEN::BURNFT(DPST_PARAMS, '3RDPARTYAPPL')

```
// 'mint' the user's authentication token
```

THEZEOSTOKEN::MINTAUTH(AUTH_PARAMS)

```
// save the deposited 'quantity' with the user's 'hash_commitment' as
// primary key in 'balances' table
balances[AUTH_PARAMS.hash_commitment] = DPST_PARAMS.quantity;
```

```
// more 3rd party deposit logic...
```



ALICE DEPOSITA DE FORMA PRIVADA 3 TOKENS EOS DESDE UNA BILLETERA ZEOS PRIVADA A UN CONTRATO INTELIGENTE DE TERCEROS

Incorpora dos acciones en línea de ZEOS: En primer lugar, el activo que se va a depositar (fungible o no fungible) se quema desde su dirección privada a la cuenta del contrato inteligente de EOS. En segundo lugar, en la misma acción, el usuario crea un token de autenticación y comparte el correspondiente hash públicamente con el contrato inteligente.

Para los depósitos y retiros tradicionales, los contratos inteligentes de EOS suelen mantener una tabla en la que los saldos de los usuarios se registran con sus correspondientes nombres de cuenta EOS. En el caso de los depósitos privados, los contratos inteligentes reservan el saldo bajo el hash que el usuario envía.

| USERN (key) | BAL... | HASH (key) | BAL... |
|--------------------|---------------|-------------------|---------------|
| geztomzxguge | 10... | 2d0e643f840... | 10... |
| gi2dmmzbzgene | 95... | 507f8dcda46... | 95... |
| gm3dmnzshege | 0.1... | 6cdb5439c8ce... | 0.1... |
| gq3dsnjthage | 73... | c384782a431... | 73... |
| guydkobvgige | 5.3... | b0d724bf725... | 5.3... |
| ... | | ... | |

VS

TABLA DE SALDOS EOS TRADICIONAL FRENTE A UNA TABLA DE SALDOS ZEOS

Un retiro privado de ese contrato inteligente podría entonces implementarse de esta forma:

3RDPARTYAPPL::PRIVWITHDRAW(WTHDRW_PARAMS, AUTH_PARAMS)

```
// pseudo code for withdrawal of fungible tokens. It works analogously
// for non-fungible tokens: instead of the user's balance the NFT with
// the given UID is minted back into the user's private wallet.
```

```
// 'mint' the quantity from the 3rd party application's EOS account into
// the user's private wallet.
```

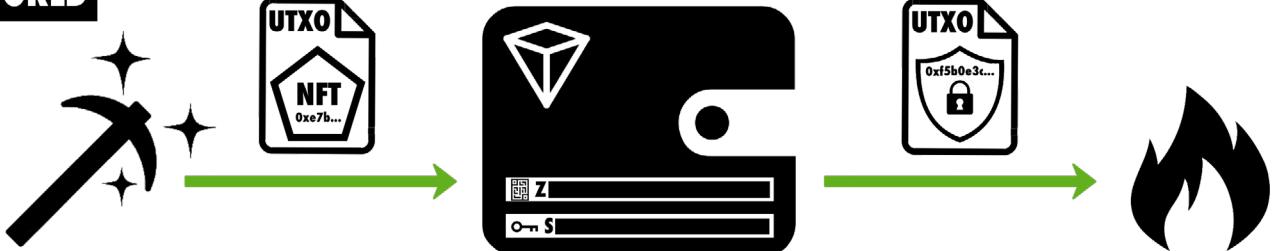
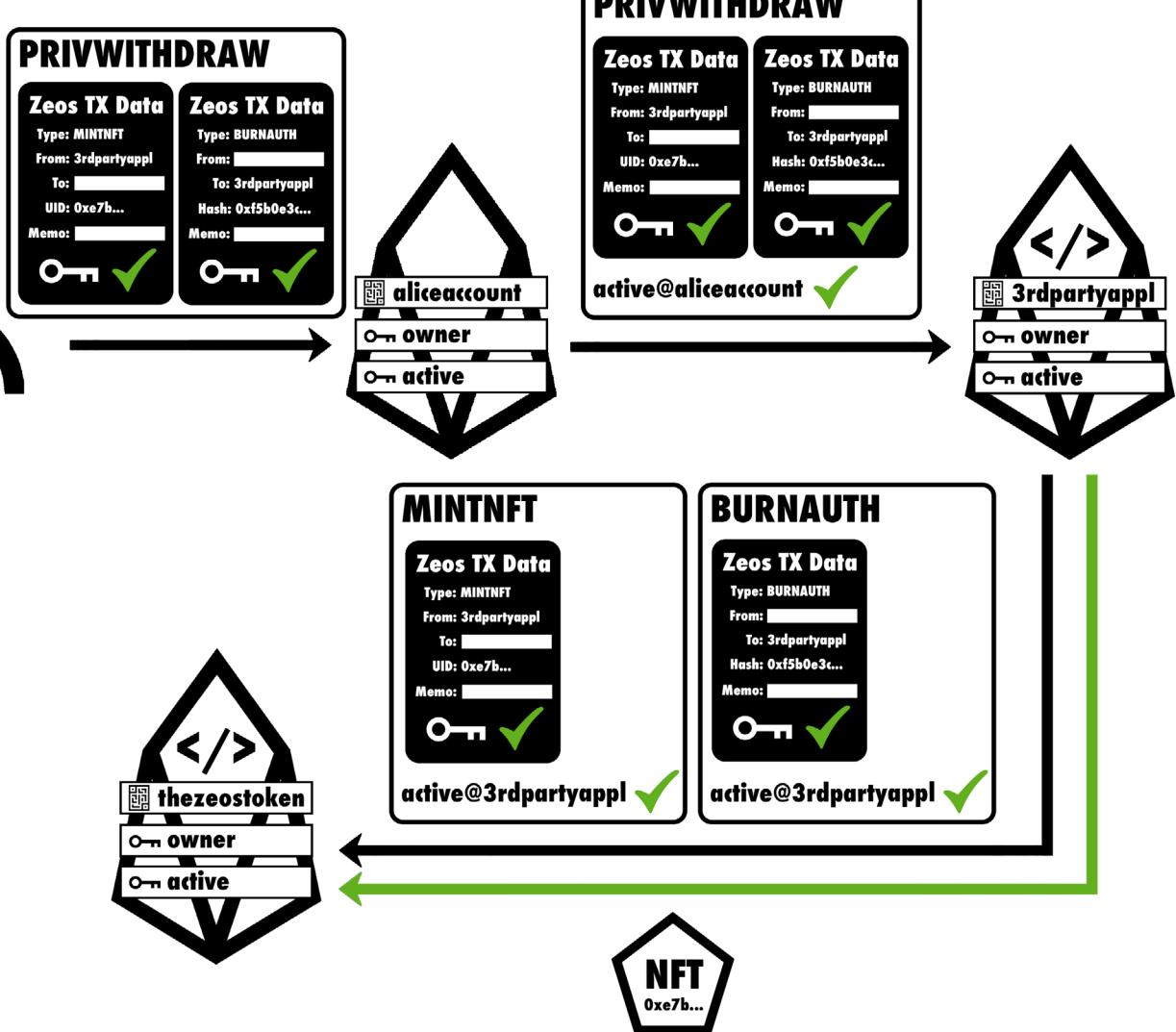
```
THEZEOSTOKEN::MINTFT(WTHDRW_PARAMS, '3RDPARTYAPPL')
```

```
// 'burn' the user's authentication token
```

```
THEZEOSTOKEN::BURNAUTH(AUTH_PARAMS)
```

```
// reset the balance of the user's 'hash_commitment' to zero. Since the
// authentication token is burned the balance must be withdrawn entirely
balances[AUTH_PARAMS.hash_commitment] = 0;
```

```
// more 3rd party withdraw logic...
```



ALICE RETIRA DE FORMA PRIVADA UN NFT DE UN CONTRATO INTELIGENTE DE TERCEROS A UN MONEDERO PRIVADO DE ZEOS

Al igual que los depósitos privados, los retiros privados incluyen dos acciones de ZEOS en línea: Primero se quema el token de autenticación. Al hacerlo, el usuario se autentifica demostrando que conozca (a) el valor secreto de rho y (b) la clave secreta de gasto de la dirección de la billetera en la que se encuentra el token de autenticación. Sólo el conocimiento de estas dos cosas

puede crear el compromiso hash público bajo el cual se almacena el saldo. En segundo lugar, el usuario crea una UTXO en una dirección privada de ZEOS para el activo a retirar, fungible o no fungible. Al hacerlo, el activo que se retira vuelve a estar bajo la custodia del contrato del token ZEOS mientras el usuario está en posesión de la correspondiente UTXO privada.

Sobre la base de este concepto para los depósitos y retiros privados, se podría implementar también una acción de "actualización" privada. En este caso, el usuario quema el permiso al mismo tiempo que crea uno nuevo para futuros accesos. Una acción de "actualización" podría por supuesto, contener varias acciones en línea de ZEOS, como crear y quemar activos (no) fungibles para, por ejemplo, retirar o depositar parcialmente activos en/desde un contrato inteligente de terceros. El saldo actualizado se almacenaría bajo el nuevo valor del hash.

Otra cosa importante a tener en cuenta aquí es que dicha acción de actualización privada también podría transferir el permiso a otro usuario. Esto podría hacerse eligiendo otra dirección de billetera al crear el nuevo token de autenticación. Así, mientras que los tokens de autenticación en sí mismos no son transferibles, los permisos sí son transferibles al actualizarlos con otra dirección de billetera. No hace falta decir que esto no podría ser detectado por un observador, ya que las direcciones de ZEOS nunca se exponen públicamente.

Utilizando las acciones en línea de ZEOS, todas las aplicaciones en EOS pueden integrar fácilmente los depósitos y retiros privados. Al permitir exclusivamente estos (es decir, no permitir depósitos y retiros de cuentas transparentes de EOS) la aplicación se convertiría en "privada por defecto".

Por supuesto, también es posible un modelo híbrido: En este caso, la tabla de contratos inteligentes contiene entradas mixtas de nombres de cuentas EOS y compromisos hash como claves primarias. Los nombres de cuentas EOS de 64-bits podrían ampliarse a 32 bytes añadiendo 24 cero bytes. Una función de clave primaria de 64-bits para esos valores de 32 bytes podría implementarse simplemente cortando los 24 bytes superiores que devuelven los valores de los nombres de cuentas EOS o los 64-bits inferiores de los compromisos hash. Las colisiones entre las confirmaciones hash y los nombres de cuenta son muy poco probables, pero necesitan ser manejadas por el contrato.

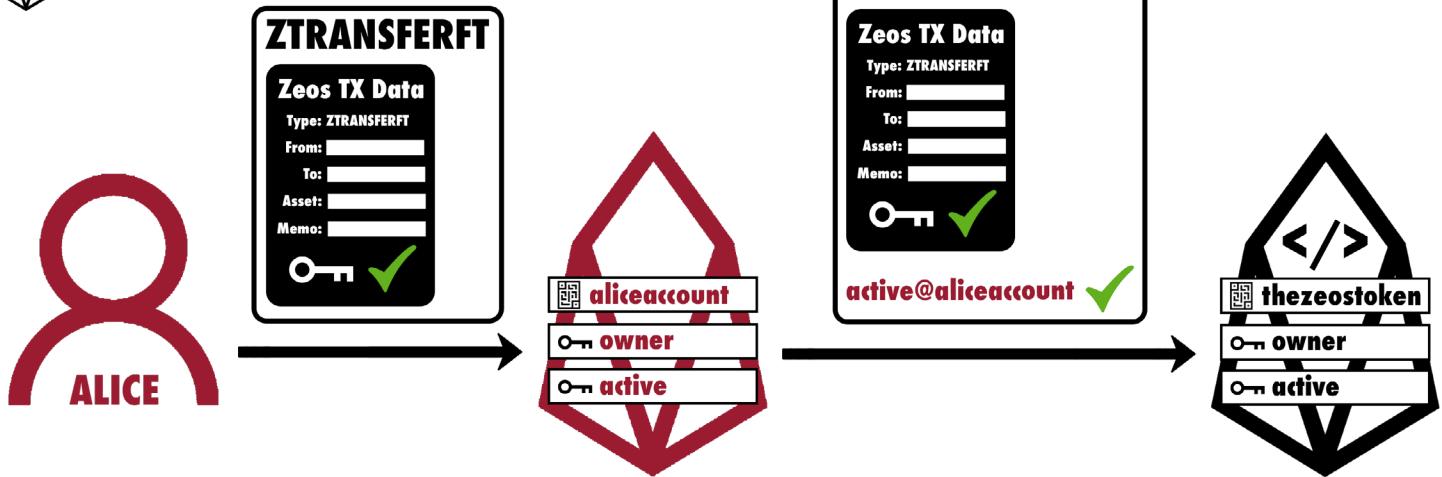
| MIXED (key) | BALANCE |
|------------------------------------|---------------------|
| 000000...000 geztomzxguge | 12.5000 EOS |
| 507f8dcda46...feb953f1c26ed | 255.1822 EOS |
| c384782a431...87202b8454cc9 | 10.7500 EOS |
| 000000...000 gq3dsnjthage | 995.1000 EOS |
| b0d724bf725...02f6d1af9a42c | 0.8912 EOS |
| ... | ... |

Nota: En caso de que una aplicación adapte los depósitos/retiros privados, el gestor de notificaciones utilizado para los depósitos de tokens de ese contrato tiene que gestionar las transferencias entrantes del contrato 'thezeostoken' por separado, es decir, ignorarlas, ya que esos saldos ya son gestionados por las acciones de depósito y retiro privados.

TRANSACCIONES PRIVADAS SIN CUENTA DE EOS

El mayor problema con respecto a las transacciones privadas en EOS es que todas las transacciones deben estar firmadas por un permiso de cuenta EOS para poder ejecutarse en la blockchain de EOS. Pero el hecho de que los usuarios firmen sus transacciones privadas con sus cuentas EOS personales les pone en riesgo, comprometiendo su privacidad: Una vez que una

cuenta EOS está vinculada a la identidad de un usuario, un observador público podría, como mínimo, detectar que ese usuario en concreto está "haciendo algo en privado". Este es el caso incluso de las transferencias privadas entre pares(p2p) en la que los datos sensibles están totalmente ocultos.



SI LA CUENTA EOS DE ALICE ESTÁ VINCULADA A SU IDENTIDAD LAS TRANSACCIONES PRIVADAS FIRMADAS CON SU CUENTA PUEDEN ESTAR VINCULADAS A ALICE COMPROMETIENDO (AL MENOS EN PARTE) SU PRIVACIDAD

En el caso de las transacciones privadas entre pares(p2p), la situación es aún peor: al depositar o retirar dinero de forma privada hacia o desde un contrato inteligente de un tercero, toda la información sobre los activos que se transfieren ya está expuesta públicamente. Lo único que queda en privado en una transacción de este tipo es la dirección de la billetera del usuario. Esto suele ser suficiente para mantener la identidad del usuario en el anonimato. Sin embargo, dado que la cuenta de EOS del usuario que se utiliza para firmar la transacción privada ya está vinculada a su identidad, un observador podría detectar que "el usuario X depositó el activo Y en el contrato inteligente Z", lo que básicamente significa que no hay privacidad en absoluto.

Esto, por supuesto, es un gran problema que podría resolverse exigiendo a los usuarios que creen de forma anónima una segunda cuenta EOS que se utilizaría exclusivamente para firmar transacciones privadas. De este modo, los usuarios utilizarían

su cuenta EOS conocida públicamente sólo para mover activos dentro y fuera de sus billeteras privadas ZEOS. Todas las demás transacciones privadas se firmarían sólo con su cuenta secundaria EOS anónima.

De este modo los usuarios consiguen una privacidad total siempre que su segunda cuenta EOS no se vincule nunca a su identidad. Aunque este enfoque proporciona una solución para una privacidad fuerte en EOS, requiere que todo el mundo gestione dos cuentas, incluido los recursos, lo que no es una solución muy fácil.

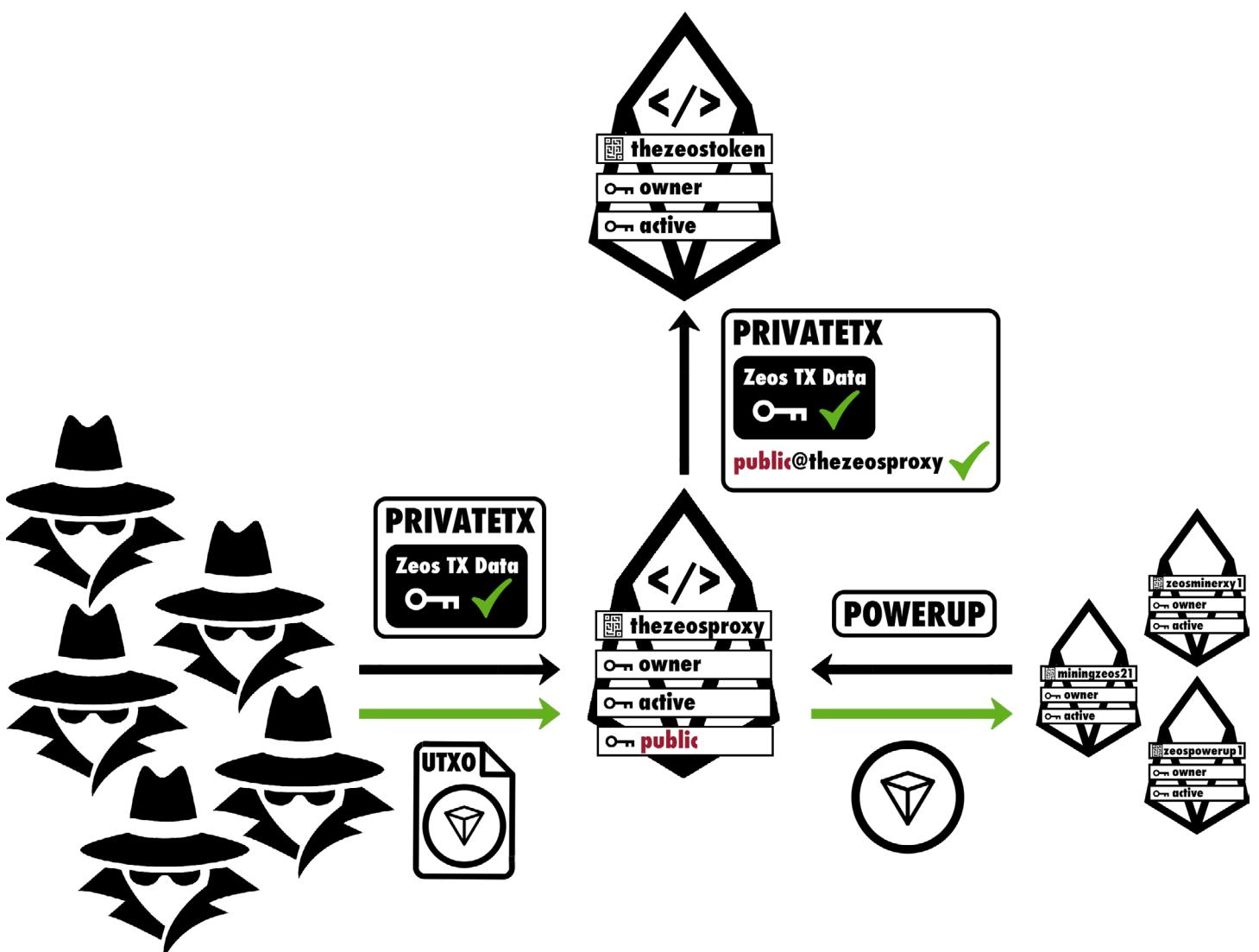
Además la ejecución de una sola transacción con la cuenta equivocada podría comprometer por completo la privacidad del usuario, incluidas todas las transacciones anteriores. También, los usuarios podrían crear, sin saberlo, vínculos entre sus cuentas por accidente a través de la gestión de recursos, como por ejemplo, que una cuenta alimente a la otra o viceversa. Hay demasiadas maneras en la que los usuarios

podrían arruinar su privacidad sin siquiera darse cuenta.

Para conseguir una privacidad real y a prueba de errores en EOS, es crucial desvincular las transacciones privadas de las cuentas EOS de los usuarios.

Aunque el mantenimiento de dos cuentas EOS podría gestionarse teóricamente de forma inteligente mediante un software de billetera, sigue existiendo el riesgo de que se produzcan errores humanos. Lo ideal sería tener dos cuentas EOS ni preocuparse de cuál utilizar para cada tipo de transacción. Por esto, ZEOS introduce un modelo de UTXO de la billetera privada ZEOS del usuario que los usuarios ni siquiera tuvieran que tener dos cuentas EOS ni preocuparse de cuál utilizar para cada tipo de transacción. Esto es posible gracias al sistema de permisos de EOSIO(Antelope) y a la función "Contract Pa-

blico y cualquiera puede utilizarla para firmar transacciones privadas de ZEOS. En lugar de pagar por los recursos necesarios de EOS de pagar por los recursos necesarios de EOS directamente cuando se ejecuta una transacción privada, el usuario que la realiza paga una tasa designada en ZEOS de forma directamente en la cuenta proxy de EOS. Esto se hace de una forma efectiva "quemando" una parte de una forma efectiva "quemando" una parte de las tasas por transacción alimentando los mineros calificados ganan sidad de tener una cuenta EOS. Esto es pos- sible gracias al sistema de permisos de recursos de CPU, NET y RAM de la cuenta pro-



TODO EL MUNDO UTILIZA EL PERMISO "PÚBLICO" PARA FIRMAR ANÓNIMAMENTE TRANSACCIONES PRIVADAS PAGANDO TASAS DENOMINADAS EN ZEOS POR LAS QUE LOS "MINEROS" COMPITEN"

EL PROXY

La cuenta proxy tiene un permiso especial "público" cuya clave privada se comparte públicamente. Esto significa que cualquiera puede utilizar este permiso para firmar transacciones de EOS. Sin embargo, este permiso sólo permite la ejecución de acciones específicas del contrato inteligente de la cuenta proxy. Estas acciones no son más que acciones "envueltas" para las tra-

nsacciones privadas de ZEOS. Esta acción envuelta incluye la transacción privada que se va a ejecutar más una llamada a "quemar" una UTXO de la billetera privada del usuario en la cuenta proxy, pagando así la tasa de la transacción. El ejemplo de una transacción privada entre pares "ztransfer" en el modelo de tasas sería el siguiente:

THEZEOSPROXY::ZTRNSFTPRXY(ZTRANSFER_PARAMS, FEE_PARAMS)

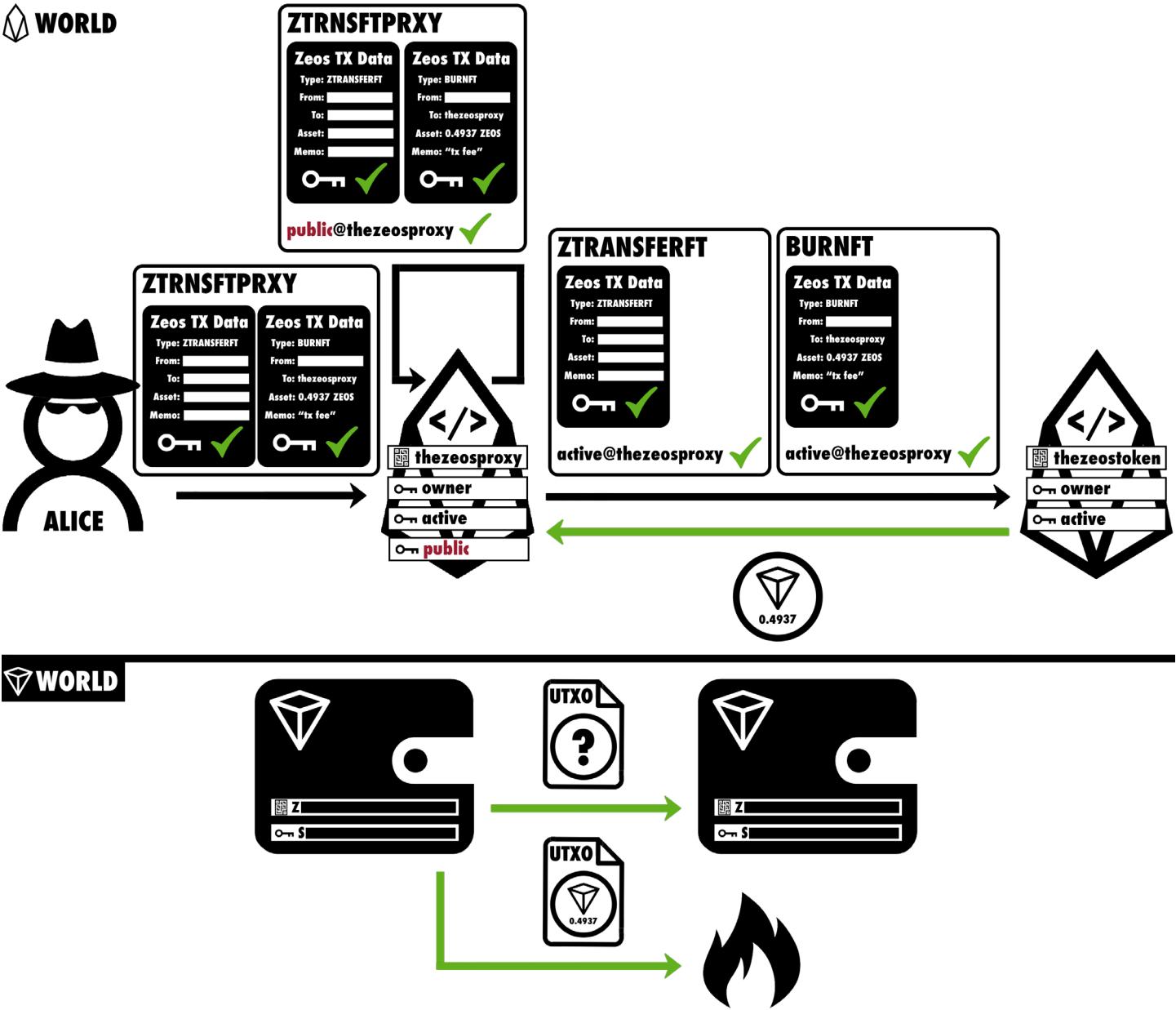
```
// pseudo code for private transfer of fungible tokens. It works analogously  
// for non-fungible tokens. The transaction fee is burned from the users  
// wallet into the proxy's EOS account. If the fee is not high enough to cover  
// all EOS resources this action fails.
```

```
// Execute the ztransfer
```

THEZEOSTOKEN::ZTRANSFERFT(ZTRANSFER_PARAMS)

```
// 'burn' the transaction fee into the proxy account
```

THEZEOSTOKEN::BURNFT(FEE_PARAMS, 'THEZEOSPROXY')



AL REALIZAR TRANSACCIONES PRIVADAS UTILIZANDO EL PROXY DE ZEOS, ALICE PERMANECE TOTALMENTE ANÓNIMA

Dado que la acción de quemar revela la cantidad y el tipo de activo, la tasa de transacción correcta puede ser fácilmente aplicada por la acción envuelta del proxy. Sólo si la tasa denominada en ZEOS es lo suficientemente alta como para cubrir los recursos de EOS de la transacción (más una pequeña cantidad como incentivo para los mineros de ZEOS) se ejecuta por la cuenta proxy. Los recursos que hay que pagar son la CPU, la NET y potencialmente la RAM (en el caso de un depósito privado a un contrato inteligente de terceros). Las tasas siempre tienen que pagarse en ZEOS.

La cuenta proxy no acepta ningún otro token para no revelar ninguna información sobre los activos que se transfieren. El hecho de que todos los recursos de EOS "CPU, NET y RAM" se paguen dinámicamente en ZEOS hace que ZEOS sea un "gas token" que potencia el acceso privado a la economía de EOS habilitando la cuenta proxy de ZEOS. El carácter de "gas token" hace que ZEOS sea comparable a ETH, que es el token de gas de Ethereum.

EL MINERO

Cualquiera que quiera ganar tokens de ZEOS proporcionando recursos de EOS a la cuenta proxy puede convertirse en "minero". Un minero supervisa constantemente los recursos disponibles de la cuenta proxy. Si los recursos restantes de la CPU/NET caen por debajo de un determinado umbral, los mineros son capaces de ejecutar una acción envuelta de la cuenta proxy para "alimentarla". La acción envuelta incluye dos acciones en línea: Una es la llamada para alimentar los recursos. La otra es una transferencia para pagar las tasas de transacción acumuladas en forma de tokens ZEOS a la cuenta EOS del minero. Por lo tanto, el minero sólo recibe la recompensa si la activación se lleva a cabo.

La cuenta proxy dispone de una segunda acción de "alimentar" para la RAM. Si la RAM es consumida por depósitos privados, el usuario pagará una cuota adicional para cubrir el coste de la RAM. Por lo tanto, el proxy de ZEOS mantiene un segundo balance sólo para la acumulación de tasas de RAM. Para la gestión de la RAM se produce exactamente la misma competencia entre mineros: Si la RAM restante de la cuenta del proxy cae por debajo de un determinado umbral, los mineros pueden ejecutar la segunda acción de alimentar que compra RAM³ para la cuenta del proxy y paga las tasas de RAM acumuladas, denominadas en ZEOS, a la cuenta EOS del minero.

Dado que las acciones de alimentar del proxy sólo son ejecutables bajo ciertas condiciones, se añade un poco de "teoría del juego" al proceso de minería: Los mineros compiten por el bote de las tasas y sólo pueden reclamarlas si son los que ejecutan el primer "alimentar" por debajo del umbral. Cualquiera puede ejecutar la acción de "alimentar" para participar en la

competición y aumentar el bote de dinero sin coste alguno para los mineros. Los mineros también podrían ejecutar transacciones privadas por sí mismos para hacer que el bote(s) se pueda reclamar antes de que lo haga el siguiente usuario. En cualquier caso: los mineros sólo pagan por los recursos si también son los que reciben el bote. Por otro lado, la recompensa del bote debería cubrir siempre los costes de "alimentar" o RAM, más un pequeño extra para el minero. Además de la prima para los mineros (que se establece a través de la gobernanza de ZEOS) se podría añadir una segunda prima a las tasas de transacción. Esto abriría una vía para que el protocolo genere ingresos y financie el trabajo de la organización inteligente de ZEOS que gobierna el ecosistema ZEOS.

³ Los mineros tendrían que mantener un pequeño saldo de EOS en la cuenta proxy para poder ejecutar "alimentar" o compras de RAM a través de llamadas de acción en línea. En aras de la simplicidad, esto se ignora en este documento.

CONCLUSIÓN

El modelo de tasas de ZEOS, basado en el concepto de un único permiso de cuenta pública que firma todas las transacciones privadas de ZEOS, es lo que hace que el protocolo sea verdaderamente privado e infalible. Las únicas interacciones con el protocolo que utilizan una cuenta EOS personal y enlazable es cuando los activos se mueven dentro o fuera de los monederos privados. Todas las demás transacciones, como las transferencias de tokens privados o las interacciones privadas con contratos inteligentes de terceros, se pagan con tasas de transacción denominadas en ZEOS y están firmadas por el permiso público de la cuenta proxy.

Esto tiene implicaciones aún más amplias: Las billeteras ZEOS pueden ser utilizadas por cualquier persona ajena al ecosistema EOS, ya que no se requiere una cuenta EOS personal para realizar transacciones privadas. Esto permitiría a cualquier persona ajena a EOS poseer todo tipo de activos EOS con total privacidad utilizando únicamente las billeteras ZEOS.

Además, permite a esos usuarios interactuar de forma privada con todo tipo de aplicaciones EOS que decidan integrar los depósitos y retiros privados de ZEOS. Todo lo que se necesita es una billetera ZEOS, la cual puede crearse con un clic, y por su puesto, algunos tokens ZEOS para cubrir las tasas de transacción.

También las aplicaciones web tradicionales como los exchanges centralizados, podrían integrar fácilmente las billeteras ZEOS, ya que funcionan casi exactamente igual que las billeteras de Bitcoin. Por ejemplo: Los exchanges centralizados podrían listar cualquier activo de EOS y permitir depósitos y retiros en el exchange usando direcciones privadas de ZEOS en lugar de las tradicionales cuentas EOS. Esto haría que todos los depósitos y retiros de activos EOS de los

usuarios de -no importa qué- fueran "privados por defecto".

El uso de billeteras ZEOS en lugar de cuentas EOS transparentes hace que cualquier activo en EOS sea una moneda con mayor privacidad que Monero. ZEOS ofrece transferencias de tokens totalmente privadas, además de un acceso potencialmente privado a cualquier aplicación DeFi construida sobre la misma blockchain, algo que las monedas de privacidad nativas como Monero son tecnológicamente incapaces de hacer debido a su falta de funcionalidad de "contrato inteligente". Los depósitos y retiros de ZEOS sin cuenta EOS podrían cambiar las reglas del juego, no solo para EOS, sino para todo el sector DeFi.

DEPÓSITOS Y RETIROS PRIVADOS USANDO LA CUENTA PROXY

Para completar los usos más complejos de la cuenta proxy de ZEOS, también se muestran aquí: Depósitos y retiros privados a/de un contrato inteligente de terceros.

Un depósito privado de tokens fungibles podría implementarse así (análogo a NFT):

THEZEOSPROXY::PRVDPSTPXY(FEE_PARAMS, DPST_PARAMS, AUTH_PARAMS)

3RDPARTYAPPL::PRIVDEPOSIT(DPST_PARAMS, AUTH_PARAMS)

```
// 'burn' the quantity from the user's private wallet into the 3rd party
// application's EOS account. NOTE: In your deposit notification handler
// make sure to ignore incoming transfers from 'thezeostoken'!
```

THEZEOSTOKEN::BURNFT(DPST_PARAMS, '3RDPARTYAPPL')

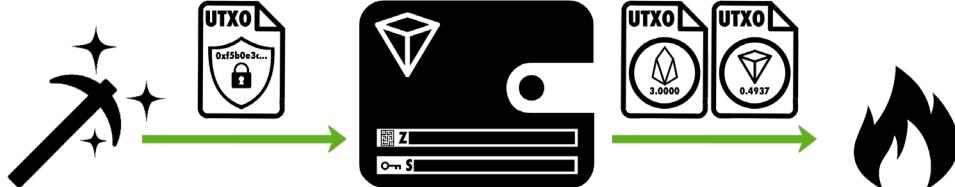
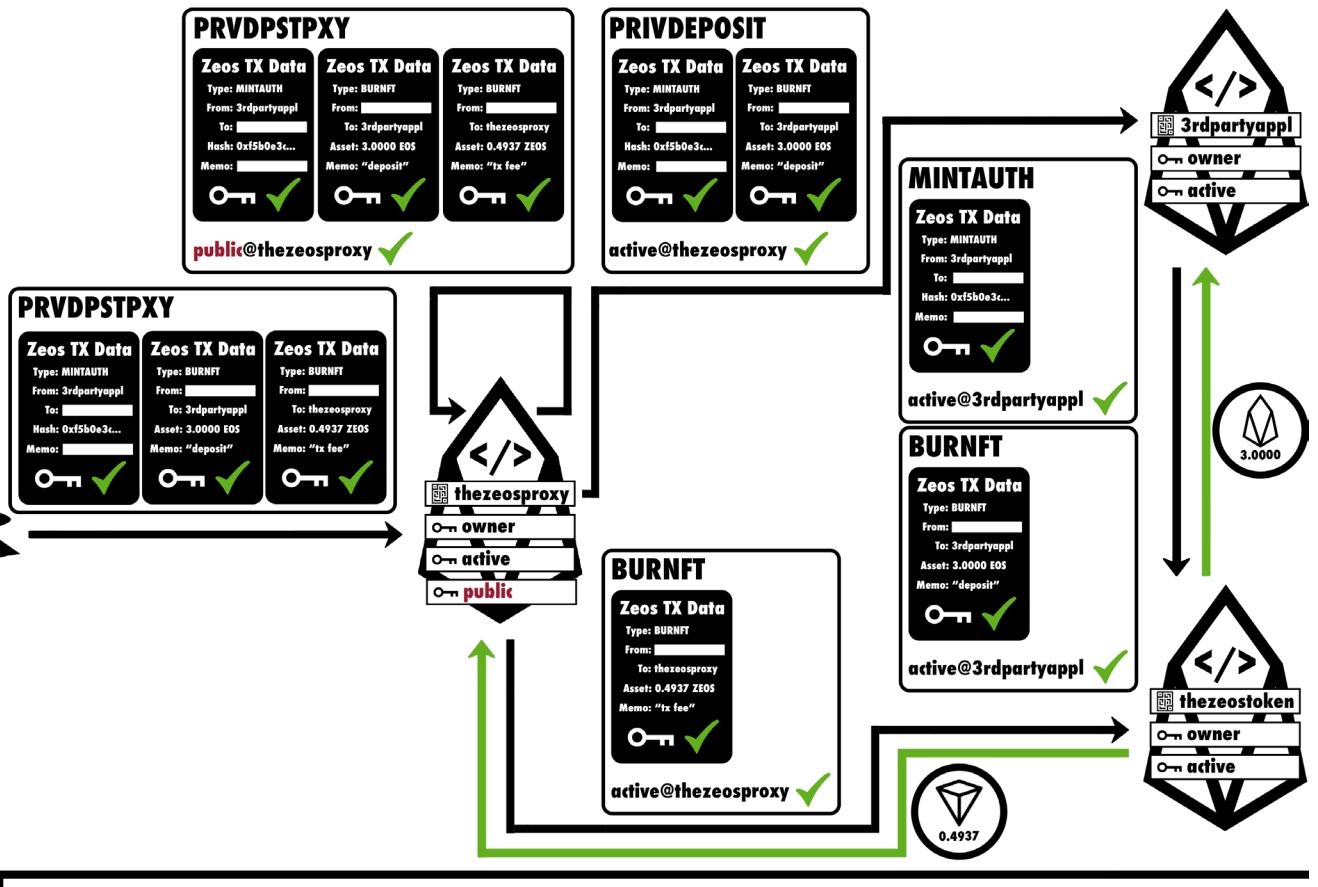
```
// 'mint' the user's authentication token
```

THEZEOSTOKEN::MINTAUTH(AUTH_PARAMS)

```
// 3rd party deposit logic...
```

```
// 'burn' the transaction fee into the proxy account
```

THEZEOSTOKEN::BURNFT(FEE_PARAMS, 'THEZEOSPROXY')



UN USUARIO DEPOSITA DE FORMA PRIVADA 3 TOKENS EOS DE SU BILLETERA ZEOS A UN CONTRATO INTELIGENTE DE TERCEROS UTILIZANDO LA CUENTA PROXY PAGANDO UNA TASA DE TRANSACCIÓN DENOMINADA EN ZEOS

Un retiro privado de tokens fungibles podría implementarse así (análogo a NFT):

THEZEOSPROXY::PRVWTHDRWPXY(FEE_PARAMS, WTHDRW_PARAMS, AUTH_PARAMS)

3RDPARTYAPPL::PRIVWITHDRAW(WTHDRW_PARAMS, AUTH_PARAMS)

// 'mint' the quantity from the 3rd party application's EOS account into
// the user's private wallet.

THEZEOSTOKEN::MINTFT(WTHDRW_PARAMS, '3RDPARTYAPPL')

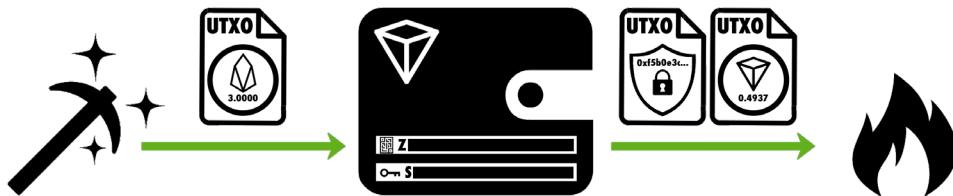
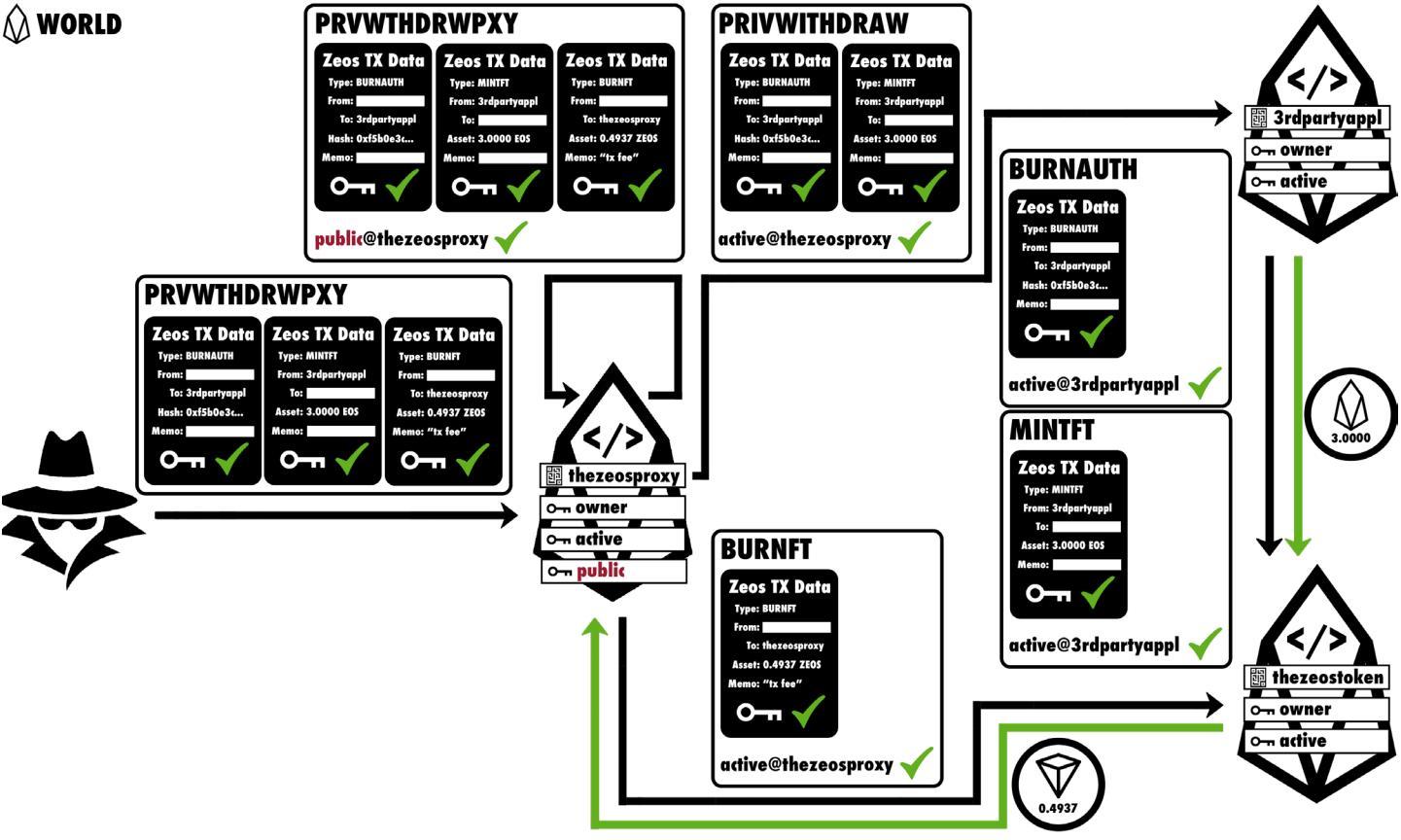
// 'burn' the user's authentication token

THEZEOSTOKEN::BURNAUTH(AUTH_PARAMS)

// 3rd party withdraw logic...

// 'burn' the transaction fee into the proxy account

THEZEOSTOKEN::BURNFT(FEE_PARAMS, 'THEZEOSPROXY')



UN USUARIO RETIRA EN PRIVADO 3 TOKENS EOS DE UN CONTRATO INTELIGENTE DE TERCEROS A SU CARTERA ZEOS UTILIZANDO LA CUENTA PROXY PAGANDO UNA TASA DE TRANSACCIÓN DENOMINADA EN ZEOS

ZASSETS

Inspirados por el [protocolo Haven](#), los zAssets son conjuntos sintéticos que representan cualquier tipo de activo del mundo real, como monedas fiduciarias, metales preciosos, materias primas o acciones. Todo lo que se necesita para crear un zAssets es un oráculo de precios para el correspondiente activo del mundo real. Los zAssets son creados por los usuarios a cambio de quemar una cantidad equivalente de tokens ZEOS y/o bloqueando un colateral externo en el protocolo. Sin embargo, a diferencia de Haven, la capitalización de mercado de todos los zAssets en circulación es auditável, ya que el proceso de creación y quema de zAssets es totalmente transparente.

La transparencia de los zAssets y, por tanto, el conocimiento del valor exacto de todos los activos sintéticos existentes, permite incorporar mecanismos de protección en el protocolo. Por ejemplo, el protocolo debería ralentizar y, eventualmente, detener por completo la quema de tokens ZEOS para crear activos sintéticos con el fin de protegerlo ante un apalancamiento excesivo. Algo de lo que carecen la mayoría de "monedas algorítmicas" en el mundo cripto. En lugar de quemar exclusivamente tokens ZEOS sin límite, el protocolo empieza a ralentizar automáticamente el ritmo de quema de ZEOS y, en su lugar, bloquea los colaterales externos si la demanda de activos sintéticos supera el crecimiento de la capitalización de mercado de ZEOS.

Todos los zAssets son tokens fungibles EOS estándar emitidos por el contrato del token ZEOS. Como todos los otros activos en EOS,

pueden moverse libremente entre billeteras privadas de ZEOS y cuentas EOS transparentes. Esto significa que pueden cotizar en todas las exchanges de EOS e integrarse en cualquier aplicación de EOS, como cualquier otro activo de EOS. Esto añade toda una nueva clase de activos al ecosistema DeFi en EOS.

A diferencia de muchos otros protocolos DeFi para activos estables sintéticos en el mundo cripto, los zAssets no son puramente algorítmicos. Este enfoque se inspira en el Protocolo Frax, pero con una diferencia: En lugar de requerir únicamente la quema de tokens ZEOS, el protocolo mantiene una reserva de colaterales externos que aumenta gradualmente en función del grado de *apalancamiento del protocolo*. El ratio de apalancamiento del protocolo L se define como:

$$L \stackrel{\text{def}}{=} \frac{M_{zeos}}{M_{syn}}$$

with:

$$M_{zeos} = s_{zeos} \cdot p_{zeos}$$

$$M_{syn} = \sum_{asset \in zAssets} s_{asset} \cdot p_{asset}$$

donde:

s_x es el suministro del activo x
 p_x es el precio del activo x

El apalancamiento del protocolo se define como la relación entre la capitalización de mercado del propio token ZEOS M_{zeos} y la capitalización de mercado de todos los activos sintéticos M_{syn} en circulación. El coeficiente de apalancamiento L determina cuántas veces todos los zAssets en circulación están respaldados por la valoración real del token ZEOS. Si este ratio de apalancamiento alcanza un determinado umbral el protocolo

lo comienza a acumular colaterales externos ya que la demanda de zAssets supera el crecimiento de la capitalización de mercado de ZEOS.

En función del coeficiente de apalancamiento L el colateral externo C_{ext} que requiere el protocolo se determina:

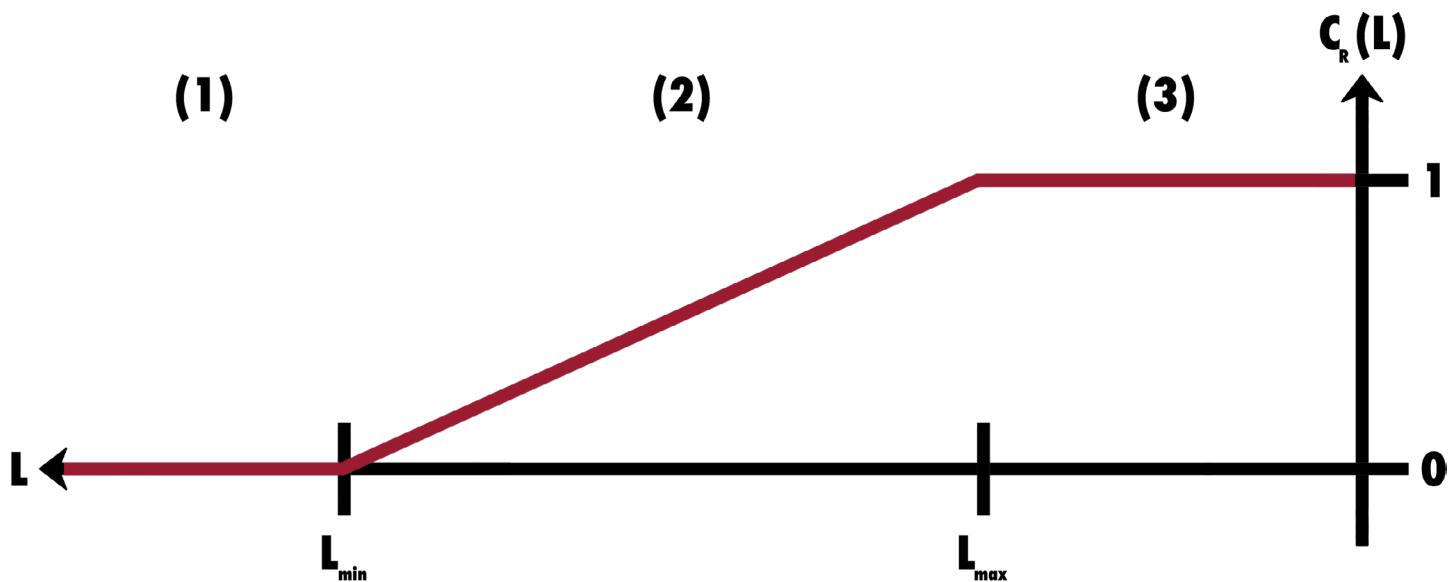
$$C_{ext} \stackrel{!}{=} C_R(L) \cdot M_{syn}$$

Donde $C_R(L)$ es el ratio de colaterales del protocolo definido en función de L :

$$C_R(L) = \begin{cases} 0 & \text{if } L_{min} < L \quad (1) \\ \frac{L - L_{min}}{L_{max} - L_{min}} & \text{if } L_{max} < L < L_{min} \quad (2) \\ 1 & \text{if } L < L_{max} \quad (3) \end{cases}$$

Donde L_{min} y L_{max} son valores del umbral establecidos por la gobernaza del protocolo. L_{min} determina cuándo el protocolo comienza a bloquear el capital externo y L_{max} determina cuándo el protocolo

requiere exclusivamente un colateral externo para crear activos sintéticos. Ambos valores determinan implícitamente la tasa de quema y creación de ZEOS como colateral para los zAssets.



UN MENOR COEFICIENTE DE APALANCAMIENTO L DA LUGAR A UN MAYOR COEFICIENTE DE COLATERALES AUMENTANDO LA EXIGENCIA DE BLOQUEO DEL COLATERAL EXTERNO

Por lo tanto, el protocolo tiene tres estados diferentes que transitan sin problemas entre sí: (1)puramente algorítmico/no colateralizado.

(1) Puramente algorítmico

Si el ratio de apalancamiento L es mayor que el ratio mínimo, L_{min} , el protocolo se encuentra en el estado puramente algorítmico en el que sólo se queman tokens ZEOS para crear zAssets y viceversa. En este estado, el protocolo no acumula ningún colateral.

(2) Algoritmo fraccionado/colateralizado

Si el ratio de apalancamiento L está entre L_{min} y un ratio de apalancamiento máximo L_{max} el protocolo se encuentra en un estado de algoritmo fraccionado. En este estado, el requisito de colateral externo C_{ext} es igual a un determinado porcentaje de la capitalización total del mercado de todos los zAssets M_{syn} en circulación. Para crear nuevos activos sintéticos, los usuarios tendrán que añadir en parte algún colateral externo y en parte quemar tokens Z-EOS. La quema de activos sintéticos, por otro lado, devolverá al usuario algunos colaterales externos y creará algunos tokens de ZEOS para que vuelvan a existir.

Sólo si el valor del colateral externo C_{ext} es demasiado pequeño, el protocolo creará (mayoritariamente o exclusivamente) tokens ZEOS en lugar de devolver el colateral externo al quemar zAssets. Por otro lado, si el valor del colateral externo es superior al nivel requerido, el protocolo devolverá (mayoritaria o exclusivamente) el colateral externo al usuario en lugar de crear tokens ZEOS.

(3) Completamente colateralizado

Si el ratio de apalancamiento cae por debajo de L_{max} el protocolo entra en el estado de colateralización total. En este estado ya no se queman tokens ZEOS. Todos los zAssets creados en este estado requieren un colateral externo. La quema de activos sintéticos devolverá el colateral externo a los usuarios en lugar de crear nuevos tokens ZEOS. Sólo si el valor global del colateral externo es demasiado bajo, el protocolo empezará a crear tokens ZEOS de nuevo para aumentar el ratio de colateralización externa y así poder pagar a los usuarios a cambio de quemar zAssets.

Con cada interacción del usuario, es decir, crear y quemar zAssets, el protocolo intenta mantener la cantidad correcta de colaterales externos C_{ext} , determinada por la ratio de apalancamiento L . Dado que los valores de M_{syn} , M_{zeos} y el propio C_{ext} cambian constantemente debido a las fluctuaciones de los precios en los mercados abiertos, la necesidad del colateral externo C_{ext} también cambia permanentemente. Aunque los usuarios siempre obtienen exactamente el mismo valor que añaden por parte del protocolo, la proporción de tokens ZEOS que se queman (o crean) frente al colateral externo que se añade (o elimina) puede cambiar en cada transacción.

Equilibrar la cantidad correcta de colaterales externos C_{ext} según lo determinado por el apalancamiento actual de protocolo L es la única prioridad del protocolo.

COLATERAL

El colateral externo C_{ext} bloqueado dentro del protocolo consta de dos activos:

- El token nativo de EOS
- La ficha estable Vigor, que está vinculada al dólar estadounidense

Al elegir un criptoactivo con una capitalización de mercado significativamente más alta que el token de ZEOS en sí mismo -EOS-

más un token estable que está vinculado al dólar -Vigor-, se espera que la valoración global del colateral externo C_{ext} sea mucho menos volátil que la propia capitalización de mercado de ZEOS M_{zeos} .

El valor total del colateral externo C_{ext} es la suma del valor total del colateral EOS más el valor total del colateral Vigor:

$$C_{ext} = C_{eos} + C_{vigor}$$

El protocolo siempre intenta equilibrar los fondos del colateral: El valor de todos los tokens EOS bloqueados dentro del protocolo deben ser siempre igual al valor de todos los tokens estables de Vigor bloqueados. Amb-

os fluctúan en valor y, por tanto, la proporción en la que los usuarios tienen que añadir o quitar tokens EOS y Vigor cuando crean o queman zAssets también fluctúa.

$$C_{eos} \stackrel{!}{=} C_{vigor}$$

La condición de mantener siempre los dos en equilibrio resulta en el siguiente comportamiento de acumulación: En un mercado cripto alcista, cuando se crean zAssets, el protocolo requiere que se añadan nominalmente más tokens de Vigor que de EOS, ya que EOS está aumentando su valor frente a Vigor. Al quemar zAssets, los usuarios recibirían más tokens de EOS de vuelta que de Vigor.

En un mercado bajista, en cambio, sería lo contrario: El protocolo acumula tokens EOS mientras paga predominantemente Vigor. Este comportamiento anticíclico de acumulación permite a los usuarios deshacerse de sus activos "débiles" al crear zAssets y recibir activos "fuertes" al quemar zAssets, dependiendo del estado del mercado.

CREACIÓN Y QUEMA DE ZASSETS

Dependiendo del estado actual del protocolo, así como de la valoración del colateral externo C_{ext} bloqueado en el protocolo, la creación y quema de zAssets requerirá una proporción diferente de tokens ZEOS, EOS y Vigor que se añadirán o eliminarán, respectivamente.

Por ejemplo, un usuario quiere crear (o quemar) una determinada cantidad a_x de un activo sintético X . Este activo tiene un precio p_x en el mercado abierto definido por el correspondiente oráculo de precios. Se aplica la siguiente ecuación:

$$a_x \cdot p_x = a_{zeos} \cdot p_{zeos} + a_{eos} \cdot p_{eos} + a_{vigor} \cdot p_{vigor}$$

Los importes a_{zeos} , a_{eos} y a_{vigor} deben determinarse ahora de forma que el valor real del colateral externo C_{ext} del protocolo se aproxime lo mejor posible al valor del colateral externo requerido, incluida la transacción actual.

No importa si se trata de crear o quemar zAssets, y no importa en qué momento: Los usuarios reciben por parte del protocolo exactamente el mismo valor que han introducido en él, calculado en el momento de la transacción. Sólo la proporción de ZEOS, EOS y Vigor cambia según sus valores actuales y el estado actual del protocolo.

REEQUILIBRIO DEL COLATERAL

En determinadas condiciones de mercado, la demanda de colaterales externos puede aumentar mucho más rápido de lo que el protocolo es capaz de "ponerse al día" sólo con la creación y quema continua de zAssets por parte de los usuarios. Por esto se introduce una acción de "reequilibrio" que pueden ejecutar los usuarios en caso de que el valor del colateral requerido se despegue demasiado del valor real del colateral bloqueado en el protocolo. El porcentaje en el que el valor del colateral puede despegarse antes de que la acción de reequilibrio sea ejecutable se establece a través de la gobernanza de ZEOS.

Si el protocolo está demasiado infracolateralizado, los usuarios pueden ejecutar la acción de reequilibrio para añadir más tokens EOS y Vigor al protocolo a cambio de tokens ZEOS recién creados. Esto resulta en un mayor colateral externo bloqueado dentro del protocolo, así como un aumento en la capitalización de ZEOS, ya que más tokens entran en circulación. Ambas acciones aumentan la solvencia del protocolo.

Por otro lado, si el protocolo se sobrecolateraliza, los usuarios pueden ejecutar la acción de reequilibrio para intercambiar tokens ZEOS por un colateral externo del protocolo. Los tokens ZEOS añadidos son quemados por el protocolo mientras que una cantidad equivalente de colateral externo es librada para pagar al usuario. Esto da lugar a una disminución de la capitalización de mercado de ZEOS, así como una disminución del colateral bloqueado. Ambas acciones reducen la solvencia del protocolo y aumentan la escasez de ZEOS.

RETRASOS Y MEDIAS MÓVILES

Para prevenir el abuso de creación y quema de la tokenomía del protocolo en momentos de tensión en los mercados o mediante manipulaciones del oráculo de precios a corto plazo, se introducen dos medidas para filtrar las fluctuaciones de precios de alta frecuencia:

- tiempos de espera para quemar zAssets
- medias móviles para los valores de precios del oráculo

Un tiempo de espera evita que los usuarios quemen zAssets justo después de haberlos creado. Esto hace que el protocolo sea más atractivo para los traders e inversores a largo plazo en lugar de traders a corto plazo que podrían aprovecharse de la tokenomía: Dado que las tasas para la creación y quema de zAssets son muy bajas, pero proporcionan una liquidez infinita, los traders podrían provocar pequeñas fluctuaciones en el precio del token ZEOS mientras utilizan las diferencias de precio para crear y quemar zAssets y, por lo tanto, acumular tokens ZEOS a costa de la inflación. Un simple tiempo de espera para quemar zAssets resuelve este problema.

Además, se usan medias móviles en todos los valores del oráculo para filtrar fluctuaciones de alta frecuencia. Esto suaviza todas las señales de precio que entran en el protocolo, lo que lleva a un estado del protocolo menos volátil y más estable. Esto es también una medida contra las manipulaciones de los oráculos de precio a corto plazo.

El valor del tiempo de espera y el tamaño de las medias móviles se fijan a través de la gobernanza del protocolo.

EFICIENCIA DEL CAPITAL

El suministro flexible de tokens ZEOS permite que los zAssets sean mucho más eficientes en términos de capital que las posiciones de deuda sobrecolateralizadas (CDPs) como Vigor y otros activos estables de esa categoría. Por ejemplo: Es necesario bloquear colaterales por valor de varios dólares para crear sólo un dólar de CDP. Además, hay que pagar intereses por cada dólar de CDP existente.

Esto hace que los CDPs sean muy ineficientes en términos de capital, pero también relativamente de bajo riesgo. Por el contrario, el protocolo de ZEOS para los zAssets es más eficiente en términos de capital, ya que sólo se necesita un dólar de ZEOS y/o un colateral externo para crear un dólar de zAssets. No hace falta decir que esto tiene un coste de riesgo para los holders de tokens ZEOS.

RIESGO

La capitalización de ZEOS, así como el capitalización de ZEOS, así como el colateral externo C_{ext} está bloqueado en el colateral externo C_{ext} bloqueado en el protocolo. Por lo tanto, el verdadero ratio de apalancamiento que determina la solvencia S del protocolo se define por: La solvencia global del protocolo se define por:

$$S = L_{true} \stackrel{\text{def}}{=} \frac{M_{zeos} + C_{ext}}{M_{syn}}$$

El valor de S nunca debe ser inferior a 1.0 para que el protocolo no se liquide

Establecer los límites L_{min} y L_{max} lo pagarse por los mercados y en reflejar un suficientemente altos para que el protocolo incremento en el precio del token ZEOS. sea capaz de sobrevivir a eventos como el Estas dinámicas son importantes consi- "cisne negro" depende de la gobernanza del derarlas cuando hablamos de la gober- protocolo. Es crucial que el protocolo pase nanza del protocolo. Los posibles valores con suficiente antelación de un estado pura- iniciales del umbral podrían ser: $L_{min}=100$ y mente algorítmico al estado de colatera- $L_{max}=10$.
lización fraccionada y, por tanto, comience a bloquear el colateral externo. Además, el protocolo también debería pasar al estado de colateralización total con la suficiente antelación para tener un "colchón" suficiente para el riesgo de caída.

L_{min} y L_{max} determinan implícitamente el ratio de quema de los tokens ZEOS: Cuanto mayor sea el valor de L_{min} antes empieza el protocolo a acumular colaterales externos, lo cual significa bajar el ratio de quema de los tokens ZEOS. Cuanto mayor sea el valor de L_{max} , antes dejará el protocolo de quemar tokens ZEOS por completo y, en su lugar, sólo requerirá colaterales externos para ser bloqueados. La reducción de ambos valores aumenta el ratio de quema de ZEOS, pero también supone un mayor riesgo para los holders de tokens ZEOS.

La solvencia del protocolo depende en gran medida de la capitalización de mercado del token ZEOS, que a su vez depende no sólo del precio de ZEOS sino también de la cantidad de tokens ZEOS en circulación. Quemar tokens ZEOS significa ante todo quemar valor de ZEOS. La escasez tarda un tiempo en pro-



EL TOKEN

Todo el ecosistema de ZEOS está impulsado por el token ZEOS, que cumple múltiples funciones. Por un lado, sirve como token de pago para las tasas de las transacciones privadas sin cuenta EOS. Dado que todas las aplicaciones en EOS podrían integrar los depósitos y retiros privados de ZEOS, el proxy de ZEOS podría proporcionar acceso anónimo a todas las aplicaciones en EOS. ZEOS se convierte en un "token de gas" con el que hay que pagar por las transacciones privadas para cubrir los costes de los recursos de la blockchain de EOS, como el CPU, NET y RAM. Esto incluye las transferencias privadas entre usuarios, así como depósitos y retiros privados hacia desde aplicaciones de terceros en EOS. Esto da a ZEOS un carácter similar al de "Ether", que es el token de gas de la blockchain de Ethereum.

Además, ZEOS es el llamado "share token" necesario para crear (fraccional-)algorítmicamente los zAssets. El suministro flexible de tokens permite crear activos sintéticos eficientes en términos de capital, añadiendo una nueva clase de activos al ecosistema DeFi en EOS. La demanda de zAssets implica la demanda del "share token" ZEOS, el cual se quema al crear los zAssets.

Es posible que en el futuro se añadan más protocolos DeFi, lo que podría añadir aún más utilidad al token ZEOS.

Por último, ZEOS es el token de gobernanza de la organización inteligente encargada del ecosistema. La tokenonomía de gobernanza aún no se han concretado. Los detalles aparecerán en una futura versión de este documento. Lo más probable es que incluya un requisito de "staking" para los miembros de la organización.

Se han planeado recompensas de staking atractivas para los holders de ZEOS, los cuales asumen el riesgo de los zAssets. Los posibles flujos de ingresos son:

- ***tasas de transacciones privadas sin cuenta EOS***
- ***tasas de creación/quema de zAssets***
- ***tasas del DEX de ZEOS***
- ***tasas de otros posibles protocolos DeFi basados en ZEOS (como préstamos)***

Mientras que el 90% de los ingresos podría distribuirse entre los holders de ZEOS, el 10% podría ser custodiado por la organización inteligente para financiar el desarrollo. Aunque todavía no se ha concretado la tokenonomía para el staking.

El suministro total de tokens ZEOS en circulación, así como el suministro total de todos los zAssets en circulación, es totalmente transparente y auditabile en cualquier momento. La cantidad de tokens ZEOS creados o quemados a cambio de zAssets es rastreada por el contrato, y por tanto, también es auditabile.

AIRDROP

El suministro inicial de tokens ZEOS asciende a casi mil millones. Mientras que el 90% de los tokens se han distribuido a los titulares de tokens PEOS, el 10% del suministro se ha mantenido como incentivo para el desarrollador.

Todas las cuentas de EOS que tuvieran al menos 0,1 PEOS el 24 de diciembre de 2021 recibieron 4,76 veces la cantidad de tokens ZEOS a través de un airdrop el 25 de diciembre de 2021. Las siguientes cuentas se incluyeron en la instantánea de tokens PEOS, pero se excluyeron porque la comunidad las identificó como cuentas de exchanges, block.one o cuentas del equipo PEOS. Los tokens ZEOS sobrantes por resultado de esta exclusión se distribuyó entre los holders de tokens PEOS, lo que llevó a la elevada proporción de casi 5 ZEOS por cada PEOS.

- hitbtc payout
- eoshoowallet
- binancecleos
- hitbtc payins
- newdexpublic
- hotbitioeos2
- otcbtcdotcom
- bitfinexdep1
- zbeoscharge1
- krakenkraken
- qpalmwoskxg
- heztanrqgene
- eospstotoken
- peosmarketin
- okbtothemoon
- peosteamfund
- bitfinexcw55
- thepeostoken
- okexoffiline
- fepxecwzm4lt
- binancecold1
- wlqdprkffody
- b1

ORGANIZACIÓN INTELIGENTE



La organización inteligente de ZEOS gobierna el ecosistema. La gobernanza significa esencialmente la propiedad, la gestión y el mantenimiento de todos los contratos inteligentes y sus correspondientes cuentas EOS, respectivamente. Inicialmente son "thezeostoken" y "thezeosproxy".

Esto implica un control total por parte de la organización sobre todos los ajustes del protocolo, recursos y activos bloqueados.

La organización inteligente es responsable del mantenimiento de los contratos inteligentes de ZEOS. Le corresponde a la organización adoptar las actualizaciones de código abierto de los repositorios de ZEOS oficiales de GitHub. El trabajo de la organización inteligente de ZEOS es configurar los contratos inteligentes y los protocolos de ZEOS de forma responsable para maximizar el valor del token ZEOS. Estas tareas incluyen, pero no se limitan, a:

- fijar las primas que se pagan a los mineros, junto a las tasas de gas y el fondo de ZEOS.
- establecer L_{min} y L_{max} de forma responsable
- añadir/eliminar nuevos tipos de zAssets al protocolo así como los correspondientes oráculos de precio.

Como propietaria del contrato del token ZEOS, la organización es también responsable de los recursos y paquetes DAPP reservados por el protocolo. En particular, la gobernanza es necesaria para añadir o eliminar paquetes DSP. Idealmente, se irán añadiendo más DSPs para descentralizar aún más la VRAM, VCPU y los servicios de oráculos a medida que la red DAPP crece.

En una versión futura de este documento se ofrecerán más detalles sobre la estructura y la configuración exacta de la organización inteligente. Lo más probable es que los miembros deban aportar una cantidad de ZEOS a la organización para participar en la gobernanza. El CAD de Vigor podría servir de modelo para la organización inteligente de ZEOS. La democracia fractal podría adoptarse como mecanismo de consenso para los miembros de la organización.