# Dominando Zcash

Maxime Desalle

2026-01-12

Figure 1: Concierto "Oda a la Libertad" de Leonard Bernstein, realizado el día de Navidad de 1989 para celebrar la caída del Muro de Berlín. La orquesta estuvo compuesta por miembros que representaban a los dos Estados alemanes y a las cuatro potencias ocupantes del Berlín de posguerra. El concierto fue transmitido en vivo a una audiencia estimada de 100 millones de personas en más de veinte países. La victoria de la libertad, la democracia y el capitalismo sobre la opresión, el totalitarismo y el comunismo, representada en la imagen.

---

*Con profundo agradecimiento a Giulia Mouland por sus comentarios y revisión editorial, a Arjun Khemani por su apoyo, y a César Oswaldo Navarro por la traducción al español.*

---

---

Las contribuciones a este artículo son más que bienvenidas en GitHub mediante solicitudes de incorporación de cambios.

---

## 1. Introducción

A menos que utilices dinero en efectivo, la información sobre cada compra que realizas es registrada y almacenada indefinidamente. No importa qué sea, ni cuán sensible sea. La infraestructura que impulsa el comercio, tanto fuera de línea como en línea, se ha convertido efectivamente en un aparato de vigilancia del que es casi imposible escapar.

Cuando Bitcoin fue lanzado por primera vez, existía la esperanza de que pudiera solucionar esto, pero lamentablemente no ha sido así. De hecho, contrariamente a lo que muchas personas creen, Bitcoin es increíblemente transparente, ya que cada transacción realizada queda almacenada permanentemente y visible para todos. Es cierto que las billeteras digitales son seudónimas, pero para recibir BTC necesitas proporcionar tu dirección, lo que implica facilitar al remitente todo tu historial de transacciones y tu saldo. Además, servicios como Arkham han hecho que sea sencillo, incluso para el público en general, rastrear e identificar billeteras digitales.

Por esta razón, las autoridades toleran Bitcoin, ya que para los actores estatales las cadenas transparentes son, en muchos aspectos, preferibles a las monedas digitales que ellos mismos controlan (a menudo llamadas Monedas Digitales de Bancos Centrales, o CBDC, por sus siglas en inglés). Dado que no existe resistencia por parte de la población a usar Bitcoin, y no hay supervisión sobre cómo las autoridades utilizan los datos de cadena, esto ofrece a los actores estatales una visibilidad perfecta para rastrear todo, con total impunidad.

En algunos aspectos, Bitcoin es en realidad peor que el sistema bancario que buscaba reemplazar. Al menos los registros bancarios son privados para el público en general; Bitcoin no lo es.

Es por esta razón que Zcash adopta un enfoque diferente: ofrecer privacidad por defecto, en lugar de transparencia por defecto. Esto significa que cuando realizas una transacción protegida de Zcash, el remitente, el destinatario y el monto de la transacción están todos encriptados. La red verifica que la transacción sea válida, verificando que tengas los fondos y que no estás gastando más ZEC de los que posees, pero no tiene acceso a ninguna información sobre la transacción en sí.

**Note** ZEC es el símbolo o ticker de Zcash,

de la misma manera que BTC lo es para Bitcoin.

Al principio, cuando uno lo piensa, esto parece imposible. ¿Cómo se puede demostrar que algo es verdadero sin revelar aquello que se está demostrando? La respuesta son las pruebas de conocimiento cero, específicamente una construcción llamada *zk-SNARKs*. La explicación de zk-SNARKs en este artículo se mantendrá simple y accesible para el lector general, ya que comprenderlas a fondo requiere una base considerable en álgebra y en esquemas de compromiso, lo cual queda fuera del alcance de este artículo.

También abordaremos los orígenes de Zcash en la criptografía académica, la filosofía que lo moldeó y el protocolo tal como existe hoy en día.

Algunas partes de este estudio integral sobre Zcash serán más técnicas. Aunque he intentado hacer todo lo más claro y accesible posible para todos, si tienes dificultades con ciertos conceptos, recomiendo preguntar a un modelo de lenguaje (LLM) para obtener aclaraciones o simplemente saltarte esa parte y volver a ella más tarde. Si eso no funciona, no dudes en comunicarte con cualquier pregunta.

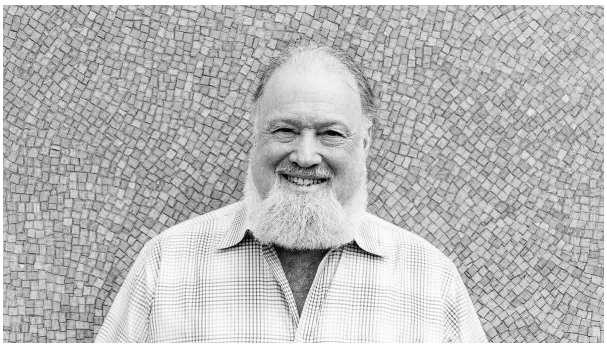


Figure 2: David Chaum, pionero de la criptografía.

## 2. Orígenes

### 2.1 David Chaum y el Nacimiento del Dinero Digital

La idea del dinero digital privado está lejos de ser nueva; de hecho, se remonta a 1982. David Chaum, quien en ese momento era candidato a doctorado en ciencias de la computación, publicó un artículo titulado *"Firmas ciegas para pagos no rastreables."*

La idea central de este artículo era simple y elegante: un banco podía firmar un token digital sin ver su contenido, del mismo modo en que podrías firmar el exterior de un sobre sellado. Luego, cuando el token se gastaba, el banco podía verificar su validez mediante su propia firma, pero no sería capaz de vincular el gasto con el retiro.

Posteriormente, en 1989, David Chaum fundó DigiCash, una empresa creada para comercializar esta idea. El producto se llamaba ecash y permitía a los usuarios retirar tokens digitales de sus cuentas bancarias y gastarlos en comercios sin dejar un rastro que conectara al comprador con la compra. Varios bancos probaron esta tecnología, entre ellos Deutsche Bank y Credit Suisse.

Lamentablemente, DigiCash no tuvo éxito; el momento no era el adecuado. Recordemos que esto fue creado antes de que el comercio por internet se generalizara y antes de que las personas comprendieran la importancia de la privacidad en línea. La empresa se declaró en bancarrota en 1998, pero con ecash, Chaum había demostrado que el dinero digital privado era posible.

### 2.2 Los Cypherpunks

Poco después, un tipo diferente de movimiento comenzó a tomar forma. En 1992, un grupo de criptógrafos, hackers y libertarios empezó a reunirse en el área de la Bahía de San Francisco y a comunicarse mediante una lista de correo electrónico. Se llamaron a sí mismos los *Cypherpunks*.

Los Cypherpunks no eran académicos que escribían artículos; eran ideólogos que escribían código. Su premisa fundamental era que, en la era digital, la privacidad no sería otorgada por los gobiernos ni por las corporaciones; en cambio, tendría que ser construida, implementada y defendida por los propios individuos utilizando herramientas criptográficas. En 1993, el miembro del grupo Eric Hughes cristalizó esta idea en *Un manifiesto de un Cypherpunk*:

*"La privacidad es necesaria para una sociedad abierta en la era electrónica.... No podemos esperar que los gobiernos, las corporaciones u otras grandes organizaciones impersonales nos otorguen privacidad por su beneficencia.... Debemos defender nuestra propia privacidad si esperamos tener alguna.... Los Cypherpunks escriben código."*

La lista de correo se convirtió en un crisol de ideas que darían forma a las siguientes tres décadas de desarrollo criptográfico. Entre sus miembros se encontraban Julian Assange (antes de WikiLeaks), Hal Finney (quien más tarde recibiría la primera transacción de Bitcoin), Nick Szabo (quien propuso *bit gold*, un precursor conceptual de Bitcoin) y Wei Dai (cuya propuesta *b-money* fue citada por Satoshi Nakamoto).

En 1997, otro miembro, Adam Back, inventó *Hashcash*, *el sistema de Prueba de Trabajo* (PoW, por sus siglas en inglés) que posteriormente sería adoptado por Bitcoin.

Los cypherpunks no construyeron una criptomoneda exitosa... ¿o sí? La creación de Bitcoin se atribuye al seudónimo Satoshi Nakamoto, de quien se rumorea que fue un desarrollador o un grupo de desarrolladores vinculados al movimiento cypherpunk, y que no ha estado activo durante más de una década. En cualquier caso, lo que sí sabemos con certeza es que los cypherpunks construyeron la cultura, las herramientas y el marco intelectual que hicieron posible la existencia de monedas privadas.

> **Note** Poco después de que se publicara este artículo, Zooko Wilcox, cofundador de Zcash, se puso en contacto señalando lo siguiente: - ¡Él *estaba* en la lista de correo de los cypherpunks! Lo que significaría que los cypherpunks *sí crearon* una criptomoneda exitosa. Mea culpa por esa omisión. - Zooko se hizo amigo allí de varios de los fundadores, entre ellos Tim May, quien fundó el movimiento criptoanarquista; Eric Hughes, quien escribió *"Un manifiesto de un Cypherpunk"*, como se mencionó anteriormente; Bram Cohen, creador del protocolo BitTorrent, con quien trabajó en una empresa emergente enfocada en cadenas de hash seguros; y John Gilmore, cofundador de la Electronic Frontier Foundation. - La lista de correo de los cypherpunks fue fundamental en su desarrollo. Por ejemplo, John Gilmore se convirtió en amigo, mentor e inspiración.

### 2.3 Bitcoin: El Compromiso Equivocado

El 31 de octubre de 2008, Satoshi Nakamoto publicó un artículo en una lista de correo sobre criptografía titulado *"Bitcoin: Un Sistema de Dinero Electrónico entre Pares."* El artículo describía una solución a un problema que había afectado a los diseñadores de monedas digitales durante décadas: ¿cómo evitar el doble gasto sin depender de una autoridad central?.

La respuesta propuesta por Satoshi fue la blockchain: un libro mayor público mantenido por una red descentralizada de mineros, asegurado mediante o PoW. Fue una idea brillante, ¡y funcionó! Bitcoin se lanzó en enero de 2009, y por primera vez las personas pudieron transferir valor a través de internet sin bancos, intermediarios ni permisos.

> **Note** Más adelante en este artículo explicaremos qué son los mineros y la Prueba de Trabajo (PoW, por sus siglas en inglés) y cómo funcionan en el contexto de Zcash.

Sin embargo, había un problema evidente, como se mencionó antes: Bitcoin no es privado. La blockchain es completamente pública por diseño: cada transacción, cada dirección y cada saldo son visibles para cualquiera que esté interesado. Satoshi reconoció este problema en el artículo, sugiriendo que los usuarios podrían preservar parte de su privacidad utilizando nuevas direcciones para cada transacción. No obstante, esta era una mitigación débil, ya que las direcciones pueden agruparse, los grafos de transacciones pueden analizarse y las identidades del mundo real pueden vincularse a través de intercambios, comerciantes y metadatos.

Más tarde, Nakamoto también reconoció que una forma de Bitcoin que preservara la privacidad permitiría una implementación más limpia del protocolo, pero en ese momento no podía imaginar cómo lograrlo utilizando pruebas de conocimiento cero.

El problema de la privacidad, de manera problemática, permaneció ignorado durante años. Los primeros usuarios de Bitcoin asumían que el seudonimato era prácticamente lo mismo que el anonimato, pero estaban equivocados. A comienzos de la década de 2010, investigadores demostraron que el análisis de la blockchain podía desanonimizar a los usuarios con alta precisión. Empresas como Chainalysis, fundada en 2014, convirtieron esto en un negocio al vender análisis forense de blockchain a agencias de aplicación de la ley, intercambios e incluso gobiernos.

Bitcoin había resuelto el problema del doble gasto, pero había empeorado el problema de la privacidad.

### 2.4 Zerocoin: El Intento de Mejora

En 2013, Matthew Green, criptógrafo de la Universidad Johns Hopkins, junto con dos estudiantes de posgrado, Ian Miers y Christina Garman, publicaron *"Zerocoin,"* un artículo que proponía una solución al problema de Bitcoin.

> **Note** Un dato curioso compartido por Zooko Wilcox después de la publicación de este artículo: Ian Miers y Christina Garman posteriormente se convirtieron en científicos fundadores de la Zcash Company (ver sección 2.6), y Christina Garman más tarde también se unió a la Junta Directiva.

Su idea era añadir una capa de privacidad sobre Bit-

coin, de modo que los usuarios pudieran convertir sus bitcoins en zerocoins, tokens anónimos sin historial de transacciones. Posteriormente, cuando quisieras gastarlos, podrías volver a convertirlos en Bitcoin. El proceso de conversión se basaba en técnicas criptográficas conocidas como pruebas de conocimiento cero, que permiten demostrar que posees un zerocoin válido sin revelar su origen.

Zerocoin funcionaba en teoría, pero tenía problemas. Primero, las pruebas eran grandes, aproximadamente dos órdenes de magnitud más grandes que los pocos cientos de bytes requeridos para una transacción normal de Bitcoin. Segundo, la criptografía también era limitada: podías demostrar la propiedad, pero no podías ocultar los montos de las transacciones. Tercero, y lo más crítico, requería que Bitcoin lo adoptara como un cambio en el protocolo, pero la cultura de desarrollo conservadora de Bitcoin hacía que esto fuera poco probable.

La comunidad de Bitcoin debatió Zerocoin y finalmente decidió rechazarlo. La propuesta nunca llegó a incorporarse al protocolo.

### 2.5 Zerocash: La Reconstrucción

En 2014, se publicó un nuevo artículo. La lista de autores se amplió para incluir a Eli Ben-Sasson y Alessandro Chiesa, criptógrafos que habían estado trabajando en una nueva generación de pruebas de conocimiento cero, además de Eran Tromer y Madars Virza.

El artículo se titulaba *"Zerocash: Pagos Anónimos Descentralizados a partir de Bitcoin."* A pesar de lo que su título podría sugerir, no era simplemente una extensión de Bitcoin, sino un rediseño completo.

La innovación clave fue el uso de zk-SNARKs, que significa *Argumentos de Conocimiento Sucintos No Interactivos de Conocimiento Cero.* Estas eran pruebas de conocimiento cero que eran pequeñas (unos pocos cientos de bytes), rápidas de verificar (milisegundos) y lo suficientemente expresivas como para demostrar afirmaciones complejas sobre datos ocultos. Con zk-SNARKs se puede demostrar no solo que posees una moneda válida, sino también que toda una transacción es válida. Esto no es trivial: significa que el sistema puede verificar que los montos de la transacción son correctos, que no hay doble gasto, etc., todo sin revelar el remitente, el destinatario ni el monto.

Sin embargo, había un inconveniente: los zk-SNARKs requerían una configuración confiable. Alguien tenía que generar un conjunto de parámetros públicos que el sistema usaría permanentemente. Pero si esa persona conservaba los valores secretos utilizados para generar esos parámetros, lo que se conoce como residuos tóxicos, podría crear monedas falsificadas sin que nadie lo detectara. Aunque esto era una preocupación seria, los investigadores creían que podía evitarse mediante un diseño cuidadoso de la ceremonia de configuración.

### 2.6 El Bloque Génesis

Zooko Wilcox llevaba décadas trabajando en el ámbito de la privacidad y la criptografía. Había trabajado en DigiCash en la década de 1990 y también participó en proyectos de almacenamiento descentralizado con fuertes propiedades de privacidad, como Tahoe-LAFS. Por eso, cuando se publicó el artículo sobre Zerocash, encajó inmediatamente con sus intereses.

En 2016, Wilcox fundó la *Zcash Company*, que posteriormente fue renombrada como *Electric Coin Company*, y reunió un equipo para convertir Zerocash en una criptomoneda lista para su uso en producción. Los autores académicos mencionados anteriormente se unieron como asesores y colaboradores en el proyecto.

El problema de la configuración confiable mencionado antes requería una solución creativa. El equipo diseñó una compleja ceremonia de computación multipartita: seis participantes, todos en diferentes lugares del mundo, aportarían aleatoriedad para generar los parámetros públicos. Mientras al menos uno de los participantes destruyera su entrada secreta, los residuos tóxicos serían irrecuperables. La ceremonia tuvo lugar a finales de 2016, con participantes que incluían a Peter Todd, desarrollador de Bitcoin Core, y periodistas que documentaron el proceso. Se realizó un trabajo exhaustivo para asegurarse de que la ceremonia no fuera comprometida, tal como se describe aquí.

El 28 de octubre de 2016, se minó el bloque génesis de Zcash. Por primera vez, una criptomoneda en funcionamiento ofrecía privacidad criptográfica real. Treinta y cuatro años después del primer artículo de David Chaum, el sueño de un dinero digital imposible de rastrear estaba funcionando en una red activa.

## 3. ¿Qué es Zcash?

### 3.1 Introducción a Bitcoin

> **Tip** Si ya entiendes cómo funciona Bitcoin, puedes saltar esta sección; está pensada

para lectores que no están familiarizados con su funcionamiento interno.

Bitcoin es esencialmente un sistema de pagos sin un operador central. No hay un banco, una empresa ni un servidor único al que se pueda señalar como responsable. Su mecanismo descentralizado funciona gracias a miles de computadoras alrededor del mundo que mantienen copias idénticas de un libro mayor compartido, llamado blockchain, y siguen un conjunto de reglas para mantenerlo sincronizado.

Figure 3: Hiperinflación en la República de Weimar. Los billetes habían perdido tanto valor que se utilizaban como papel tapiz.

La blockchain es una estructura de datos de solo adición, y literalmente es una cadena de bloques. Esto significa que puedes agregar nuevas entradas (bloques), pero nunca modificar o eliminar las antiguas. Cada nuevo bloque contiene las transacciones realizadas en la red en el momento en que el bloque fue creado. Además, cada bloque hace referencia al bloque anterior, lo que da lugar a la formación de una cadena. Si quisieras cambiar una transacción del pasado, tendrías que reescribir todos los bloques posteriores, lo cual se vuelve computacionalmente imposible una vez que ha pasado suficiente tiempo. Más adelante veremos por qué ocurre esto.

**Llaves y Propiedad** Bitcoin utiliza criptografía de clave pública para sus billeteras digitales. Cuando 'creas una billetera', lo que realmente estás haciendo es generar un par de claves: una clave privada (un número aleatorio grande, que se mantiene en secreto) y una clave pública correspondiente (derivada matemáticamente de la clave privada). Una dirección de Bitcoin se deriva de una clave pública mediante hashing y codificación.

> **Example** Aquí hay un ejemplo de cómo se ven en la práctica (abreviado usando ...):
>
> - Clave privada: `1E99423A4ED27608A15...` ↪ `E6E9F3A1C2B4D5F6A7B8C9D0`
> - Clave pública: `03F028892BAD7ED57D2F` ↪ `...3A6A6C6E7F8C9D0A1B2C3D4E5F607182`
> - Dirección de Bitcoin: `1` ↪ `BoatSLRHtKNngkdXEeobR76b53LETtpyT`

La clave privada te permite firmar mensajes, mientras que la clave pública permite que cualquiera verifique que una firma proviene de la clave privada correspondiente, sin revelar la clave privada en sí. Esta criptografía es lo que mantiene la privacidad de la clave privada, ya que puedes firmar un mensaje autorizando una transferencia usando tu clave privada, y la red puede verificar tu firma utilizando tu clave pública, sin ver nunca tu clave privada.

Una conclusión importante aquí es que las billeteras

digitales en realidad no "guardan" BTC en un sentido real. No existe un archivo en tu computadora que contenga monedas. En cambio, la blockchain mantiene el registro de qué direcciones controlan qué salidas, y tu billetera digital es simplemente una herramienta de firma: almacena tus claves privadas y las utiliza para autorizar transacciones. Si pierdes tus claves privadas, pierdes el acceso a tus fondos; no porque las monedas hayan desaparecido, sino porque ya no puedes demostrar que eres su propietario.

**Transacciones y UTXO** Las transacciones son la forma en que se mueve el valor en Bitcoin. Cuando envías BTC, estás publicando un mensaje firmado que, en esencia, dice: "Autorizo la transferencia de estas monedas a esta dirección." Pero entonces surge la pregunta: ¿qué son exactamente esas monedas?

Bitcoin no registra saldos. No existen entradas en una base de datos que digan "la dirección X tiene 3.5 BTC". En su lugar, Bitcoin utiliza *Salidas de Transacción No Gastadas* (UTXO, por sus siglas en inglés). Cada transacción consume salidas existentes y luego crea nuevas. Las salidas que controlas pero que aún no has gastado son tus UTXO. Esto significa que tu "saldo" es simplemente la suma de todas tus salidas no gastadas. No hay un conteo continuo de monedas, sino una colección de unidades discretas que controlas.

> **Example** Ejemplo: Aquí hay un ejemplo rápido. Imagina que tienes un billete de $20 y quieres comprar algo que cuesta $12. Obviamente no puedes partir el billete por la mitad, así que entregas los $20 y recibes $8 de cambio.
>
> Los UTXO funcionan de la misma manera. Si posees una salida de 5 BTC y quieres enviar 3 BTC a alguien, necesitas consumir completamente esa salida de 5 BTC y crear dos nuevas salidas a partir de ella: 3 BTC para el destinatario, y 2 BTC que regresan a ti como cambio. Tu salida original de 5 BTC ahora está "gastada" y nunca podrá utilizarse nuevamente.

Como resultado, una transacción de Bitcoin es una estructura de datos que contiene cierta información de metadatos, además de: 1. **Entradas**: Referencias a los UTXO que estás gastando, junto con firmas que demuestran que los controlas. 2. **Salidas**: Nuevos UTXO que se crean, cada uno bloqueado con la clave pública del destinatario.

Los nodos verifican que las entradas existan, no hayan sido gastadas todavía y tengan firmas válidas. Si todo está correcto, la transacción se retransmite por la red y espera a ser incluida en el bloque de un minero.

> **Example** Aquí se muestra cómo se ve una transacción en la práctica (los hashes y direcciones están abreviados con ...):

```
{
  "txid": "c1b4e693...cbdc5821e3",
  "inputs": [
    {
      "prev_txid": "7b1eabe...98a14f3f",
      "output_index": 0,
      "signature": "304402204e4...1
    ↪ a8768d1d09",
      "pubkey": "0479be66...ffb10d4b8"
    }
  ],
  "outputs": [
    {
      "amount": 3.0,
      "script": "OP_DUP OP_HASH160 89...ba
    ↪  OP_EQUALVERIFY OP_CHECKSIG"
    },
    {
      "amount": 1.99,
      "script": "OP_DUP OP_HASH160 12...78
    ↪  OP_EQUALVERIFY OP_CHECKSIG"
    }
  ]
}
```

> Cada entrada apunta a la salida de una transacción anterior, haciendo referencia a su ID de transacción y su índice, y cada salida especifica un monto. La firma demuestra que controlas la clave privada. La diferencia de 0.01 BTC entre la entrada de 5 BTC y las salidas de 3 BTC + 1.99 BTC corresponde a la comisión de la transacción, que es reclamada por el minero.

**Minería y Prueba de Trabajo (PoW)** Las transacciones no se confirman por sí solas. Permanecen en un área de espera en los nodos llamada mempool (pool de memoria) hasta que un minero las incluye en un bloque. La minería es el proceso mediante el cual se agregan nuevos bloques a la cadena, y está diseñado para ser costoso. Esto no es un error, sino una característica, como veremos en un momento.

El problema que la minería intenta resolver es el siguiente: en una red descentralizada sin autoridad central, ¿quién decide qué transacciones son válidas? ¿Quién decide su orden? Si aparecen dos transacciones en conflicto, por ejemplo, cuando alguien intenta gastar las mismas monedas dos veces, ¿quién resuelve ese conflicto?

La solución de Bitcoin es la siguiente: para crear un bloque válido, un minero debe encontrar un número llamado *nonce* de tal forma que, cuando el encabezado del bloque (que contiene el hash del bloque anterior, una marca de tiempo, etc.) se combine con ese nonce y se aplique una función hash, el hash resultante quede por debajo de un determinado valor objetivo. Dado que los hashes criptográficos son esencialmente aleatorios, no existe una forma directa de encontrar un nonce válido excepto probando una y otra vez. Por eso, los mineros realizan miles de millones de intentos por segundo.

> **Example** Por ejemplo, imagina un bloque como una página con información fija y un único número ajustable en ella (el nonce). Supongamos que comenzamos a contar el nonce desde `0`.
>
> Una computadora convierte toda la página en un único número de salida llamado hash. Un hash puede ser algo como `6` o `03a5b20` ↪ ; en última instancia solo es un número (sí, `03a5b20` también es un número, porque equivale a `3,824,416` en decimal). Recuerda que el nonce es el único número ajustable en la página, y cambiar solo el nonce produce un hash completamente diferente cada vez.
>
> La red exige que el hash esté por debajo de un valor umbral fijo, y si no lo está, el minero cambia el nonce e intenta nuevamente. Finalmente, el nonce se acepta cuando el hash cumple con el requisito del umbral.
>
> Por ejemplo, imagina un caso donde el valor umbral es `5`. El minero tiene su página de información y comienza con un nonce de `0`. Si la computadora devuelve `6`, que es mayor que `5`, el minero lo intenta otra vez, ahora con `1` como nonce. Si esta vez la computadora devuelve `4`, que es menor que `5`, entonces `1` se acepta como nonce válido.

La dificultad se ajusta cada 2,016 bloques (aproximadamente cada dos semanas), manteniendo un tiempo promedio de diez minutos por bloque. Si los bloques se generan demasiado rápido, el valor objetivo disminuye, lo que hace que el problema sea más difícil. Si, por el contrario, los bloques se generan demasiado lentamente, el valor objetivo aumenta. Este ajuste de dificultad es la razón por la que la tasa de bloques de Bitcoin se mantiene estable, incluso cuando la potencia total de minería fluctúa.

> **Example** Así es como se ve un bloque en la práctica:
>
> ```
> {
>   "header": {
>     "version": 536870912,
>     "prev_block_hash": "0000000...
>      ↪ de0e5c842",
>     "merkle_root": "8b30c5ba1...1
>      ↪ e0d5f8a2c1",
>     "timestamp": 1701432000,
>     "target": "0000004f2c0...0000000",
>     "nonce": 2834917243
>   },
>   "transactions": [
>     {
>       "txid": "3a1b9c7e...7e8f9a0b1c",
>       "inputs": [{ "coinbase": "03a5b20
>      ↪ ...706f6f6c" }],
>       "outputs": [{ "amount": 6.25, "
>      ↪ script": "OP_HASH160
>         f1c3...4c6a8 OP_EQUAL" }]
>     },
>     { "txid": "c1b4e...5821e3" },
>     { "txid": "7d5e8...b5c6d7e" }
>   ]
> }
> ```
>
> El encabezado del bloque es lo que se somete al proceso de hash. Los mineros incrementan el nonce repetidamente hasta que se cumpla la condición: `SHA256(SHA256(header))` ↪ `< target`, es decir, aplican dos veces la función hash SHA-256 al encabezado hasta que el hash resultante sea menor que el valor objetivo. La primera transacción del bloque siempre es la transacción "coinbase", que crea nuevas monedas y paga la recompensa al minero.

Una vez que un minero encuentra un nonce válido, difunde el bloque a la red, y los otros nodos lo verifican, comprobando que: el hash cumple con el objetivo, todas las transacciones son válidas, y el minero no creó más monedas de las permitidas. Si todo es válido, los nodos agregan el bloque a su cadena y empiezan a trabajar en el siguiente. El minero gana una recompensa en forma de bitcoins recién creados, además de las comisiones de las transacciones incluidas en el bloque.

Entonces, ¿cómo evita este sistema que se reescriba el pasado? Porque el hash de cada bloque forma parte del bloque siguiente. Esto significa que cambiar una sola transacción modifica el hash del bloque y rompe inmediatamente todos los bloques que vienen después.

> **Example** Imagina que tienes dos bloques consecutivos, A y B. El hash de A es `5` y el hash de B es `6`. Si cambias una transacción en A, entonces el hash de A cambia, y eso

obliga a que también cambie el hash de B. Esto ocurre porque el hash de B incluye el hash de A, ya que B viene después de A y contiene su hash. Por lo tanto, el hash de B ya no será `6` si se modifica una transacción en A.

Para volver a hacer válida la cadena, un atacante tendría que rehacer la Prueba de Trabajo (el proceso de encontrar un nonce por debajo de un determinado valor objetivo) no solo para ese bloque, sino también para todos los bloques posteriores. Mientras tanto, los mineros honestos siguen minando y extendiendo la cadena "real" con nuevos bloques. Además, Bitcoin sigue la cadena que tenga la mayor cantidad acumulada de Prueba de Trabajo, lo que dificulta enormemente que un atacante tenga éxito.

Por lo tanto, para que un ataque tenga éxito, el atacante necesitaría controlar el 51 % del poder de minería, para eventualmente alcanzar y convertirse en la cadena "real". El poder de minería también se conoce como *poder de hash*, ya que los mineros básicamente calculan funciones hash de información innumerables veces por segundo, todos los días.

**La Compensación de la Transparencia** Es importante señalar que para que este sistema funcione sin una autoridad central, todos deben poder verificar todo. Cada nodo revisa cada transacción comparándola con todo el historial de la cadena, cada UTXO es rastreado y cada firma es validada.

Esto tiene un costo en términos de privacidad, ya que cada transacción y el saldo de cada dirección son públicos. Todo el flujo de fondos, desde el bloque génesis de 2009 hasta el bloque más reciente minado, es visible para cualquiera que descargue la blockchain.

Así, Bitcoin resolvió el problema del dinero digital sin necesidad de confianza, pero no resolvió el problema del dinero digital privado sin confianza. Ahí es donde entra Zcash.

### 3.2 Bitcoin, Pero Privado

Zcash es, en esencia, similar a Bitcoin, pero con la adición de encriptación. De hecho, muchas personas lo describen como *Bitcoin encriptado*, aunque en realidad es una criptomoneda completamente diferente.

La economía de Zcash es casi idéntica a la de Bitcoin, por lo que si entiendes la política monetaria de Bitcoin, también entiendes la de Zcash. Zcash tiene un límite máximo de 21 millones de ZEC, del mismo modo que Bitcoin tiene un límite máximo de 21 millones de BTC. Las nuevas monedas entran en circulación mediante recompensas de minería, que se reducen a la mitad aproximadamente cada cuatro años, al igual que en Bitcoin.

El mecanismo de consenso también es Prueba de Trabajo, aunque Zcash utiliza Equihash en lugar del sistema basado en SHA256 que usa Bitcoin para la minería. Algo interesante de Equihash es que fue diseñado con el objetivo explícito de resistir los ASIC especializados que dominan la minería de Bitcoin, con el fin de mantener la minería accesible para personas con GPUs de consumo. Esta elección reflejaba el énfasis inicial de Zcash en la descentralización, aunque hoy en día ya no funciona del todo, porque ahora también existen ASICs para Equihash.

> **Note** ASIC, por sus siglas en inglés significa *Circuito Integrado de Aplicación Específica*. Puedes pensar en ellos como computadoras diseñadas específicamente para minar criptomonedas. Existen ASICs especializados en distintos algoritmos, por ejemplo, para minería con SHA256, Equihash, entre otros.
>
> Los ASICs calculan hashes de información (bloques de transacciones) continuamente durante todo el día, con la esperanza de encontrar un hash que esté por debajo del valor objetivo de la red.

En su funcionamiento interno, Zcash utiliza el mismo modelo de transacciones UTXO que Bitcoin.

Sin embargo, Zcash difiere de Bitcoin en lo que se puede hacer con esos UTXO. Bitcoin tiene un solo conjunto de pools: la cadena pública, mientras que Zcash tiene varios, divididos entre el pool transparente y los pools protegidos, pero, ambos pools utilizan ZEC como moneda, y es posible mover fondos entre ellos. El pool transparente funciona exactamente igual que en Bitcoin: las direcciones comienzan con la letra t, las transacciones son completamente visibles, y cualquiera puede rastrear el flujo de los fondos.

Los pools protegidos son completamente diferentes y exclusivos de Zcash. Existen tres pools: *Sprout*, *Sapling* y *Orchard*, Orchard es el más reciente y avanzado. Sprout y Sapling actualmente casi no se utilizan, ya que provienen de actualizaciones anteriores de la red y dependen de *configuraciones de confianza*, algo que Orchard ya no requiere. Más adelante en el artículo explicaremos esto con mayor detalle. Las direcciones protegidas comienzan con z, y las transacciones no revelan nada sobre el remitente, el destinatario o el importe.

> **Note** En adelante, nos referiremos a los

pools de Zcash como el pool transparente y el pool protegido, como si hubiera varios pools protegidos, pero en la práctica se consideran un todo unificado y Orchard es el que se utiliza principalmente en la actualidad.

El pool transparente existe por motivos de compatibilidad y opcionalidad. Algunos usuarios quieren auditabilidad, algunas aplicaciones incluso la requieren, y las bolsas suelen utilizar direcciones transparentes para cumplir con la normativa. En este caso, la transparencia es una característica, no un error, y la confianza de Zcash en el cifrado para la privacidad en el pool protegido no se ve afectada por el uso de los pools transparentes.

Debemos considerar el pool transparente y el pool protegido como dos sistemas totalmente independientes que no se afectan entre sí. A menudo se critica erróneamente la función de transparencia de Zcash por considerar que reduce de alguna manera su privacidad, pero eso es falso. El conjunto de anonimato de Zcash es matemáticamente independiente de la cantidad de ZEC que se encuentra en direcciones transparentes. Por lo tanto, incluso si el 99 % de los ZEC fueran transparentes, la privacidad del 1 % protegido solo vendría determinada por el propio pool protegido.

### 3.3 El Problema Fundamental

En Bitcoin, validar una transacción es sencillo. Se comprueba que las entradas existen y no se han gastado antes, que las firmas son válidas y que las salidas no superan las entradas. Toda la información necesaria para verificar estas condiciones se encuentra en la cadena de bloques, visible para todos.

Esa transparencia es lo que hace que Bitcoin sea fiable. No es necesario confiar en nadie porque se puede verificar todo por uno mismo. Si se quisiera, se podría incluso ejecutar un nodo para lograr la máxima confianza. Sin embargo, esto es también lo que convierte a Bitcoin en una herramienta de vigilancia, ya que los mismos datos que permiten la verificación son los que permiten el seguimiento.

Zcash quiere ambas cosas: verificación sin confianza y privacidad, pero ambas parecen contradecirse entre sí. ¿Cómo puede la red verificar que una transacción es válida si no puede verla?

Piensa en lo que realmente requiere la validación:

1. Las entradas existen, ya que no se pueden gastar monedas que no existen.
2. Las entradas no se han gastado antes, por lo que no hay doble gasto.
3. La autorización para gastar, ya que tú controlas la clave privada.
4. Las matemáticas funcionan y las salidas no superan las entradas.

En Bitcoin, los nodos y los mineros comprueban estos cuatro criterios examinando los datos. En Zcash, el remitente, el destinatario y la cantidad están encriptados, y los datos no son visibles. Entonces, ¿cómo se pueden comprobar estos criterios?

La respuesta es que Zcash no pide a los nodos y mineros que comprueben los datos. En su lugar, el remitente proporciona un zk-SNARK, una prueba criptográfica, que demuestra que la transacción es válida sin revelar ninguna de la información subyacente. Los mineros y los nodos no saben cuáles son las entradas, quién es el destinatario ni cuánto se transfiere, solo saben una cosa: la prueba es válida y, por lo tanto, la transacción es válida.

Parece una locura, ¡podemos verificar que una transacción financiera es válida sin verla!

En las siguientes secciones se explica por qué esto es posible, incluyendo cómo Zcash representa el valor y realiza un seguimiento de lo que se gasta, así como cómo las pruebas de conocimiento cero lo unen todo.

### 3.4 Notas Protegidas

Como se mencionó anteriormente, Bitcoin utiliza UTXO. El pool protegido de Zcash utiliza algo conceptualmente similar llamado notas; se puede pensar en las notas como UTXO encriptado.

Entonces, ¿qué es una nota? Una nota es un objeto cifrado que representa una cantidad específica de ZEC. Es una porción discreta de valor, al igual que los UTXO, pero a diferencia de estos, su contenido está oculto. Cuando se recibe ZEC protegido, se crea una nota. Cuando se gasta el Zec protegido, esa nota se consume y se crean nuevas notas para el destinatario y el cambio, si procede, exactamente igual que con los UTXO.

**Example** Así es como se ve una nota de Orchard después del descifrado:

```
{
  "addr": "u1pg2aaph7jp8rpf6...
    ↪ sz7nt28qjmxgmwxa",
  "v": 150000000,
  "rho": "0x9f8e7d6c5b4a...
    ↪ f8e7d6c5b4a39281706f5e4d3c2b1a0",
  "psi": "0x1a2b3c4d5e6f70...
    ↪ c4d5e6f708192a3b4c5d6e7f809",
```

```
  "rcm": "0x7a3b4c5d6e7b...
    ↪ d8e9f0a1b3d4e5f6a7b8c9d0e1f2a3b"
}
```

En este ejemplo, el campo de valor v muestra 1,5 ZEC (150 000 000 zatoshis). Los otros campos, `rho`, `psi` y `rcm`, se tratarán más adelante. Por ahora, basta con entender que son los que hacen posible la criptografía que respalda las notas de Zcash.

Las notas nunca se modifican, no hay actualización del saldo. Más bien, se crean, existen y se destruyen cuando se gastan. Si tienes 10 ZEC y gastas 3 ZEC, la nota original de 10 ZEC se consume por completo y se crean dos notas nuevas: 3 ZEC que se entregan al destinatario y 7 ZEC que se te devuelven, al igual que los UTXO.

La diferencia fundamental entre las notas de Zcash y los UTXO de Bitcoin es su visibilidad. Un UTXO de Bitcoin es público: todo el mundo puede ver su valor, cuándo se gasta, etc. Una nota de Zcash está cifrado: solo el propietario y cualquier persona con la que comparta su clave de visualización puede ver su contenido. La cadena de bloques almacena un compromiso criptográfico con el nota, no almacena el nota en sí.

**Example** La cadena de bloques nunca ve la nota descifrada. En Orchard, cada "acción" agrupa un gasto y una salida. Esto es lo que se registra realmente:

```
{
  "cv": "0x9a8b7c6d5...8
    ↪ d7e6f5a4b3c2d1e0f9a8b",
  "nullifier": "0x2c3d4e5f6a7b...
    ↪ d2e3f48e9f0a1b2c3d",
  "rk": "0x5e6f7a8b...5
    ↪ a6b7c8d9e0f1a2b3c4d5e6f",
  "cmx": "0x1a2b3c4d5e6f7...
    ↪ d3e4f5a6b7c8d9e0f1a2b",
  "ephemeralKey": "0x4d5e6f7a8b9...4
    ↪ f5a6b7c8d9e0f1a2b3c4d5e",
  "encCiphertext": "0x8f7e6d5c4b3...
    ↪ a29180f7e6d5c",
  "outCiphertext": "0x3c4d5e6f7a8...
    ↪ b9c0d1e2f3a4b5c"
}
```

Como puede ver, todo está encriptado. Más adelante repasaremos los detalles de cada campo.

Puede que se pregunte: si las notas están ocultas, ¿cómo sabe la red que existen? ¿O cómo sabe cuándo se han gastado? Aquí es donde entran en juego los compromisos y los anuladores.

### 3.5 Compromisos y Anuladores

El pool protegido de Zcash se enfrenta a dos problemas que Bitcoin resuelve de forma trivial mediante la transparencia:

1. **Demostrar que las notas existen:** Cuando alguien te envía ZEC protegidos, ¿cómo sabe la red que la nota es real?
2. **Evitar el Doble Gasto:** Cuando gastas una nota, ¿cómo sabe la red que no la has gastado antes?

La solución para Zcash es una combinación de dos mecanismos criptográficos: compromisos y anuladores.

**Compromisos** Un compromiso es un valor calculado mediante el hash de los campos de la nota. Así es como se ve en Orchard:

```
cmx = Hash(addr, v, rho, psi, rcm) = 0x1a2b3c4d...9
    ↪ ca6b7c8d9e0f1a2b
```

"Hash" denota la función hash utilizada. Tomamos los campos de la nota protegida, los introducimos en la función hash y esta devuelve un hash (en este caso 0x1a2b3c4d‹→ . . . 9ca6b7c8d9e0f1a2b).

Hay dos propiedades que lo hacen útil:

1.**Unidireccional:** dado el hash devuelto, 0x1a2b3c4d‹→ . . . 9ca6b7c8d9e0f1a2b, no se pueden recuperar los campos `addr`, `v`, `rho`, `psi` o `rcm`, y el contenido de la nota queda oculto. 2.**Resistente a colisiones:** no se pueden encontrar dos notas diferentes que produzcan el mismo compromiso, cada nota se asigna exactamente a un compromiso.

Cada vez que se crea una nota, su compromiso se añade al árbol de compromisos (un árbol Merkle) que contiene todos los compromisos de notas creados en la red.

**Info** Un árbol Merkle es una estructura de datos que permite demostrar que un elemento se encuentra en un conjunto grande sin revelar el elemento ni descargar todo el conjunto.

Así es como funciona. Comience con una lista de valores (en nuestro caso, compromisos de notas): `cm0 cm1 cm2 cm3`

Empareje los valores y aplique un hash a cada par:

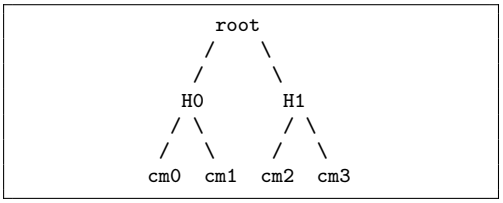
- `H0 = Hash(cm0, cm1)`
- `H1 = Hash(cm2, cm3)`

Ahora tiene dos hash. Empareje y vuelva a hacer el hash:

```
root = Hash(H0, H1)
```

Hasta ahora, hemos tomado pares de elementos del conjunto original y hemos combinado cada par utilizando una función hash. A continuación, agrupamos los hash resultantes en pares y los volvemos a hash, repitiendo este proceso capa por capa hasta llegar a un único hash final. Este valor final se denomina raíz hash o raíz Merkle.

Este hash raíz resume eficazmente todo el conjunto:




La propiedad clave de los árboles Merkle es que si se cambia cualquier hoja (compromiso), es decir, los valores `cm0`, `cm1`, etc., todos los hash superiores también cambian, hasta llegar a la raíz. La raíz actúa como la huella digital de todo el árbol, si se tiene la misma raíz, entonces se debe tener el mismo árbol.

Además, las pruebas Merkle proporcionan una forma eficaz de comprobar un elemento del árbol sin tener que comprobar todo el árbol.

Por ejemplo, para demostrar que cm1 está en el árbol no es necesario revelar todos los compromisos. Para ello, basta con proporcionar una ruta Merkle, es decir, los hash hermanos a lo largo del camino hasta la raíz. Para `cm1`, la ruta Merkle es `[cm0, H1]`.

Así es como un verificador podría comprobarlo: 1. Tome el primer elemento de `[cm0, H1]`, es decir, `cm0`, y hágalo hash con `cm1`, el elemento que queremos comprobar, lo que nos da `H0: Hash(cm0, cm1) = H0` 2. Hash la salida del primer paso (`H0`) con el siguiente elemento en `[cm0, H1]`, es decir, `H1`. Esto nos da el hash raíz: `Hash(H0, H1) = root`.

Si el resultado coincide con la raíz conocida, podemos concluir que `cm1` está en el árbol. Es importante destacar que el verificador nunca ve `cm2` ni `cm3`, ya que no son necesarios para la verificación.

El árbol de compromisos contiene todos los compromisos de notas protegidas que se han creado, lo que equivale a millones de hojas (compromisos). Por lo tanto, cuando gastas una nota, demuestras (dentro del zk-SNARK) que conoces un compromiso y la ruta Merkle válida hasta la raíz actual, sin revelar cuál es tu compromiso.

El árbol de compromisos se almacena por nodos, como parte del estado de la cadena que mantienen. Cada bloque introduce nuevos compromisos de notas que los nodos añaden a su copia local del árbol, actualizando la raíz en consecuencia. La raíz actual, conocida como el ancla, es a lo que hacen referencia las transacciones cuando demuestran su pertenencia.

**Anuladores** Los compromisos pueden resolver el problema de la existencia, pero también crean uno nuevo: ¿cómo se evita gastar la misma nota dos veces?

En Bitcoin, esto es trivial, porque cuando gastas un UTXO, haces referencia directa a su identificación de transacción y al índice de salida, de modo que todo el mundo puede ver que el UTXO se ha gastado. Si intentas gastarlo de nuevo, los nodos rechazarán las transacciones porque el UTXO ha sido marcado como consumido.

Lo mismo no es posible en Zcash. Si para gastar una nota fuera necesario señalar su compromiso, se revelaría qué compromiso se está gastando y se vincularía esa nota a todas las transacciones futuras, lo que supondría una violación de la privacidad.

En Zcash, la solución para evitar gastar dos veces la misma nota son los *anuladores*. Los anuladores son valores derivados de una nota y solo pueden ser calculados por el propietario de la nota.

> **Example** Supongamos que el árbol de compromisos tiene un millón de notas y que una de ellas es suya, específicamente "compromiso `0x1a2b...`"
>
> Si para gastar la nota tuvieras que decir "Estoy gastando `0x1a2b...`" entonces:
>
> Todo el mundo sabría que `0x1a2b...` es tuya, y ya no sería solo uno de los millones de compromisos anónimos. Se etiqueta como perteneciente a quien haya realizado esta transacción y, aunque no se sepa qué hay en ese compromiso, sigue siendo problemático que se sepa que es suyo.

Los remitentes ahora pueden rastrearlo, ya que quien haya creado esa nota al enviarle el ZEC conoce el compromiso que ha creado. Por lo tanto, cuando gasta y lo señala, ellos pueden observar que se ha gastado el pago y saber cuándo ha movido sus fondos.

Con el tiempo, el gasto puede llegar a ser vinculable. Un observador podría correlacionar las transacciones basándose en los patrones de gasto, el momento y el destino, de modo que tus compromisos se agruparan como "probablemente la misma persona."

Los anuladores resuelven estos problemas. Si publicas el anulador `0x2c3d...`, que corresponde al compromiso `0x2c3d...`, es imposible calcular la correspondencia entre los compromisos y los anuladores sin conocer tu clave privada. El compromiso permanece anónimo en el árbol Merkle, tus gastos no se pueden vincular y el remitente no puede saber si su pago se ha gastado.

Aquí tienes un ejemplo de un anulador en Orchard:

```
nullifier = Hash(nk, rho, psi) = 0x2c3d4e5f6a7b...
    ↪ d2e3f48e9f0a1b2c3d
```

nk es la clave derivada del anulador, una clave secreta que solo usted posee. rho y psi son valores de la propia nota, como se ha visto anteriormente. Nadie más puede calcular este anulador porque nadie más tiene su nk. Hash, como en los ejemplos anteriores, es la función hash que se utiliza (lo veremos más adelante).

Cada vez que gastas una nota, también publicas su anulador. La red mantiene un conjunto de anuladores, es decir, una colección de todos los anuladores que se han publicado. Por lo tanto, si un anulador ya está en el conjunto, la transacción se rechaza, lo que evita el doble gasto.

**Example** Así es como crece el conjunto de anuladores con el tiempo:

- Block 1000000: `nullifier set = { }`
- Block 1000001: `nullifier set = { 0 ↪ x2c3d...3d }`
- Block 1000002: `nullifier set = { 0 ↪ x2c3d...3d, 0x8f7a...2b }`
- Block 1000003: `nullifier set = { 0 ↪ x2c3d...3d, 0x8f7a...2b, 0x1e4c...9 ↪ a }`

Cada gasto añade exactamente un anulador. El conjunto no puede reducirse, solo crecer.

A riesgo de ser repetitivos, veamos una vez más por qué la imposibilidad de vincular es una propiedad fundamental. El anulador no revela nada sobre a qué compromiso corresponde. Un observador ve aparecer un anulador y sabe que se ha gastado alguna nota, pero no puede saber cuál. El árbol de compromisos podría contener millones de notas y el anulador podría corresponder a cualquiera de ellas.

**Poniendo todo junto** Dado que los compromisos nunca se eliminan, ya que el árbol de compromisos es de solo añadir y crece indefinidamente, los compromisos permanecen en el árbol incluso después de que se haya gastado una nota.

Esto es precisamente lo que hace que el conjunto de anonimato de Zcash sea tan fuerte. Para gastar es necesario demostrar "Sé cuál es uno de los N millones de compromisos de este árbol" sin revelar cuál es. El compromiso de la nota gastada se mezcla con los demás, de modo que, aunque un observador vea aparecer un anulador, no podría determinar a cuál de los millones de compromisos corresponde.

Su conjunto de privacidad incluye todas las notas protegidas que se han creado en la red.

En resumen, cada transacción protegida implica:

1. Crear notas, lo que añade nuevos compromisos de notas al árbol de compromisos.
2. Gastar notas, lo que publica y añade un anulador al conjunto de anuladores.

Para construir una transacción, debes proporcionar un zk-SNARK que demuestre:

- Que conoce una nota con un compromiso en el árbol, a través de una ruta Merkle válida.
- Que conoce la clave secreta necesaria para calcular el anulador de esa nota.
- El anulador que está publicando corresponde a esa nota.
- El saldo de las cantidades de toda la transacción; las entradas son iguales a las salidas más la comisión.

La red verifica la prueba, comprueba si el anulador está en el conjunto y acepta la transacción. Es importante destacar que nunca se sabe qué compromiso se gastó, quién envió los fondos a quién ni cuánto se transfirió.

### 3.6 Claves y Direcciones

Bitcoin tiene un modelo de claves sencillo: una clave privada, una clave pública y una o más direcciones. El sistema protegido de Zcash es más complejo, ya

que las diferentes operaciones requieren diferentes niveles de acceso. Zcash aprovecha una jerarquía de claves para abordar esta complejidad.

**La Clave de Gasto** La clave de gasto (`sk`) es tu secreto maestro, es un número muy largo y aleatorio de 256 bits. Quien la tenga puede gastar tus fondos, ya que todo lo demás se deriva de la clave de gasto.

**La Clave de Visualización Completa** La clave de visualización completa (`fvk`), derivada de la clave de gasto, le permite ver todo lo relacionado con la actividad de su monedero: pagos entrantes, pagos salientes, importes y campos de notas, pero no puede gestionar los gastos.

La clave de visualización completa es útil en los casos en los que se desea conceder a alguien acceso de auditoría sin darle control. A través de la clave de visualización, un contable podría verificar su historial de transacciones, una empresa podría permitir que el departamento de cumplimiento normativo revisara sus libros o una autoridad fiscal podría confirmar los ingresos declarados; todo ello sin correr el riesgo de que el auditor se lleve los fondos.

**Claves de Visualización de Entradas y Salidas** La clave de visualización completa también se puede dividir en sus elementos constitutivos:

Clave de visualización entrante (`ivk`), que le permite detectar y descifrar las notas que le envían, pero no las que usted envía a otros. Clave de visualización saliente (`ovk`), que le permite descifrar los textos cifrados salientes, de modo que pueda ver lo que ha enviado y a quién.

Esta granularidad existe porque es posible que los usuarios solo deseen compartir información limitada. Por ejemplo, si desea proporcionar un servicio con su clave de visualización entrante para que el servicio pueda notificarle los pagos recibidos, sin revelar ninguna información sobre sus patrones de gasto.

Una billetera también puede optar por hacer que los detalles de las notas enviadas sean irrecuperables, incluso para los titulares de la clave de visualización completa. Para ello, utiliza una OVK aleatoria en el momento del envío y la borra inmediatamente de la memoria. A continuación, el `outCiphertext` se cifra con una clave que nadie posee, lo que hace imposible determinar la dirección del destinatario solo a partir de la FVK. El valor aún se puede inferir restando el cambio del total de la entrada, pero el destino se pierde.

**La Clave Derivada del Anulador** La clave derivada del anulador (`nk`), también derivada de la clave de gasto, se utiliza para calcular los anuladores al realizar un gasto. Esto es necesario para marcar las notas como gastadas, por lo que las claves de visualización por sí solas no pueden autorizar transacciones, ya que no tienen acceso a `nk`.

**Direcciones** En la parte inferior de la jerarquía se encuentran las direcciones: lo que le das a las personas para que puedan pagarte. En Orchard, las direcciones se derivan de la clave de visualización completa utilizando un diversificador, que es solo un pequeño fragmento de datos aleatorios.

El diversificador permite direcciones diversificadas, lo que significa que puedes generar miles de millones de direcciones no vinculables desde una sola billetera. Aunque cada dirección es completamente diferente, todas se canalizan hacia el mismo conjunto de claves. Además, puedes dar una dirección única a cada persona o servicio con el que interactúas.

> **Example** Supongamos que recibes pagos de un empleador, un cliente y una bolsa de cambio. Les das a cada uno una dirección diversificada diferente:
>
> - El empleador paga a: `u1employer8jp8rpf6...qjmxgmwxa`
> - El cliente paga a: `u1clientaph7jp8rpf` ↪ `...sz7nt28qj`
> - La bolsa paga a: `u1exchng2aaph7jp8...` ↪ `gmwxasz7n`
>
> Las tres direcciones le pertenecen y su monedero recibe los pagos entrantes de cada remitente, pero el empleador, el cliente y la plataforma de intercambio no pueden deducir que están pagando al mismo usuario comparando sus direcciones.

**La Jerarquía de Clave** Esta es la jerarquía:

```
spending key (sk)
    |
    +---> full viewing key (fvk)
    |         |
    |         +---> incoming viewing key (ivk)
    |         |
    |         +---> outgoing viewing key (ovk)
    |         |
    |         +---> addresses (via diversifiers)
    |
    +---> nullifier deriving key (nk)
```

A medida que se desciende en la jerarquía, cada nivel revela menos información. La clave de gasto puede

hacer todo, la clave de visualización completa lo ve todo, pero no puede gastar, y la clave de visualización de ingresos solo ve los fondos entrantes. Por último, las direcciones no revelan nada, solo son destinos.


Figure 4: Eli Ben-Sasson, cofundador de Zcash y ahora al frente de StarkWare.

## 4. El ciclo de vida de una transacción

Este capítulo cubrirá exactamente lo que ocurre cuando envías ZEC protegido, desde el momento en que presionas "enviar" hasta el momento en que el destinatario ve su saldo actualizado. Para ilustrar esto, seguiremos cada etapa de una sola transacción, examinando lo que calcula tu billetera, lo que ve la red y lo que queda registrado en la cadena de bloques.

### 4.1 La configuración

Alice quiere enviar 5 ZEC a Bob. Abre su billetera digital, introduce la dirección protegida de Bob, especifica la cantidad y confirma el envío. Lo que ocurre a continuación implica cada uno de los mecanismos que hemos visto hasta ahora: notas, compromisos, anuladores, claves, pruebas Merkle y zk-SNARK.

La billetera digital de Alice contiene dos notas sin gastar:

- **Nota A:** 3 ZEC
- **Nota B:** 4 ZEC

Ella gastará ambas (7 ZEC en total) para enviar a Bob 5 ZEC, pagará una tarifa de 0,001 ZEC y recibirá 1,999 ZEC de cambio.

### 4.2 Selección y recuperación de notas

Recuerde que la billetera digital de Alice no almacena realmente ZEC, sino la información necesaria para gastar las notas: los datos descifrados de las notas y las claves que las controlan. Cuando Alice sincronizó su billetera digital, esta escaneó el blockchain, intentó descifrar cada salida protegida utilizando su clave de visualización entrante y almacenó las que tuvieron éxito.

Aquí hay un ejemplo de la nota A:

```
{
  "addr": "u1alice...",
  // 3 ZEC in zatoshis
  "v": 300000000,          // 3 ZEC denominados
  ↪ en zatoshis
  "rho": "0x7a8b9c...",
  "psi": "0x1d2e3f...",
  "rcm": "0x4a5b6c...",
  // Position in commitment tree
  "position": 847291,      // Position in
  ↪ commitment tree
  // The commitment
  "cmx": "0x9f8e7d..."     // El compromiso
}
```

El campo de posición es crucial porque le indica a la billetera digital dónde se encuentra esta nota en el árbol de compromisos, información necesaria para construir la prueba Merkle.

### 4.3 Obtención de rutas Merkle

Para gastar una nota, Alice debe demostrar que su compromiso existe en el árbol, sin revelar de qué compromiso se trata. Esto requiere demostrar una ruta Merkle desde el compromiso hasta la raíz.

La billetera de Alice mantiene los testigos Merkle localmente mientras sincroniza la cadena de bloques, actualizándolos a medida que se agregan nuevos compromisos al árbol. Esto es fundamental: consultar un nodo completo para obtener una ruta Merkle en una posición específica revelaría qué nota se está gastando, lo que representaría una grave fuga de privacidad. Los nodos completos ni siquiera mantienen todo el

árbol de compromisos de notas, solo las fronteras recientes y el conjunto de anclajes válidos.

Para la nota A, en la posición 847 291 en un árbol con una profundidad de 32, la ruta consta de 32 hash similares:

```
merkle_path_A = [
  "0x1a2b3c...",  // Sibling at level 0
  "0x4d5e6f...",  // Sibling at level 1
  ...            // 30 more siblings
  "0x7g8h9i..."   // Sibling at level 31
]
```

Cualquiera que tenga acceso a esta ruta puede verificar que `cmx_A` está en el árbol mediante un hash de vuelta a la raíz. Sin embargo, dentro del zk-SNARK, Alice puede demostrarlo sin revelar `cmx_A` ni la propia ruta.

La billetera digital también registra el ancla, es decir, la raíz Merkle en el momento de la recuperación de la ruta. La transacción hará referencia a esta ancla y los nodos podrán utilizarla para verificar que se trata de una raíz reciente y válida.

### 4.4 Cálculo de anuladores

Alice tiene sus notas y sus rutas Merkle, ahora necesita marcarlas como gastadas.

Recordemos de la sección 3.5 que los nulificadores resuelven el problema fundamental de evitar el doble gasto sin revelar la nota que se está gastando. Con Bitcoin, tienes que señalar un UTXO directamente y cualquiera puede ver que ya fue consumido, pero con Zcash, señalar un compromiso destruiría la privacidad al vincularte a esa nota específica.

Alice calcula un anulador para cada nota que gasta, el anulador se deriva de los datos de la nota y de su clave secreta de derivación del anulador (`nk`):

```
nullifier_A = Hash(nk, rho_A, psi_A) = 0x2c3d4e5f
    ↪ ...
nullifier_B = Hash(nk, rho_B, psi_B) = 0x8f7a9b2c
    ↪ ...
```

Los valores `rho` y `psi` son únicos para cada nota, lo que significa que se establecieron cuando se creó la nota. El `nk` se deriva de la clave de gasto de Alice, pero solo ella lo posee.

La construcción tiene dos propiedades críticas:

1. **Es determinista:** Cada nota produce exactamente un anulador. Si Alice intentara gastar la nota A dos veces, tendría que publicar 0x2c3d4e5f... dos veces. La red mantiene un conjunto de anuladores de todos los anuladores publicados, por lo que el segundo intento sería rechazado porque ese anulador ya existe.
2. **Es inseparable:** Nadie más puede calcular el anulador de las notas de Alice porque nadie más tiene su nk y, lo que es más importante, nadie puede trabajar retrospectivamente a partir de un anulador para determinar su compromiso correspondiente. Por lo tanto, cuando 0x2c3d4e5f... aparece en la cadena de bloques, los observadores verán que se ha gastado alguna nota, pero no podrán saber de cuál de los millones de compromisos del árbol procede.

Los anuladores se incluirán en la transacción de Alice y se publicarán en la cadena, pero son el único rastro público de su gasto. Solo dos valores opacos de 32 bytes que no revelan nada sobre los billetes en sí, sus cantidades o quién los controlaba.

> **Note** El conjunto de anuladores solo crece. A diferencia del árbol de compromisos (que solo se puede añadir, pero que rastrea todas las notas creadas), el conjunto de anuladores rastrea las notas gastadas. El compromiso de una nota permanece en el árbol para siempre, incluso después de haber sido gastado. La presencia del anulador en el conjunto de anuladores es lo que lo marca como consumido.

### 4.5 Creación de notas de salida

Alice está gastando 7 ZEC (3 ZEC + 4 ZEC) y necesita crear dos notas nuevas: 5 ZEC para Bob y 1,999 ZEC para su cambio; hay una comisión por transacción de 0,001 ZEC.

Cada nota requiere una aleatoriedad novedosa, por lo que la billetera digital de Alice genera los componentes criptográficos que hacen que cada nota sea única y solo pueda ser gastada por su destinatario previsto.

**Generación de componentes de nota** Para la nota de 5 ZEC de Bob:

```
{
    "addr": "u1bob...",           // Bob's shielded
    ↪  address
    "v": 500000000,               // 5 ZEC in
    ↪ zatoshis
    "rho": "0x3e4f5a6b...",       // Derived
    ↪ deterministically
    "psi": "0x7c8d9e0f...",       // Random
    "rcm": "0x1a2b3c4d..."        // Random (
    ↪ commitment randomness)
}
```

Para el cambio de 1,999 ZEC de Alice:

```
{
    "addr": "u1alice...",          // Alice's own
    ↪ address
    "v": 199900000,                // 1.999 ZEC in
    ↪ zatoshis
    "rho": "0x5f6a7b8c...",
    "psi": "0x9d0e1f2a...",
    "rcm": "0x4e5f6a7b..."
}
```

El valor `rho` en Orchard se deriva de forma determinista de la transacción, lo que previene ciertos tipos de ataques criptográficos. Los valores `psi` y `rcm` son números aleatorios recién muestreados. Juntos, estos valores garantizan que, incluso si Alice envía a Bob 5 ZEC mil veces, el compromiso de la nota sería diferente cada vez.

**Cálculo de compromisos** Una vez que los componentes de la nota están listos, Alice calcula el compromiso para cada salida:

```
cmx_bob = Hash(addr_bob, 500000000, rho_bob,
    ↪ psi_bob, rcm_bob)
                  = 0x8a9b0c1d...
```

```
cmx_alice = Hash(addr_alice, 199900000, rho_alice,
    ↪ psi_alice, rcm_alice)
                  = 0x2d3e4f5a...
```

Estos compromisos son los que se publicarán en la cadena y se añadirán al árbol de compromiso. No revelan nada sobre las notas en sí, son hash opacos de 32 bytes, pero cualquiera que conozca los valores subyacentes (el destinatario, concretamente) puede verificar que un compromiso corresponde a una nota específica.

**Cifrado de las notas** Los compromisos se incluyen en cadena, pero Bob necesita los datos reales de la nota para poder gastar posteriormente sus 5 ZEC. Necesita conocer el valor, `rho`, `psi` y `rcm`, ya que sin ellos, el compromiso es inútil, ya que no puede construir un anulador válido ni demostrar la propiedad.

Alice cifra cada nota para que solo el destinatario previsto pueda leerla:

**Para Bob:** Alice utiliza la dirección de Bob (que contiene su material de clave pública) para cifrar la nota. El resultado es el texto cifrado `encCiphertext`: un bloque de datos cifrados que solo se puede descifrar utilizando la clave de visualización entrante de Bob. Cuando la billetera digital de Bob escanea la cadena de bloques y descifra con éxito este texto cifrado, se entera de que ha recibido 5 ZEC y almacena todos los datos necesarios para gastarlos.

**Para los registros de Alice:** Hay un segundo texto cifrado llamado `outCiphertext`: este está cifrado con la clave de visualización saliente de Alice, lo que permite a su billetera digital recordar lo que envió. Sin esto, Alice no tendría un registro de dónde fueron sus fondos. Está cifrado, en lugar de almacenado en texto plano, para que los operadores de nodos y los observadores no puedan leerlo.

```
{
    "cmx": "0x8a9b0c1d...",
    "ephemeralKey": "0x6b7c8d9e...",
    "encCiphertext": "0x9f8e7d6c5b4a...[512 bytes
    ↪ ]...",
    "outCiphertext": "0x3c4d5e6f7a8b...[80 bytes
    ↪ ]..."
}
```

El `ephemeralKey` es una clave pública de un solo uso generada para este cifrado específico, y Bob puede utilizarla junto con su clave privada para descifrar el `encCiphertext`. Esto es habitual en el cifrado de clave pública, pero la novedad es que se produce dentro de un sistema que nunca ha vinculado la dirección de Bob a una identidad y en el que el texto cifrado no revela nada a los observadores externos.

> **Note** El cifrado no forma parte de lo que prueba zk-SNARK. El cifrado es una capa independiente que garantiza que solo los destinatarios puedan acceder a sus fondos, mientras que la prueba verifica que las notas estén correctamente formadas y que los importes de las transacciones cuadren. Si Alice cifrara incorrectamente (o utilizara maliciosamente la clave equivocada), la transacción seguiría siendo válida en cadena, pero Bob nunca podría encontrar ni gastar su nota. En la práctica, las billeteras digitales gestionan esto correctamente, y la incapacidad del destinatario para descifrarlo sería un error de la billetera digital, no una violación del protocolo.

En este punto, Alice tiene todo lo necesario para las salidas: dos compromisos para publicar y cargas útiles cifradas para que cada destinatario pueda reclamar su nota. Ahora viene la parte difícil: demostrar que es válida sin revelar nada.

### 4.6 La prueba

Alice ha reunido todas las piezas: Las dos notas para gastar, sus rutas Merkle, los anuladores que las marcarán como consumidas y dos notas de salida

nuevas con sus compromisos y cargas útiles cifradas. Ahora bien, ¿cómo convencer a la red de que todo es válido sin revelar los detalles?

Aquí es donde entra en juego zk-SNARK.

**Lo que demuestra la prueba** La prueba es un objeto criptográfico que demuestra que todo lo siguiente es cierto:

1. **Las notas de entrada existen.** Alice conoce dos de los compromisos que se encuentran en el árbol de compromiso. Ella lo demuestra al esbozar las rutas Merkle válidas desde esos compromisos hasta el ancla (la raíz del árbol). La prueba no revela a qué compromisos se refiere Alice, solo que están ahí en algún lugar entre los millones.

2. **Alice controla las entradas.** Alice conoce las claves de gasto de ambas notas, concretamente, conoce los valores secretos necesarios para derivar los anuladores y autorizar el gasto. Sin esto, cualquiera podría intentar gastar las notas de otra persona.

3. **Los anuladores son correctos.** Los anuladores que publica corresponden realmente a las notas que está gastando. Alice no puede publicar anuladores arbitrarios, deben derivarse de notas reales que controla utilizando la fórmula adecuada.

4. **Los importes de la transacción cuadran.** La suma de los valores de entrada ($3 + 4 = 7$ ZEC) es igual a la suma de los valores de salida ($5 + 1{,}999 = 6{,}999$ ZEC) más la comisión (0,001 ZEC). No se crea ni se destruye ningún ZEC. Esta es la ley fundamental de conservación del sistema.

5. **Los compromisos de salida están bien formados.** Los compromisos que publica para la nota de Bob y su nota de cambio se calculan correctamente a partir de datos válidos de la nota. No puede publicar compromisos sin sentido, deben seguir la estructura adecuada.

La red no sabe qué notas se gastaron, quién es el destinatario ni la cantidad que se transfirió de una parte a otra. Solo sabe que alguien realizó una transacción válida: entradas reales, salidas reales, cálculos matemáticos correctos y autorización adecuada. Eso es suficiente para actualizar el estado global, es decir, añadir compromisos y registrar anulaciones, sin saber nada sobre la transacción en sí.

**Qué es realmente la prueba** Después de toda esta complejidad, la prueba en sí misma es casi anticlimática: aproximadamente uno o dos kilobytes de datos, ¡eso es todo! Es solo un pequeño bloque de bytes que codifica un argumento matemático.

La verificación es rápida, solo unos milisegundos en un hardware modesto. Un nodo recibe la prueba, ejecuta el algoritmo de verificación y devuelve una respuesta binaria: válida o no válida. Sin juicios de valor, sin heurística, sin conjeturas probabilísticas; las matemáticas se verifican o no se verifican.

Esta asimetría es la magia de zk-SNARKs. Crear la prueba es computacionalmente costoso, la billetera digital de Alice hace un trabajo real, procesando operaciones de curvas elípticas y matemáticas polinómicas. Sin embargo, verificar la prueba es barato. La asimetría hace que el sistema sea práctico: cada nodo de la red puede verificar cada transacción protegida sin tener que volver a realizar el complicado cálculo.

**El circuito** ¿Cómo genera Alice esta prueba? Procesando los datos de su transacción a través de algo llamado circuito, una especificación formal de las condiciones exactas que deben cumplirse para que un gasto en Orchard sea válido.

Piensa en el circuito como una enorme lista de verificación codificada en restricciones matemáticas. El paso para demostrar que "la ruta Merkle debe ser válida" se convierte en una serie de cálculos hash que deben producir el resultado correcto, el paso "el anulador debe derivarse correctamente" se convierte en restricciones sobre cómo se relacionan entre sí ciertos valores y, finalmente, "las cantidades deben cuadrar" se convierte en una ecuación que debe cumplirse.

La cartera digital de Alice toma sus entradas privadas (notas, claves, rutas, aleatoriedad) y las procesa a través de este circuito para encontrar valores que satisfagan todas las restricciones. A continuación, el mecanismo zk-SNARK comprime toda esta asignación satisfactoria en una pequeña prueba que cualquiera puede comprobar.

> **Note** El circuito se fija a nivel de protocolo, y todas las transacciones de Orchard utilizan el mismo circuito, tal y como se define en la especificación de Zcash. Alice no puede modificar las reglas, solo puede demostrar que las ha seguido. Esto es lo que hace que el sistema sea fiable: los nodos no necesitan confiar en Alice, solo tienen que verificar que su prueba supera el circuito universal acordado.

La billetera digital de Alice ha generado ahora una prueba: un objeto de ~1,5 KB que afirma que existe una transacción válida, sin decir cuál es. Ahora es el momento de empaquetarlo todo y enviarlo a la red.

### 4.7 Ensamblaje de la transacción

Alice tiene sus anuladores, sus notas de salida, sus cargas útiles cifradas y su prueba; ahora necesita empaquetar todo en una transacción que la red pueda procesar.

**La estructura de la acción** Orchard utiliza una estructura denominada "acción". Cada acción agrupa exactamente un gasto y una salida, lo cual es una elección de diseño deliberada. Los protocolos anteriores de Zcash (Sprout y Sapling) separaban los gastos y las salidas, pero esto filtraba información sobre la estructura de la transacción. Si veías una transacción con tres gastos y una salida, obtenías cierta información. Orchard elimina este problema al forzar un emparejamiento 1:1.

Alice gasta dos notas y crea dos salidas, por lo que su transacción contiene dos acciones:

- **Acción 0:** Gasta la nota A (3 ZEC), crea la nota de Bob (5 ZEC)
- **Acción 1:** Gasta la nota B (4 ZEC), crea la nota de cambio de Alice (1,999 ZEC)

El emparejamiento dentro de cada acción es arbitrario. La acción 0 no significa que la Nota A "se convirtiera" en los 5 ZEC de Bob. Los valores no coinciden, y eso está bien. Lo que importa es la restricción global: el total de entradas es igual al total de salidas más la comisión. La estructura de la acción solo garantiza que los observadores no puedan inferir la forma de la transacción.

> **Note** ¿Qué pasaría si Alice quisiera gastar dos notas, pero solo crear una salida? Para ello, seguiría necesitando dos acciones, por lo que tendría que crear una salida ficticia en la segunda acción. Una salida ficticia es una nota de valor cero que solo existe para equilibrar la estructura. Lo mismo se aplica a la inversa: si tuviera una entrada pero necesitara dos salidas, incluiría un gasto ficticio. Los observadores no pueden distinguir las acciones reales de las ficticias.

**Lo que se registra en cadena** Esto es lo que contiene realmente la transacción de Alice:

```
{
    "anchor": "0x7f8e9d0c...",
    "actions": [
        {
            "cv": "0x9a8b7c6d...",
            "nullifier": "0x2c3d4e5f...",
            "rk": "0x5e6f7a8b...",
            "cmx": "0x8a9b0c1d...",
            "ephemeralKey": "0x6b7c8d9e...",
            "encCiphertext": "0x9f8e7d6c...[580
    ↪ bytes]",
            "outCiphertext": "0x3c4d5e6f...[80
    ↪ bytes]"
        },
        {
            "cv": "0x1b2c3d4e...",
            "nullifier": "0x8f7a9b2c...",
            "rk": "0x4d5e6f7a...",
            "cmx": "0x2d3e4f5a...",
            "ephemeralKey": "0x8c9d0e1f...",
            "encCiphertext": "0x7e8f9a0b...[580
    ↪ bytes]",
            "outCiphertext": "0x5a6b7c8d...[80
    ↪ bytes]"
        }
    ],

    "proof": "0x1a2b3c4d...[~1.5 KB]",
    "bindingSig": "0x4e5f6a7b...[64 bytes]"
}
```

Desglosemos esto:

`anchor` **(ancla):** la raíz de Merkle a la que hace referencia la prueba de Alice. Esto compromete su transacción a un estado específico del árbol de compromisos. Los nodos verificarán que esta sea una raíz válida que existió en algún momento del historial del árbol. Aunque los anchors antiguos son técnicamente válidos, las billeteras suelen usar anchors recientes para maximizar el conjunto de anonimato.

- `cv (value commitment)` **(compromiso de valor):** Un compromiso criptográfico del valor que se gasta o se crea en cada acción. Estos no revelan los montos reales. En su lugar, están construidos de tal forma que la suma de todos los valores cv de la transacción codifica el flujo neto. Si la transacción está balanceada (entradas = salidas + comisión), la matemática cuadra. Si no, la verificación falla.

- `nullifier` **(anulador):** Los anuladores para la Nota A y la Nota B. Estos se añaden al conjunto de anuladores, marcando esas notas como gastadas para siempre.

- `rk (randomized verification key)` **(clave de verificación aleatorizada):** Se utiliza para verificar la firma de autorización de gasto. Esto demuestra que Alice autorizó esta transacción específica sin revelar su clave de gasto real.

- **cmx:** Los compromisos para la nota de Bob y la nota de cambio de Alice. Estos se añaden al árbol de compromisos.

- **ephemeralKey + encCiphertext + outCiphertext:** Los datos de la nota encriptados, como se describe en la sección 4.5. Estos no afectan al consenso, pero sin ellos, los destinatarios no podrían reclamar sus fondos.

- `proof` **(prueba):** El zk-SNARK que demuestra que todo es válido. Una sola prueba cubre la transacción completa (ambas acciones).

- `bindingSig` **(firma vinculante):** Una firma que une todas las piezas. Demuestra que los valores cv de todas las acciones suman correctamente (garantizando la conservación del valor) y que la transacción no ha sido manipulada. Este es el chequeo final de que los montos realmente cuadran.

**La Comisión** Notarás que la comisión no aparece explícitamente en ningún lugar, y eso se debe a que es implícita. El total de entradas de Alice es 7 ZEC y el total de salidas es 6.999 ZEC. La diferencia, 0.001 ZEC, es la comisión de la transacción, que es reclamada por los mineros.

Los compromisos de valor codifican el flujo neto, por lo que cuando un minero verifica la firma vinculante, en realidad está confirmando que las entradas menos las salidas son iguales a la comisión declarada. Si Alice intentara afirmar que sus salidas suman 7 ZEC, dejando sin comisión, entonces la firma vinculante fallaría. Si ella intentara crear ZEC adicionales de la nada y afirmara 8 ZEC de salidas a partir de 7 ZEC de entradas, la prueba en sí sería inválida.

La comisión es pública. Los observadores pueden ver cuánto se pagó para procesar la transacción, pero ese es el único valor visible. Los montos de las entradas, los montos de las salidas y la transferencia de valor entre las partes permanecen ocultos.

Es importante destacar que ZIP 317 estandariza el cálculo de las comisiones, de modo que las billeteras compatibles no permiten montos discrecionales. Esto importa para la privacidad: si las billeteras permitieran comisiones arbitrarias, la elección de la comisión filtraría información que podría ayudar a identificar transacciones o distinguir entre implementaciones de billeteras.

### 4.8 Difusión y Mempool

La billetera digital de Alice ya ha ensamblado la transacción completa; ahora esta necesita llegar a la red.

**Envío a la Red** El proceso de envío ocurre de la siguiente manera: la billetera digital de Alice se conecta a uno o más nodos de Zcash y difunde la transacción. El mensaje se propaga a través de la red entre pares, pasando de nodo a nodo hasta llegar a los mineros y al resto de la red. El proceso de envío funciona exactamente igual que en Bitcoin: la transacción es simplemente datos que los nodos difunden entre sus pares en la red.

Desde la perspectiva de Alice, esto toma uno o dos segundos. En su billetera digital aparece "transacción difundida" y luego solo tiene que esperar la confirmación.

**Validación Inicial** Cuando un nodo recibe la transacción de Alice, no la acepta ciegamente. Antes de reenviarla a otros nodos o agregarla al mempool, el nodo ejecuta una serie de verificaciones:

1. **Verificación de Prueba:** El nodo ejecuta el verificador zk-SNARK sobre la prueba de Alice. Esto toma unos pocos milisegundos. Si la prueba es inválida, la transacción se rechaza inmediatamente. No se necesitan más verificaciones.

2. **Verificación del ancla:** El nodo verifica que el ancla `anchor` utilizada por Alice (la raíz de Merkle a la que hace referencia su prueba) sea una raíz válida del historial del árbol de compromisos. El protocolo de consenso no prohíbe los `anchor` antiguos: se acepta cualquier `anchor` que haya sido alguna vez una raíz válida del árbol. Sin embargo, es muy recomendable utilizar un `anchor` ↪ reciente, ya que maximiza el conjunto de anonimato: cuantas más notas haya en el árbol al momento del `anchor`, mayor será la multitud en la que se oculta la nota de Alice. Algunas billeteras, como YWallet, permiten seleccionar `anchor`s más antiguos para poder gastar notas antiguas sin necesidad de que la billetera haya escaneado todos los bloques posteriores.

3. **Verificación de anuladores:** El nodo revisa ambos anuladores en su conjunto local de anuladores. Si alguno de ellos, por ejemplo `0` ↪ `x2c3d4e5f...` o `0x8f7a9b2c...`, ya existe en el conjunto, significa que Alice está intentando gastar dos veces. En ese caso, la transacción se rechaza.

4. **Validez estructural:** El nodo confirma que la transacción esté bien formada: longitudes de campo correctas, codificaciones válidas, que la firma vinculante sea verificable, entre otros aspectos. Las transacciones mal formadas se descartan.

Si todas las verificaciones se superan, el nodo considera que la transacción es válida. Luego la agrega a su mempool, que es un área de espera para transacciones no confirmadas, y la retransmite a otros nodos.

**Esperando en el Mempool** El mempool es como un purgatorio para las transacciones. La transacción de Alice permanece allí junto con cientos o miles de otras, todas esperando que un minero las seleccione y las incluya en un bloque.

Los mineros seleccionan transacciones del mempool basándose en las comisiones. Generalmente, las transacciones con comisiones más altas se eligen primero. Alice pagó 0.001 ZEC, lo cual es típico para Zcash, y en condiciones normales de la red esto suele ser suficiente para que se incluya en el siguiente bloque o en los dos siguientes.

Durante este período de espera, la transacción de Alice está sin confirmar. La red ya la ha validado, pero todavía no se ha escrito en la blockchain. La billetera de Bob podría detectar la transacción pendiente, algunas billeteras muestran transacciones entrantes no confirmadas, pero él no puede gastar esos fondos hasta que la transacción sea minada.

> **Note** El mempool no es global ni está sincronizado; cada nodo mantiene su propio mempool. Debido a los retrasos en la propagación de la red, diferentes nodos pueden tener conjuntos ligeramente distintos de transacciones pendientes en un momento dado. Esto no importa para el consenso; lo que realmente importa es qué transacciones se incluyen en los bloques.

La transacción ha sido difundida y los nodos la han validado. Ahora, Alice espera que un minero haga el trabajo final.

### 4.9 Inclusión en el Bloque y Finalidad

Un minero selecciona la transacción de Alice de su mempool, la agrupa con otras transacciones y comienza el trabajo de minar un nuevo bloque.

**Minado del Bloque** Zcash utiliza Prueba de Trabajo, al igual que Bitcoin. El minero construye un encabezado de bloque que contiene el hash del bloque anterior, una marca de tiempo, una raíz de Merkle de las transacciones incluidas y un nonce. Luego, prueban repetidamente diferentes nonces hasta encontrar uno que produzca un hash por debajo de la dificultad objetivo.

Este proceso es idéntico al que cubrimos en la introducción a Bitcoin (sección 3.1), con una excepción: Zcash utiliza el algoritmo Equihash en lugar de SHA256. Las propiedades de seguridad son las mismas: encontrar un bloque válido requiere un trabajo computacional significativo, mientras que verificar ese trabajo es trivial.

Cuando un minero encuentra un nonce válido, difunde el bloque y otros nodos lo verifican: prueba de trabajo válida, transacciones válidas, estructura correcta. Si todo está en orden, los nodos añaden el bloque a su cadena y la transacción de Alice se convierte en parte del registro permanente.

**Actualizaciones del Estado** Una vez que el bloque es aceptado, el estado de la red cambia:

- **El árbol de compromisos crece:** El compromiso de la nota de Bob `0x8a9b0c1d...` y el de la nota de cambio de Alice `0x2d3e4f5a...` se añaden al árbol. Ahora el árbol contiene dos "hojas" más que antes y se calcula una nueva raíz de Merkle. Esta raíz se convierte en un ancla válido para futuras transacciones.
- **El conjunto de anuladore se expande:** Los dos anuladores de Alice (`0x2c3d4e5f...` y `0x8f7a9b2c...`) se agregan al conjunto. Esas notas quedan ahora marcadas permanentemente como gastadas. Cualquier transacción futura que intente usar alguno de estos dos anuladores será rechazada.
- **Se emite la recompensa del bloque:** El minero recibe ZEC recién creados (el subsidio del bloque) más la suma de todas las comisiones de las transacciones incluidas en el bloque, incluyendo los 0.001 ZEC de Alice.

Estas actualizaciones del estado son decisivas. Cada nodo que procesa el bloque llega exactamente al mismo estado nuevo. El árbol de compromisos tiene la misma raíz nueva en todas partes. El conjunto de anuladores contiene las mismas entradas en todas partes. Esto es lo que hace que la red sea consistente sin necesidad de una coordinación centralizada.

**Confirmaciones** La transacción de Alice ahora está confirmada, pero confirmación no significa final-

idad.

Al igual que Bitcoin, Zcash utiliza un sistema de Prueba de Traabajo puro, el cual no posee finalidad criptográfica. La cadena con la mayor cantidad de trabajo acumulado es la que gana, pero nada impide que un atacante con recursos suficientes construya una cadena más larga que reescriba la historia. Las transacciones en bloques huérfanos regresan al mempool o se invalidan si entran en conflicto con la cadena del atacante.

La creencia convencional de que tras seis confirmaciones el riesgo de reversión es "insignificante" es engañosa. Presenta la seguridad como una propiedad estadística cuando, en realidad, es una propiedad adversarial. Esto aplica a todas las cadenas de PoW puro, incluido Bitcoin. Contra un atacante que posee la mayoría del poder de cómputo, ningún número de confirmaciones ofrece certeza criptográfica; solo ofrece suposiciones económicas sobre los incentivos del atacante y los costos del hashpower.

> **Note** El tiempo de bloque de 75 segundos de Zcash significa que las confirmaciones se acumulan más rápido: seis confirmaciones toman unos siete minutos y medio, frente a la hora que toma en Bitcoin. Cada bloque representa menos trabajo, pero las confirmaciones se suman rápidamente.

La transacción ha sido minada y el estado se ha actualizado. Las notas antiguas de Alice han desaparecido para siempre, reemplazadas por dos notas nuevas en el árbol de compromisos. Una le pertenece a Bob, y ahora él necesita encontrarla.

### 4.10 Detección del Destinatario

La transacción de Alice ya está en la cadena. Los 5 ZEC de Bob existen como un compromiso en el árbol, pero Bob aún no lo sabe. Su billetera necesita encontrar la nota correspondiente.

**Escaneo de la Blockchain** La billetera de Bob se sincroniza periódicamente con la red, descargando nuevos bloques y escaneando en busca de pagos entrantes. El desafío es que Bob no puede simplemente buscar su dirección. Las salidas protegidas no contienen direcciones en texto plano; cada salida parece datos encriptados aleatorios.

La billetera de Bob intenta desencriptar cada salida protegida con la que se encuentra; por lo tanto, para cada encCiphertext de cada acción de cada bloque, la billetera intenta la desencriptación utilizando la clave de visualización entrante de Bob. La mayoría de estos intentos fallan y producen datos inservibles, pero eso es lo esperado, ya que esas salidas pertenecen a otra persona.

Finalmente, cuando la billetera de Bob llega a la transacción de Alice e intenta desencriptar el ciphertext en la Acción 0, la desencriptación tiene éxito y surgen los datos válidos de la nota.

**Recuperación de la Nota** Cuando la desencriptación funciona, la billetera de Bob recupera el texto plano completo de la nota:

```
{
    "addr": "u1bob...",
    "v": 500000000,
    "rho": "0x3e4f5a6b...",
    "psi": "0x7c8d9e0f...",
    "rcm": "0x1a2b3c4d..."
}
```

Bob ahora tiene todo lo que necesita:

- **El valor:** 5 ZEC (500,000,000 zatoshis). Su billetera actualiza su saldo en consecuencia.
- **Los componentes de la nota:** Los valores `rho`, `psi` y `rcm` que Alice generó. Estos son esenciales. Sin ellos, Bob no podría calcular el compromiso para verificar que coincide con lo que está en la cadena, ni derivar el anulador para gastar la nota más tarde.
- **La posición:** La billetera de Bob también registra en qué parte del árbol se encuentra este compromiso. Cuando se procesó el bloque, el compromiso se añadió en un índice de hoja específico. Bob necesita esta posición para construir una ruta de Merkle cuando eventualmente decida gastar los fondos.

**Verificando la Nota** La billetera de Bob no confía ciegamente en los datos desencriptados. Vuelve a calcular el compromiso a partir de los valores recuperados:

```
cmx_check = Hash(addr_bob, 500000000, rho, psi, rcm
    ↪ )
```

Si el `cmx_check` coincide con el `cmx` publicado en cadena en la transacción de Alice, la nota es válida. Si no coinciden, algo es incorrecto (ya sea corrupción o remitentes malintencionados) y la billetera descarta la nota.

Durante las operaciones normales, esta verificación siempre es exitosa. La billetera de Alice construyó la nota correctamente y la desencriptación recuperó exactamente lo que ella encriptó.

**Una Nota Gastable** Bob ahora posee una nota de 5 ZEC gastable. Su billetera almacena los datos de la nota localmente y la mantiene lista para cuando él desee utilizarla. En ese momento, él seguirá el mismo proceso que Alice utilizó para enviársela a él:

1. Seleccionar la nota
2. Obtener su ruta de Merkle a partir de los testigos mantenidos localmente
3. Calcular su anulador
4. Crear notas de salida para sus destinatarios
5. Generar una prueba
6. Difundir la transacción

El ciclo se repite: el gasto de Bob revelará un anulador, marcando su nota como consumida, se añadirán nuevos compromisos al árbol, y luego nuevos destinatarios escanearán, descifrarán y descubrirán sus fondos.

> **Note** El escaneo de notas es el punto crítico de rendimiento para las billeteras protegidas, ya que una billetera que ha estado sin conexión durante meses necesita probar millones de salidas encriptadas para ponerse al día. Por esta razón, los clientes ligeros y los protocolos de sincronización optimizados son importantes. El Proyecto Tachyon, mencionado en la sección 2, busca mejorar drásticamente el proceso de actualización mediante sincronización inconsciente, permitiendo que las billeteras consulten servidores por datos relevantes sin revelar qué información se está buscando.

Alice envió 5 ZEC a Bob. La red verificó la transacción sin conocer quién envió qué a quién, pero Bob aún pudo detectar su pago sin que nadie más supiera que lo recibió. La transacción está completa.

## 5. La Filosofía de la Privacidad

### 5.1 La Privacidad como Precondición para el Progreso

Privacidad no significa secreto, ya que el secreto busca ocultar algo vergonzoso. La privacidad es el derecho a elegir qué revelas y a quién. La privacidad es la autonomía sobre tu propia información; es el fundamento de la libertad misma.

Esta distinción es importante porque los críticos de la privacidad suelen confundir ambas cosas. El refrán de los sistemas autoritarios proclama que "si no tienes nada que ocultar, no tienes nada que temer", y asume que la privacidad solo es valiosa para aquellos con algo que encubrir. Sin embargo, la privacidad es valiosa

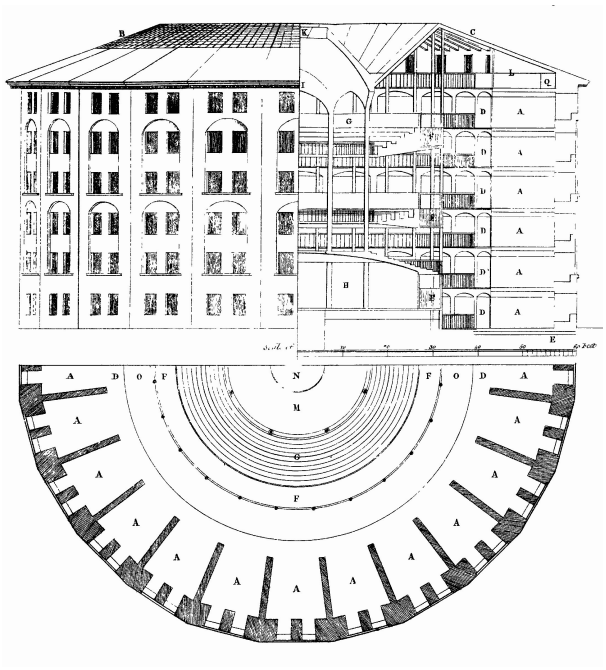


Figure 5: El Panóptico de Jeremy Bentham, 1791. Una prisión diseñada para que los reclusos nunca sepan si están siendo observados. Aprenden a vigilarse a sí mismos.

para todos, precisamente porque crea las condiciones para todo lo demás que valoramos: el pensamiento libre, la libertad de expresión, los mercados libres y el progreso.

**Las Condiciones para el Progreso** Karl Popper pensaba que el progreso dependía de la crítica. Se deben proponer ideas audaces, probarlas y corregirlas, para que los errores puedan ser identificados y descartados. Este proceso requiere estar libre de castigos por proponer ideas atrevidas antes de que sean probadas y que corren el riesgo de ser erróneas. El panóptico asegura que la disidencia sea silenciada antes de que pueda ser expresada. La innovación comienza a requerir permisos y la crítica es castigada: el mecanismo del progreso se rompe.

David Deutsch extendió las ideas de Popper, postulando que los seres humanos son únicos en virtud de ser explicadores universales. Nuestra capacidad para crear conocimiento, para entender el cosmos e incluso transformarlo, es lo que nos hace especiales. Sin embargo, la creación de conocimiento requiere experimentación, y la experimentación requiere la libertad de fallar en privado antes de tener éxito en público. La vigilancia inhibe la libertad de experimentar, porque cuando cada acción es observada y

registrada, se asfixia el pensamiento creativo.

Estas no son preocupaciones abstractas. Son la realidad de cualquiera que se haya autocensurado sabiendo que sus palabras estaban siendo vigiladas. De cualquiera que decidió no donar a una causa controversial o novedosa sabiendo que la transacción sería visible o rastreable. De cualquiera que evitó investigar un tema sensible sabiendo que la consulta quedaría registrada. La vigilancia cambia el comportamiento; esa es una de sus funciones principales. A veces, cambiar comportamientos equivale a restringir ideas y, por lo tanto, a restringir el progreso.

**El Dinero como el Monopolio Final** A lo largo de la historia, la libertad ha dependido de las herramientas que teníamos para protegerla.

La imprenta fomentó la libertad de expresión. Antes de Gutenberg, las ideas solían estar encadenadas a escribas y sacerdotes, encerradas tras la autoridad institucional. La imprenta rompió el monopolio de la información.

El internet rompió el monopolio de la geografía. Ahora las ideas pueden compartirse a través de las fronteras en un instante. La coordinación se hizo posible sin proximidad física. La censura se volvió más difícil cuando la información podía "enrutarse" alrededor de los obstáculos.

La pólvora destrozó el monopolio de los caballeros y los reyes sobre la violencia. Un campesino con un mosquete podía desafiar a un señor con armadura. El poder se distribuyó más.

Cada vez, una nueva herramienta aplastó un viejo monopolio. Ahora, queda un último monopolio: el dinero.

El dinero es la tecnología de coordinación más poderosa que los humanos hayan construido jamás. Es la forma en que señalamos el valor, asignamos recursos y cooperamos a gran escala; pero sigue estando significativamente restringido. El dinero es la tecnología más vigilada y controlada. Cada transacción puede ser monitoreada. Los gobiernos pueden congelar cuentas con solo pulsar una tecla. Los bancos pueden cancelarte de la noche a mañana. Cada vez más, los controles de capital pueden impedirte retirar tu propio efectivo.

Podría decirse que tu dinero no es tuyo si alguien más puede ver cada transacción que realizas, decidir si la aprueba o no, y más tarde incluso decidir revertir esa decisión e impedir el acceso.

**La Privacidad en los Mercados** Los mercados libres requieren privacidad; esta conclusión se desprende de entender cómo funcionan los mercados:

Los mercados agregan información a través de los precios, lo que significa que a medida que los participantes toman decisiones basadas en su conocimiento privado, los precios emergen de la suma de esas decisiones. El mecanismo funciona solo si los participantes pueden tomar decisiones basadas en su información privada sin revelarla prematuramente. Un trader que debe difundir cada posición antes de tomarla será adelantado. Una empresa que deba publicar cada relación con sus proveedores será perjudicada. Un donante que deba anunciar cada contribución será presionado.

Cualquier filtración, incluso de pequeñas piezas de información, cambia el mercado porque introduce sesgos y distorsiona las decisiones. Cuanta más vigilancia hay en un sistema, más distorsión enfrenta. Los mercados perfectos requieren participantes que puedan actuar libremente sobre información privada, lo cual se vuelve imposible bajo condiciones de vigilancia absoluta.

Tu patrimonio neto no debería ser una API pública. Tu historial de transacciones no debería ser una base de datos consultable. Tu vida financiera no debería estar sujeta a la aprobación de observadores. Estos no son casos aislados ni preocupaciones paranoicas; son los requisitos básicos para que los mercados funcionen y para que los individuos sean libres.

### 5.2 La Trampa de la Transparencia

Se suponía que las criptomonedas nos liberarían de la vigilancia financiera, pero hicieron lo contrario.

Los cypherpunks que construyeron este movimiento entendían lo que estaba en juego. Entendían que la privacidad en la era digital no sería otorgada por gobiernos o corporaciones, sino que tendría que ser construida, desplegada y defendida con herramientas criptográficas. Bitcoin surgió de esta tradición y logró ser la primera grieta en la presa, la prueba de que el dinero podía existir fuera del control gubernamental.

Sin embargo, Bitcoin tiene un fallo importante: es transparente por defecto. Cada transacción, cada dirección y cada saldo es visible para cualquiera que tenga interés en buscarlo. La blockchain es un libro contable público y permanente de toda la actividad económica que ha pasado por ella. Satoshi reconoció esta limitación en su whitepaper original, sugiriendo que los usuarios podrían preservar algo de privacidad generando nuevas direcciones para cada transacción.

Eso era una mitigación débil entonces, y se ha convertido en una absurda ahora.

El seudonimato significa que tu identidad no está directamente vinculada a tu dirección. No obstante, tu identidad puede filtrarse a través de la observación de tu comportamiento: las horas en las que realizas transacciones, los montos que mueves, las direcciones con las que interactúas; en resumen, los patrones que repites. Con cada punto de datos, el conjunto de identidades posibles para una dirección se reduce hasta que, finalmente, con suficientes restricciones, el conjunto colapsa en una sola.

En la era de la IA, el seudonimato es privacidad con tiempo prestado; es solo una ilusión esperando a ser disuelta por el poder de cómputo.

**El Comercio Requiere Opacidad** El problema de la transparencia no se limita a los individuos, ya que el comercio también se desmorona sin privacidad.

Considera lo que sucede cuando realizas un único pago a una empresa en una cadena transparente. Ahora puedes ver su dirección y, a partir de ella, deducir potencialmente sus ingresos totales, las direcciones de sus clientes, sus relaciones con proveedores, su nómina e incluso su flujo de caja y su liquidez.

Hay una razón por la que los departamentos de Recursos Humanos tratan las estructuras de compensación como secretos celosamente guardados, por la que las empresas no publican sus contratos con proveedores y por la que los estados financieros se emiten trimestralmente, en formatos controlados, en lugar de transmitirse en tiempo real al público. Los mercados competitivos requieren asimetría de información. Las empresas deben poder actuar basándose en conocimiento privado sin transmitirlo a sus competidores.

La misma lógica se aplica a los individuos. Si tus patrones de gasto revelan tus condiciones de salud, tus afiliaciones políticas, tus prácticas religiosas y tus relaciones personales, entonces cada transacción se convierte en un punto de datos para generar una imagen de quién eres, qué valoras y cómo se te puede influir o coaccionar.

La web necesitaba HTTPS antes de que el comercio pudiera funcionar en línea. Transmitir números de tarjeta de crédito en texto plano era obviamente inaceptable por razones de seguridad. La capa de pagos de Internet necesita la misma evolución: así como las transacciones en texto plano eran un prototipo, la producción requiere ser encriptado.

### 5.3 La Privacidad Debe Ser Absoluta

Las medidas a medias no funcionan porque la privacidad es binaria: o la tienes o no la tienes.

Esto puede sonar extremo, pero se deduce de cómo funciona la información. Un secreto es solo un secreto hasta que se filtra; una vez filtrado, no puede dejar de estar filtrado. En un mundo de almacenamiento permanente, reconocimiento de patrones y análisis impulsado por IA, cualquier filtración parcial crece hasta convertirse en una filtración completa. La pregunta no es si se extraerán los bits restantes de información, sino cuándo.

**El Problema del Bit Único** Imagina un sistema de privacidad que oculta el 99% de los datos de tu transacción, pero filtra el 1% restante. Ese 1% podría parecer aceptable, pero la información se acumula. Un bit filtrado limita las posibilidades, y dos bits las limitan aún más. Cada filtración adicional estrecha las posibilidades de quién podrías ser, qué podrías estar haciendo y por qué.

Los adversarios son pacientes. Recopilarán piezas parciales de información a lo largo del tiempo, las correlacionarán entre diversas fuentes de datos y aplicarán técnicas estadísticas para extraer la "señal" del "ruido". Aunque una correlación temporal por aquí, un patrón de montos por allá y una conexión en el gráfico de red en otro lugar no sean individualmente suficientes para identificarte, pueden lograrlo una vez que convergen.

Recuerda, esto no es una hipótesis; es la metodología del análisis de cadena, el análisis de metadatos y de todo sistema de vigilancia moderno. La suposición de que las pequeñas filtraciones seguirán siendo pequeñas es incorrecta; las pequeñas filtraciones se acumulan para componer imágenes completas.

Cualquier sistema de privacidad que tenga filtraciones debe responder a la siguiente pregunta: ¿Qué sucede cuando un adversario con tiempo y poder de cómputo ilimitados optimiza su ataque contra esas filtraciones? Si la respuesta es "eventualmente ganan", entonces el sistema no proporciona privacidad; solo proporciona una exposición retrasada.

**Ofuscación vs Encriptación** Existen dos enfoques para ocultar información: puedes ofuscarla, haciéndola más difícil de encontrar entre el ruido, o puedes encriptarla, haciendo que sea matemáticamente inaccesible sin la clave.

Ofuscación es como esconder una aguja en un pajar. Funciona hasta que alguien construye un imán mejor.

La aguja sigue allí, y aún se puede encontrar con suficiente esfuerzo. La seguridad es económica, no matemática. Estás apostando a que encontrar la aguja cuesta más de lo que vale. Pero los costos disminuyen con el tiempo: la computación se vuelve más barata, los algoritmos más inteligentes y los adversarios más motivados. Lo que hoy está oculto, mañana puede ser trivialmente expuesto.

Encriptación es como destruir la aguja y conservar solo una descripción bloqueada de ella. Sin la clave, la descripción es indistinguible del ruido aleatorio. No existe un imán que ayude. No hay cantidad de computación que extraiga significado del azar. La seguridad es matemática, no económica, y no se degrada con el tiempo. Un mensaje encriptado de 2016 es tan seguro hoy como lo era entonces, suponiendo que la criptografía sea sólida.

Esta distinción es enormemente importante para la privacidad financiera. Los enfoques basados en ofuscación mezclan tu transacción con otras, para ocultarla entre señuelos o añadir ruido a los datos. Aunque estas técnicas aumentan el costo del análisis, no lo hacen imposible. A medida que mejoran las técnicas de análisis, la protección se debilita. La privacidad que era adecuada hace cinco años, hoy podría estar quebrantada, y la que parece adecuada hoy podría quebrantada con las herramientas de 2030.

Los enfoques basados en encriptación ocultan la transacción en sí misma; no hay transacción que analizar, solo una prueba de que ocurrió una transacción válida. Los datos no solo están oscurecidos, están ausentes, y por lo tanto son inmunes a futuros avances en técnicas de análisis; no se pueden encontrar patrones en datos que no existen.

**Por qué Esto Determina la Arquitectura** Esta es la razón por la que Zcash encripta las transacciones en lugar de simplemente ofuscarlas. El remitente, el destinatario y el monto no se ocultan entre señuelos ni se mezclan con ruido. En cambio, se encriptan. La blockchain almacena compromisos y pruebas, no datos oscurecidos. Por lo tanto, lo que la red ve es matemáticamente indistinguible de bytes aleatorios.

El argumento conciso es el siguiente: si se acepta que la privacidad debe ser absoluta, que las filtraciones parciales se acumulan hasta producir una exposición total, y que las capacidades de los adversarios solo aumentan con el tiempo, entonces el encriptado es la única arquitectura viable como solución permanente, mientras que la ofuscación es solo una medida temporal.

La elección no es entre más privacidad o menos privacidad. Es entre una privacidad que se mantendrá y una privacidad que eventualmente fallará. No hay punto intermedio.

### 5.4 El Argumento Macroeconómico

Hasta ahora, los argumentos a favor de la privacidad han sido filosóficos. Que la privacidad permite el progreso, que la transparencia equivale a vigilancia, y que la privacidad parcial termina fallando, son ideas que siguen siendo válidas en cualquier época. Sin embargo, no vivimos en una época cualquiera; en la actualidad, el entorno macroeconómico hace que la privacidad no solo sea valiosa, sino urgente.

**La Historia No Termina** La estabilidad de las sociedades occidentales modernas puede haber llevado a muchas personas a subestimar lo permanente que parece esa estabilidad. A lo largo de la historia y en distintas partes del mundo, la estabilidad es la excepción, no la regla. Los regímenes colapsan. Las monedas fracasan. Los ciclos de deuda se reinician. Los controles de capital aparecen de la noche a la mañana. Estos no son eventos raros ni hechos lejanos del pasado; son características del mundo moderno que le están ocurriendo a alguien, en algún lugar, en este mismo momento.

Solo en la última década, Chipre confiscó depósitos bancarios durante su crisis financiera y Grecia impuso controles de capital que impedían a los ciudadanos retirar su propio dinero. El sistema bancario del Líbano colapsó, atrapando los ahorros detrás de límites de retiro que han durado años. Argentina atravesó repetidas crisis monetarias con una regularidad preocupante. Nigeria restringió el acceso a divisas extranjeras. China endureció los controles contra la fuga de capitales.

Existe un patrón constante: cuando los gobiernos enfrentan presión fiscal, recurren a controles financieros. La economía nacional y los bancos centrales permiten que las cuentas bancarias sean congeladas, los retiros limitados, las transferencias bloqueadas y los activos confiscados. Por lo tanto, la cuestión pasa a ser qué activos pueden ser confiscados y cuáles no.

Históricamente, el oro ha servido como protección frente a escenarios de incertidumbre fiscal. Es difícil de confiscar a gran escala, complicado de rastrear y conserva valor incluso durante cambios de régimen. Sin embargo, el oro tiene una experiencia de uso muy deficiente en el mundo moderno, ya que debe adquirirse físicamente, verificarse su autenticidad, almacenarse de forma segura y transportarse, lo cual

implica un alto riesgo. Esa fricción limita su utilidad como reserva de valor práctica para la mayoría de las personas.

Bitcoin se suponía que sería el oro digital. En ciertos aspectos, podría decirse que lo es. Sin embargo, su transparencia crea una vulnerabilidad diferente. Si todas tus transacciones son visibles en un registro público, el Estado puede identificar fácilmente tus tenencias, rastrear tus movimientos y ejercer presión a través de canales legales. La transparencia que hace que Bitcoin no requiera confianza en intermediarios también lo hace vulnerable a ser objetivo de control.

**El Efecto de Trinquete de la Vigilancia** Las capacidades de vigilancia solo se mueven en una dirección: expansión.

Los gobiernos acumulan datos, construyen sistemas, contratan analistas y desarrollan nuevas técnicas de análisis. También pueden compartir información entre agencias e incluso entre fronteras. La infraestructura de vigilancia, una vez construida, no se desmantela; sino que se mejora.

La IA acelerará drásticamente estos avances. El reconocimiento de patrones, que antes requería equipos de analistas, ahora puede automatizarse. Los metadatos, que antes estaban aislados en distintos sistemas, ahora pueden correlacionarse a escala. El análisis de comportamiento, que antes tomaba meses, ahora puede realizarse en tiempo real. El costo de vigilar a cada persona tiende a acercarse a cero. El único límite es la cantidad de datos disponibles para analizar.

En blockchains transparentes, esos datos lo son todo. Literalmente cada transacción que has realizado queda preservada permanentemente, esperando herramientas de análisis más avanzadas. La blockchain no olvida, y tampoco lo hacen los adversarios que extraen información de ella.

Lo que haces hoy será analizado con las herramientas del mañana. Las transacciones que hoy parecen anónimas podrían ser fácilmente rastreables dentro de cinco años. Los patrones que hoy parecen ocultos entre el ruido podrían convertirse en señales evidentes cuando los algoritmos mejoren. Por eso, las decisiones que tomes en 2026 deben considerar el estado de la privacidad y de las tecnologías de análisis en 2030.

**El Precedente que Debemos Recordar** Una de las herramientas más efectivas del control autoritario es la divulgación obligatoria. No comienza con la confiscación, sino con la recopilación de información. Registra tu religión. Declara tus activos. Reporta tus asociaciones. Aunque estos requisitos se presentan como procedimientos administrativos y burocráticos, a menudo preceden a algo peor.

Una vez que la divulgación se vuelve obligatoria, las poblaciones pueden segmentarse, y los grupos pueden identificarse, analizarse y evaluarse. ¿Siguen una religión que desaprobamos? ¿Pertenecen a asociaciones que consideramos amenazantes? ¿Poseen activos que podríamos querer? La separación y la distinción preceden a la persecución; es una vez que los datos existen cuando las acciones específicas se vuelven posibles.

El control autoritario ha ocurrido dentro de la memoria de personas que aún viven, e incluso sucedió sin las ventajas de escala que proporciona la tecnología moderna. Los nazis utilizaban registros en papel y archivadores. Hoy en día, nuestras herramientas digitales permiten identificar y seleccionar a la gente sin realizar ningún esfuerzo.

Poseer activos privados es rechazar estas amenazas; es negar la premisa de que tu vida financiera debe ser legible para el poder. Es adoptar una postura contra una filosofía que ha demostrado ser catastrófica cuando se implementa.

La vigilancia impulsada por IA sigue expandiéndose constantemente. La instrumentalización de los sistemas legales contra grupos desfavorecidos está aumentando. Los controles de capital se están volviendo más comunes a medida que aumentan las presiones fiscales. La confiscación por motivos políticos ya no es algo impensable incluso en democracias desarrolladas.

La seguridad de tu riqueza no debería depender de quién gane las elecciones. Tus ahorros no deberían estar a una sola decisión política de ser confiscados. Tu privacidad financiera no debería depender de la buena voluntad continua de instituciones que han demostrado estar dispuestas a flexibilizar las reglas.

En resumen, el argumento macroeconómico a favor de la privacidad es que cosas malas han ocurrido, están ocurriendo y seguirán ocurriendo. La pregunta es si estarás preparado para enfrentarlas cuando lleguen a tu puerta.

### 5.5 El Punto de Inflexión en la Historia

Estamos en un punto de bifurcación. La infraestructura del dinero se está reconstruyendo. Las decisiones que se tomen ahora determinarán lo que será posible en el futuro, y los caminos divergen de manera

marcada.

**Dinero de Vigilancia** Un camino conduce a la visibilidad financiera total, donde cada transacción queda registrada, cada donación es analizada y cada compra contribuye a construir un perfil. Este resultado es la trayectoria del sistema actual.

Las monedas digitales de bancos centrales (CBDC) ya están siendo probadas en todo el mundo. Por ejemplo, el yuan digital de China ya se encuentra implementado a gran escala, el Banco Central Europeo está desarrollando el euro digital, y la Reserva Federal ha estudiado un dólar digital. Es importante señalar que estos sistemas están diseñados para habilitar la vigilancia, no para preservar la privacidad. El objetivo es aumentar la visibilidad: quién gastó qué, dónde, cuándo y con quién.

El dinero programable amplía aún más la lógica del control fiscal. Puede introducir fechas de expiración en la moneda que obliguen a gastar, restricciones sobre qué tipos de bienes pueden comprarse, sistemas de crédito social en los que el acceso financiero depende de puntuaciones de comportamiento, y pagos de estímulo que solo pueden utilizarse con proveedores aprobados. Nada de esto requiere teorías conspirativas; solo requiere que la infraestructura exista y que aparezcan los incentivos para utilizarla.

Las blockchains transparentes ya cumplen la parte de infraestructura, proporcionando vigilancia financiera sin la necesidad de construir CBDC. Los gobiernos no necesitan emitir una moneda digital cuando los ciudadanos registran voluntariamente sus transacciones en registros públicos. El resultado es el mismo: un panóptico, donde la actividad económica es legible para cualquiera que tenga las herramientas para analizarla.

El camino hacia la visibilidad financiera total termina con el dinero como instrumento de control. Ya no sería una herramienta de coordinación voluntaria, sino un mecanismo de gestión social. Si gastas "correctamente", te dejan en paz. Si gastas "incorrectamente", puedes ser marcado, restringido o bloqueado. La libertad de realizar transacciones se convierte entonces en un privilegio otorgado por el "Gran Hermano".

**Dinero de Libertad** El otro camino conduce a un dinero que no puede ser vigilado, censurado ni controlado. Las transacciones son privadas por defecto, y los saldos de las cuentas solo son visibles para sus propietarios, lo que hace que la actividad económica sea comprensible para los participantes pero opaca para los observadores externos.

Es importante señalar que este camino no conduce a la anarquía, entendida como una situación sin reglas. El resultado de este camino sí tiene reglas, pero reglas que son aplicadas por las matemáticas en lugar de por instituciones. No es posible gastar dos veces el mismo dinero porque la criptografía lo impide. No es posible inflar la oferta monetaria porque el protocolo lo prohíbe. Tampoco es posible falsificar transacciones, ya que no se tiene acceso a las claves necesarias. Las reglas están integradas en el propio sistema, aplicadas por los nodos de la red en lugar de por los gobiernos, y, lo más importante, son inmunes a modificaciones arbitrarias.

En este futuro, los mercados funcionan sin el efecto distorsionador de la observación constante. La coordinación entre grupos sigue siendo posible sin la influencia de la vigilancia. Las organizaciones disidentes pueden existir porque el apoyo financiero no puede rastrearse. La innovación continúa siendo posible porque la experimentación no puede ser monitoreada. De este modo, se preservan las condiciones para el progreso descritas en la sección 5.1.

**El Precedente de la Encriptación** Aún hay razones para creer que el camino de la libertad no está cerrado.

En la década de 1990, el gobierno de Estados Unidos intentó prohibir la fuerte encriptación. La NSA y el FBI argumentaban que las comunicaciones encriptadas favorecerían a criminales y terroristas, y promovieron sistemas de custodia de claves que permitirían al gobierno tener acceso mediante puertas traseras. Estas agencias federales clasificaron el software de encriptación como munición, lo que hacía que su exportación fuera ilegal.

Los cypherpunks se opusieron a estas medidas y finalmente las derrotaron, pero el encriptado se difundió de todos modos. Los investigadores publicaron algoritmos, los desarrolladores lanzaron software, y Internet adoptó TLS. Hoy en día, la encriptación no solo es legal, sino obligatorio. HTTPS es necesario para la banca, el comercio y la comunicación. El mismo gobierno que una vez intentó prohibir la encriptación ahora lo exige para proteger a los ciudadanos.

La transición de "la encriptación es peligrosa" a "la encriptación es necesaria" tomó aproximadamente dos décadas. Es muy probable que el dinero privado siga esta trayectoria. Actualmente, la privacidad financiera suele tratarse con sospecha: los reguladores

la consideran una herramienta para criminales, y los marcos de cumplimiento asumen la transparencia como norma por defecto. Sin embargo, los argumentos a favor de la privacidad en las comunicaciones, que hoy se consideran legítimos e importantes, también se aplican a la privacidad financiera: las personas necesitan protección frente a la vigilancia, el comercio requiere confidencialidad y la alternativa es un mundo donde los mecanismos de control están en todas partes.

Zcash es legal en Estados Unidos e incluso se negocia en plataforma de cambios reguladas. Ha operado durante casi una década sin ser prohibido. Esto no es un accidente. Refleja la misma lógica legal y política que protegió la encriptación: el derecho a usar herramientas criptográficas es defendible, y los beneficios de la privacidad van mucho más allá de quienes podrían abusar de ella.


Figure 6: Colas de depositantes frente a Northern Rock, septiembre de 2007. La primera retirada masiva de depósitos en un banco británico en 150 años..

**La Elección** Estos caminos son mutuamente excluyentes: no se puede tener dinero de vigilancia y dinero de libertad al mismo tiempo. Tampoco se puede tener privacidad financiera y monitoreo universal de transacciones. La infraestructura que se está construyendo actualmente determinará en qué mundo viviremos.

Elegir proteger tus transacciones no es solo una decisión financiera personal, sino también un voto por el tipo de futuro que queremos construir. Tus decisiones revelan tus preferencias, ya que cada transacción en el pool protegido fortalece la red, y cada usuario que adopta dinero privado lo vuelve más viable. La tecnología ya existe; ahora debemos decidir usarla.

El dinero de vigilancia conduce a un futuro donde la libertad económica es un permiso concedido por quienes tienen el poder. El dinero de libertad conduce a un mundo donde la libertad económica es fundamental y está garantizada por las matemáticas. ¿Cuál escoger?

## 6. Evolución y Economía

### 6.1 Generaciones del Protocolo

Zcash ha actualizado su criptografía central dos veces desde su lanzamiento, y con cada generación llegaron mejor rendimiento, mayor seguridad y menos supuestos de confianza. El protocolo actual es sustancialmente mejor que el de 2016.

**Sprout (2016)** El primer pool protegido demostró que una criptomoneda privada era posible, ya que por primera vez una red en producción ofrecía privacidad criptográfica respaldada por pruebas de conocimiento cero.

Sprout era básicamente un prototipo presentado como si fuera un sistema de producción. Crear una transacción protegida requería aproximadamente 40 segundos de cálculo y varios gigabytes de RAM. Sprout no podía usarse en teléfonos y apenas era usable en laptops. Por esta razón, la mayoría de las transacciones seguían siendo transparentes, simplemente porque protegerlas era demasiado costoso.

Sprout también requería una ceremonia de configuración de confianza, en la que seis participantes generaban los parámetros iniciales, cada uno tomando precauciones elaboradas para destruir sus contribuciones secretas. La ceremonia funcionó, pero dejó una pregunta incómoda: ¿Qué pasaría si alguien hubiera conservado en secreto los desechos tóxicos?

**Sapling (2018)** Dos años después, Sapling reemplazó la criptografía de Sprout por una mucho más eficiente. El tiempo necesario para generar las pruebas se redujo de cuarenta segundos a solo unos pocos segundos. Los requisitos de memoria bajaron a unas pocas decenas de megabytes, y las transacciones protegidas se volvieron prácticas en dispositivos móviles por primera vez.

Sapling también introdujo funciones que hicieron la privacidad más usable. Por ejemplo, las claves de visualización permiten a los usuarios compartir acceso de lectura a su historial de transacciones sin revelar la autoridad para gastar los fondos. Además, las direcciones diversificadas permiten que una sola

billetera genere miles de millones de direcciones de recepción no vinculables entre sí.

Es importante destacar que la configuración de confianza se mantuvo. Se realizó una nueva ceremonia llamada Powers of Tau, en la que participaron cientos de personas durante varios meses, seguida de una fase específica para Sapling. La ceremonia más grande aumentó la confianza en el proceso, pero el modelo de confianza seguía siendo el mismo: creer que al menos uno de los participantes fue honesto.

**Orchard (2022)** Orchard reemplazó todo el sistema de pruebas. Está construido sobre el sistema de pruebas Halo 2, y no requiere una configuración de confianza ni una ceremonia de generación de parámetros. Por lo tanto, no existen desechos tóxicos ni supuestos de confianza sobre eventos que ocurrieron años atrás.

El rendimiento de Orchard es comparable al de Sapling, aunque con pruebas ligeramente más grandes y sin requisitos de configuración inicial. La criptografía también está estructurada de manera diferente, utilizando un nuevo ciclo de curvas (Pallas y Vesta) diseñado específicamente para pruebas recursivas.

Orchard es el conjunto protegido que Zcash siempre estuvo destinado a tener. Las generaciones anteriores representaban la mejor tecnología disponible en su momento; Orchard es lo que se volvió posible cuando la investigación finalmente alcanzó la visión.

**Hoy en día** Orchard es el estándar por defecto para las nuevas transacciones protegidas. Algunas billeteras, como Zashi, dirigen automáticamente a los usuarios hacia Orchard y protegen automáticamente los fondos transparentes antes de gastarlos.

Sapling sigue recibiendo soporte, pero se está descartando progresivamente. Cumplió su función como puente entre el prototipo y un sistema listo para producción, pero Orchard es el destino final.

Sprout ya ha sido descontinuado. Aunque el conjunto aún existe en la blockchain, las billeteras ya no crean nuevas transacciones Sprout, y se recomienda a los aquellos usuarios con fondos allí, que los migren.

### 6.2 Torniquetes

La privacidad crea un problema de auditoría. En una blockchain transparente, puedes contar cada moneda. La oferta total es simplemente la suma de todos los saldos, y cualquiera puede verla. Si un error permitiera crear monedas de la nada, el aumento del total sería visible.

Los pools protegidos ocultan los saldos. No puedes sumar lo que todos poseen, porque no puedes ver lo que posee nadie. Entonces, si monedas falsificadas entraran en un pool protegido, ¿cómo se sabría?

La respuesta es: los torniquetes.

**El Mecanismo** Cada pool protegido tiene su propio torniquete, es decir, un registro acumulado del ZEC que ha entrado y salido del pool. Cuando las monedas se mueven del pool transparente a uno protegido, el torniquete registra el depósito. Cuando las monedas salen nuevamente hacia el conjunto transparente, el torniquete registra el retiro.

La lógica es simple. Si el torniquete muestra que 1 millón de ZEC han entrado en el conjunto y 800 000 ZEC han salido, entonces como máximo quedan 200 000 ZEC dentro. Si alguien intenta retirar 300 000 ZEC, algo está mal, la criptografía falló o alguien está intentando cometer fraude.

Los torniquetes no evitan la falsificación, sino que la detectan. Más precisamente, detectan cualquier intento de retirar monedas falsificadas. Es posible crear ZEC falsos dentro de un pool protegido (si alguien lograra romper la compleja criptografía), pero no podría gastar esas monedas en el pool transparente sin que se detecte la discrepancia.

**El Error de Sprout** En 2018, se descubrió una vulnerabilidad en la criptografía de Sprout. Se trataba de un fallo en el sistema de pruebas que podría haber permitido a un atacante crear monedas dentro del pool protegido sin ser detectado.

El error fue descubierto por el equipo de Zcash durante una auditoría de seguridad y corregido antes de que ocurriera cualquier explotación. Sin embargo, este episodio demostró la importancia de los torniquetes.

Si un atacante hubiera explotado el error, habría podido crear cantidades arbitrarias de ZEC dentro de Sprout, pero no habría podido extraer esas monedas de forma silenciosa. En el momento en que intentara mover ZEC falsificados hacia el pool transparente o hacia otro pool protegido, las cuentas del torniquete dejarían de cuadrar, y los auditores verían que habían salido más ZEC de Sprout de los que habían entrado.

Los torniquetes limitarían eficazmente el alcance del daño en caso de ataque, ya que incluso un fallo criptográfico catastrófico no produciría inflación inde-

tectable. El daño estaría limitado por la capacidad del pool, y cualquier intento de convertir ese valor falsificado en monedas utilizables activaría las alarmas.

### 6.3 Financiamiento del Desarrollo

Cuando Zcash se lanzó, tomó una decisión controversial: financiar el desarrollo directamente a nivel del protocolo. En lugar de depender de donaciones o del patrocinio de empresas, una parte de cada recompensa de bloque se destina directamente a organizaciones de desarrollo.

**Recompensa de los Fundadores (2016–2020)** Durante los primeros cuatro años, el 20 % de todas las recompensas de bloque se destinó a los fundadores, inversionistas iniciales, empleados y a la Fundación Zcash, mediante lo que se llamó la Recompensa de los Fundadores.

Esta decisión siguió siendo controversial, a pesar de que el acuerdo fue divulgado antes del lanzamiento, y de que cualquier persona que minara o comprara ZEC conocía los términos. Por un lado, los críticos lo veían como un impuesto a los mineros y una ganancia extraordinaria para personas con información privilegiada. Por otro lado, los partidarios lo consideraban una financiación necesaria para un proyecto que requería años de investigación criptográfica continua.

La Recompensa de los Fundadores terminó con el primer halving en noviembre de 2020, y cada beneficiario recibió exactamente lo que se había prometido. Actualmente, los fundadores ya no reciben recompensas del protocolo.

**Fondo de Desarrollo (2020-2024)** Antes de que la Recompensa de los Fundadores expirara, la comunidad debatió qué debía venir después. El resultado fue el Fondo de Desarrollo, una continuación de la asignación del 20 %, pero con una estructura diferente.

La nueva distribución dirigía: 7 % de las recompensas de bloque a Electric Coin Company (el equipo principal de desarrollo), 5 % a la Zcash Foundation (infraestructura y gobernanza), 8 % a subvenciones comunitarias administradas por un comité independiente. Los fundadores y los inversionistas iniciales fueron eliminados de este flujo de financiamiento.

El Fondo de Desarrollo funcionó entre el primer halving y el segundo halving, en noviembre de 2024.

**Fondo de Desarrollo Extendido (2024-2025)** A medida que se acercaba el segundo halving, la comunidad volvió a votar sobre la asignación y decidió extender el Fondo de Desarrollo con algunas modificaciones.

El financiamiento para el desarrollo continúa siendo del 20 % de las recompensas de bloque, pero ahora una parte se dirige a una "caja fuerte" controlada por mecanismos de gobernanza futuros, en lugar de organizaciones existentes. La intención es descentralizar gradualmente las decisiones de financiamiento, permitiendo que los poseedores de tokens tengan una influencia más directa sobre cómo se gasta el dinero destinado al desarrollo.

### 6.4 Gobernanza Descentralizada

Ninguna entidad controla Zcash de manera absoluta. El desarrollo, la infraestructura y la gobernanza están distribuidos entre organizaciones independientes con diferentes jurisdicciones, fuentes de financiamiento y mandatos.

**Las Organizaciones** Electric Coin Company (ECC) es el equipo principal de desarrollo del protocolo. Mantiene la implementación del nodo de referencia, desarrolla la billetera Zashi y lidera la investigación central. ECC es una subsidiaria del Bootstrap Project, una organización sin fines de lucro 501(c)(3) con sede en Estados Unidos.

Zcash Foundation maneja infraestructura, programas comunitarios y subvenciones. El equipo de la fundación desarrolló Zebra, una implementación independiente del nodo escrita en Rust, garantizando que la red no dependa de un solo código base. Es también una organización sin fines de lucro 501(c)(3) en EE. UU., operativamente independiente de ECC.

Shielded Labs se centra en investigación a largo plazo y desarrollo del ecosistema. Con sede en Suiza y financiada mediante donaciones, en lugar de recompensas del protocolo, aporta diversidad geográfica y estructural a la base de contribuyentes.

Tachyon: Liderado por el criptógrafo Sean Bowe, está construyendo la infraestructura para escalar Zcash. Bowe fue el arquitecto de Halo 2 y gran parte de la criptografía central de Zcash. Tachyon busca habilitar transacciones privadas globales mediante innovaciones en cómo las carteras sincronizan con la red sin filtrar información a los servidores.

Estas cuatro organizaciones colaboran pero no están obligadas a rendir cuentas entre sí. Pueden discrepar,

y a veces lo hacen. La diversidad de objetivos y perspectivas es una característica clave que previene la captura y asegura que múltiples perspectivas informen las decisiones del protocolo.

**El Proceso ZIP** Los cambios en el protocolo siguen el proceso de Zcash Proceso Mejorado (ZIP), lo que significa que cualquier persona puede proponer un cambio. Las propuestas se debatieron públicamente, se refinan mediante retroalimentación de la comunidad, y se aceptan o rechazan en función del mérito técnico y del consenso comunitario.

Las decisiones más importantes omiten el proceso ZIP y se resuelven mediante votaciones a nivel comunitario. Por ejemplo, las extensiones del Fondo de Desarrollo en 2020 y 2024 involucraron amplias deliberaciones públicas y recopilación de opiniones antes de su implementación. Se tomaron en cuenta las opiniones de poseedores de tokens, mineros y miembros de la comunidad.

Figure 7: La máquina Enigma, utilizada por la Alemania nazi para encriptar las comunicaciones militares durante la Segunda Guerra Mundial. Los operadores cambiaban la configuración diariamente, produciendo mensajes que parecían un conjunto aleatorio de caracteres sin sentido para los interceptores.

## 7. Zcash vs. . . .

La privacidad proviene del valor en reposo, no del valor en movimiento.

Este principio explica por qué la mayoría de las soluciones de privacidad fallan y por qué la encriptación en la capa base es la única arquitectura que realmente funciona. Una vez que se entiende esto, el panorama de las tecnologías de privacidad se vuelve mucho más claro.

### 7.1 Tornado Cash y Mixers

Consideremos lo que ocurre cuando se usa un mixer. Depositas fondos, esperas un cierto tiempo y luego los retiras a una dirección nueva. El objetivo es romper el vínculo entre la entrada y la salida, pero tanto el depósito como el retiro siguen siendo visibles. Un observador puede ver cuándo los fondos entran y cuándo salen. El mixer intenta ocultar qué entrada corresponde a qué salida, pero eso no garantiza privacidad.

En realidad, lo que hace es añadir privacidad al valor en movimiento, lo cual falla por una razón fundamental: los puntos de entrada y salida filtran información.

El depósito revela el momento y la cantidad. El retiro también revela el momento y la cantidad. Si esos datos se correlacionan, la privacidad se rompe. Por ejemplo: Si depositas 1.5 ETH y alguien retira 1.5 ETH una hora después la conexión puede volverse evidente. Los mixers intentan resolver esto utilizando denominaciones fijas y retrasos, pero la filtración de información sigue existiendo. Con suficiente información y análisis sofisticado, las correlaciones inevitablemente aparecen.

La IA empeora estos riesgos. El reconocimiento de patrones que antes era impráctico ahora puede resolverse fácilmente mediante: análisis de tiempo, agrupación por montos, patrones de comportamiento. Cada mixer se convierte así en un rompecabezas esperando a que los algoritmos encuentren la forma de resolverlo.

La única forma de separar realmente las transacciones entrantes y salientes es separarlas tanto en el tiempo como en el valor. La transacción entrante no debe causar la transacción saliente. Los montos y el momento de las transacciones deben estar no relacionados.

Por lo tanto, un sistema privado funciona como un verdadero depósito de valor. El dinero entra, permanece allí, pasa el tiempo, la vida continúa, hasta que finalmente salen montos no relacionados por razones no relacionadas. El depósito y el retiro no son dos partes de una misma operación, sino eventos independientes separados por meses o años de

almacenamiento real.

En teoría, podrías hacer esto con un mixer como Tornado Cash y dejar los fondos en el pool indefinidamente, pero esto es poco práctico porque no puedes hacer nada con esos fondos mientras permanecen allí.

Además, Tornado tiene pools de denominaciones fijas, por lo que no puedes enviar cantidades arbitrarias dentro del pool, no puedes transferir de una posición de Tornado a otra, ni usarlo para pagar a alguien o interactuar con alguna aplicación. Para utilizar tus fondos en cualquier cosa, debes retirarlos a una dirección transparente de Ethereum, exponiéndote nuevamente a la capa de vigilancia.

Zcash es diferente. Las transferencias protegidas entre direcciones protegidas son nativas, por lo que puedes recibir fondos, conservarlos, gastar cantidades arbitrarias, recibir cambio y volver a transaccionar, todo sin tocar nunca la capa transparente. Incluso es posible conectar desde el pool protegido a otras cadenas mediante Near Intents, pagando en ZEC protegido mientras el destinatario recibe el activo que desee. El pool protegido no es una sala de espera, es un sistema monetario completamente funcional.

Esta es la distinción arquitectónica que realmente importa. Los mixers son vías de escape de sistemas transparentes: los visitas, esperas y te vas. El pool protegido de Zcash es un destino: puedes vivir allí.

### 7.2 Monero

Monero es la criptomoneda de privacidad más utilizada después de Zcash. Representa un enfoque fundamentalmente diferente para el problema de la filtración de información, y al entender por qué su enfoque falla, podemos aclarar por qué el enfoque de Zcash funciona.

El enfoque de Monero consiste en utilizar firmas de anillo. Cuando gastas fondos, tu transacción incluye tu entrada real más 15 señuelos tomados de la blockchain. Un observador ve 16 posibles remitentes y no puede determinar cuál es el real.

Esto suena convincente: dieciséis posibilidades por transacción. El gasto real queda oculto entre muchos falsos. En realidad, esto significa que la privacidad es probabilística, no criptográfica.

Las agencias de aplicación de la ley han logrado rastrear transacciones de Monero. Incluso existe un caso documentado en el que la policía japonesa analizó transacciones de Monero para identificar y arrestar a dieciocho presuntos estafadores.

El problema fundamental es el conjunto de anonimato. Cada transacción de Monero se oculta entre 16 salidas, mientras que cada transacción protegida de Zcash se oculta entre todas las notas que alguna vez se han creado en el pool. Existen millones de estas notas, por lo que la privacidad que ofrece Zcash no es solo ligeramente superior, sino categóricamente superior.

Dieciséis es un número pequeño, definitivamente lo suficientemente pequeño como para ser atacado probabilísticamente, especialmente ahora que el análisis de tiempo, los patrones de montos y las heurísticas de comportamiento pueden reducir el conjunto de candidatos. Es decir, dieciséis es lo suficientemente pequeño como para que, con suficiente capacidad de cómputo y datos, eventualmente pueda ser descifrado.

No existe ningún ataque probabilístico que funcione contra un conjunto de privacidad compuesto por millones de notas. Es imposible reducir los candidatos mediante eliminación, porque no se elimina nada. La nota que gastaste permanece indistinguible de millones de otras para siempre.

Los desarrolladores de Monero comprenden esta limitación, y existe investigación activa para reemplazar las firmas de anillo con pruebas de conocimiento cero. En la práctica, esto significa que planean adoptar el enfoque de Zcash, lo cual es un reconocimiento implícito de que la privacidad basada en señuelos tiene un límite.

La diferencia es simple: Monero ofusca, Zcash encripta. La ofuscación se degrada con el tiempo a medida que mejoran las técnicas de análisis, mientras que la encriptación no.

Además de sus debilidades técnicas, Monero también carga con un peso cultural. La comunidad ha aceptado una asociación con usos ilícitos, lo que hace que su adopción institucional sea casi imposible. Esta es parte de la razón por la cual Monero ha sido retirado de prácticamente todos los grandes intercambios, mientras que Zcash sigue disponible en plataformas como Coinbase, Gemini y otras. La tecnología de privacidad necesita un camino hacia la legitimidad, y Monero ha hecho ese camino más difícil de lo necesario.

### 7.3 Pools de privacidad

Los pools de privacidad presentan un enfoque diferente para las soluciones de privacidad. En lugar de ocultarse entre señuelos aleatorios o encriptar todo, permiten que los usuarios demuestren que

no están asociados con actores maliciosos conocidos. Puedes retirar fondos de un pool demostrando que no provienen de direcciones sancionadas ni de transacciones marcadas.

El diseño es ingenioso. Los conjuntos de asociación te permiten definir con quién estás dispuesto a ser agrupado. De esta manera, pruebas que perteneces a un conjunto "limpio" sin revelar qué depósito específico es el tuyo. Así, los reguladores reciben la garantía de que los fondos no están contaminados, y los usuarios conservan cierto nivel de privacidad. En apariencia, todos quedan satisfechos.

Excepto que esto invierte el principio del debido proceso.

La premisa de los pools de privacidad es que debes demostrar tu inocencia. Es tu responsabilidad probar que tus fondos no están asociados con criminales, elegir un conjunto de usuarios "buenos" y proporcionar evidencia criptográfica de que perteneces a ese grupo. La suposición por defecto es la sospecha, y la carga de la prueba recae sobre ti.

En los sistemas legales funcionales, no tienes que demostrar que no eres un criminal: la acusación debe demostrar que lo eres. Los pools de privacidad normalizan lo contrario: que eres culpable hasta que demuestres tu inocencia mediante los conjuntos de asociación aprobados.

Las implicaciones son numerosas, ya que tu privacidad depende de lo que otros decidan revelar. Si los miembros de tu conjunto de asociación comienzan a demostrar su exclusión de ciertas actividades para limpiar sus propios nombres, los miembros restantes se vuelven más sospechosos. Existe una presión constante para demostrar más, revelar más y reducir aún más tu conjunto. El sistema genera efectos disuasorios por diseño.

No existe "encriptación conforme" para la mensajería. Signal no te pide demostrar que no estás conversando con terroristas; simplemente acepta que la privacidad en las comunicaciones es un derecho, incluso si eso implica que también pueda beneficiar a criminales.

No hay razón para que las finanzas deban ser diferentes. El argumento de que el dinero es algo especial o de que la privacidad financiera facilita de manera única el daño no resiste un análisis serio. Los criminales utilizan autos, teléfonos e internet, y aun así no exigimos pruebas de inocencia para conducir, llamar o navegar.

Los pools de privacidad intentan encontrar un punto intermedio entre la vigilancia y la libertad. Ofrecen una privacidad condicionada al cumplimiento, que requiere demostrar que la mereces y que puede retirarse si no logras convencer a otros de tu inocencia.

En esencia, es finanzas con permisos, pero con pasos adicionales.

### 7.4 Aztec y las L2 privadas

Las Capas 2 de Ethereum, cuando se complementan con funciones de privacidad, representan un trabajo de ingeniería serio. Proyectos como Aztec están construyendo rollups encriptados con criptografía sofisticada. La tecnología es sólida y el equipo es talentoso; esto no es una crítica a sus capacidades técnicas.

Fundamentalmente, Aztec y Zcash están resolviendo problemas diferentes.

Aztec es una plataforma de contratos inteligentes. Su propuesta de valor es la programabilidad privada: DeFi encriptado, cómputo confidencial y aplicaciones privadas. Esto es valioso porque permite casos de uso que Zcash no aborda. Si quieres interactuar con protocolos financieros complejos sin exponer tus posiciones, puedes usar una cadena de contratos inteligentes encriptada.

Zcash es dinero. Su propuesta de valor es ser un depósito de valor privado y un medio de intercambio. La idea es clara: es esencialmente Bitcoin encriptado. Un lugar donde guardar riqueza de forma privada durante años o décadas, con la confianza de que el sistema seguirá existiendo y funcionando.

Estos no son el mismo caso de uso, por lo que sus requisitos también son diferentes.

Un depósito de valor necesita ser Lindy. Debe sostener años de operación bajo condiciones adversas, sobrevivir a ciclos de mercado, presión regulatoria y desafíos técnicos sin colapsar. Zcash ya ha construido casi una década de este historial. Aztec es nuevo, y aunque su criptografía pueda ser perfecta, el sistema aún no ha sido probado a lo largo del tiempo. Esto puede ser aceptable para aplicaciones experimentales, pero no para la seguridad de grandes reservas de riqueza.

Un depósito de valor también necesita fuerza memética. Bitcoin tuvo éxito en parte porque "oro digital" es una narrativa poderosa que la gente entiende y cree. La idea de "Bitcoin encriptado" le da a Zcash un ancla similar, heredando las propiedades monetarias de Bitcoin mientras añade la privacidad que Bitcoin no tiene. Aztec no posee esa narrativa. Es simplemente una capa de infraestructura de privacidad, no una red monetaria.

Más allá del diseño técnico, también existe una capa social. La comunidad de Zcash se formó alrededor de un compromiso compartido con la privacidad como principio no negociable, y durante casi una década ha resistido presiones legales, políticas y reputacionales para debilitar ese compromiso. En contraste, un sistema de Capa-2 hereda en última instancia las normas y limitaciones de gobernanza de su Capa-1. En el caso de Ethereum, no está claro si la comunidad en general defendería de manera consistente la encriptación fuerte y la privacidad de las transacciones frente a presiones regulatorias. Para un activo destinado a funcionar como depósito de valor a largo plazo, esa incertidumbre en sí misma constituye un riesgo.

Es probable que Aztec y proyectos similares encuentren una demanda significativa para aplicaciones privadas, pero para el caso de uso central del dinero privado, un lugar donde la riqueza pueda descansar indefinidamente, cumplen un propósito diferente al de Zcash.

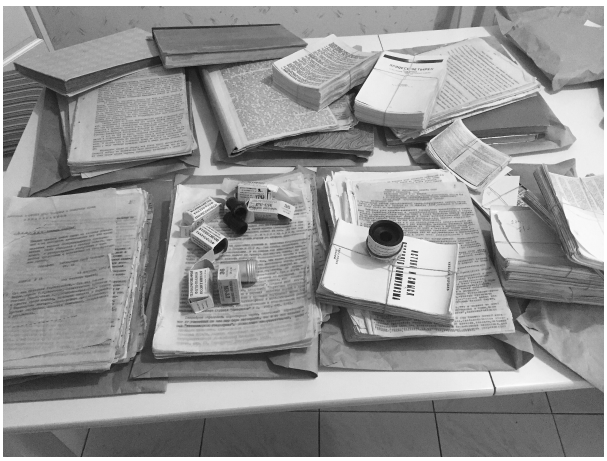


Figure 8: Samizdat — ciudadanos soviéticos copiando y distribuyendo literatura prohibida a mano para evadir la censura estatal. Su posesión podía significar prisión.

## 8. Conceptos erróneos

### 8.1 "Zcash No Es Privado Por Defecto"

Este concepto erróneo confunde lo que históricamente ha sido el comportamiento predeterminado de las billeteras digitales con el diseño del protocolo.

La idea errónea de que la privacidad no era el valor predeterminado surgió porque las primeras billeteras digitales utilizaban direcciones transparentes por defecto, principalmente por razones prácticas. En Sprout y Sapling, las transacciones protegidas eran computacionalmente costosas, y además los intercambios exigían depósitos transparentes. Por ello, el camino de menor resistencia solía ser utilizar direcciones transparentes.

Ahora Orchard ha hecho que las transacciones protegidas sean mucho más eficientes, y billeteras digitales como Zashi aplican el blindaje por defecto, moviendo automáticamente cualquier fondo transparente al pool protegido antes de permitir que se gaste. De esta manera, la experiencia de usuario se ha vuelto centrada en la privacidad.

La opción transparente sigue existiendo para casos de uso específicos, como: compatibilidad con intercambios, cumplimiento regulatorio, elección del usuario. Sin embargo, el camino predeterminado en el Zcash moderno es protegido de principio a fin.

### 8.2 "El Conjunto De Anonimato Es Pequeño"

Este concepto erróneo surge de confundir Zcash con sistemas basados en señuelos.

Como vimos anteriormente, en Monero tu transacción se oculta entre un número fijo de señuelos. Si hay 16 posibles remitentes, entonces tu conjunto de anonimato es 16. Por ello, muchos críticos asumen que Zcash funciona de forma similar: si pocas personas utilizan el pool protegido, entonces tu transacción se ocultaría entre unas pocas más.

Sin embargo, esto es incorrecto. Zcash no utiliza señuelos, sino pruebas de pertenencia a un árbol de Merkle.

Cuando gastas una nota protegida, demuestras que existe en algún lugar del árbol de compromisos, el cual contiene todas las notas que se han creado, sin revelar cuál de ellas es. El verificador solo aprende que tu nota es una entre millones, no entre cientos o miles, dentro del árbol.

El pool Orchard contiene millones de notas, y ese es el conjunto de anonimato para cada transacción protegida. Este conjunto crece con cada transacción y nunca se reduce.

El tamaño del pool transparente es irrelevante. Incluso si el 99 % de los ZEC estuviera en direcciones transparentes, el 1 % protegido seguiría teniendo un conjunto de anonimato compuesto por todas las notas protegidas que se hayan creado. Los dos pools son matemáticamente independientes.

### 8.3 "La Transparencia Opcional Debilita La Privacidad"

Este concepto erróneo asume que el pool transparente de alguna manera contamina el pool protegido.

En realidad, son dos sistemas independientes. El ZEC transparente y el ZEC protegido funcionan en paralelo. Las transacciones en el lado transparente no revelan nada sobre el lado protegido. Las garantías criptográficas de las transacciones protegidas no dependen de cuánto ZEC esté en direcciones transparentes.

Puede pensarse como dos libros contables separados que comparten la misma moneda: la actividad en uno no afecta las propiedades de privacidad del otro.

La opción transparente existe porque aporta valor real. Los intercambios pueden usar direcciones transparentes para depósitos y retiros, cumpliendo requisitos de cumplimiento regulatorio, mientras siguen listando ZEC. Esto permite que usuarios que necesitan auditabilidad puedan elegirlo, y que aplicaciones que requieren transparencia puedan construirse sobre él.

La transparencia opcional no compromete la privacidad del pool protegido; simplemente aumenta la capacidad de adopción de Zcash, algo que las cadenas completamente privadas por defecto no tienen. Un ejemplo claro de esto es que Monero ha sido retirado de los principales intercambios, mientras que Zcash sigue disponible en Coinbase y Gemini.

### 8.4 "Zcash Usa Una Configuración De Confianza"

Este concepto erróneo proviene de información que antes era cierta, pero que ya no lo es.

Sprout y Sapling requerían ceremonias de configuración de confianza, en las que los participantes generaban parámetros criptográficos y luego destruían los valores secretos utilizados para crearlos. Si alguien hubiera conservado esos secretos, podría haber falsificado pruebas y acuñar ZEC falsos.

Como se mencionó anteriormente, las ceremonias fueron muy elaboradas, con múltiples participantes, computadoras aisladas de internet, e incluso hardware que posteriormente fue destruido. A pesar de estas fuertes precauciones, el modelo de confianza introducía cierto grado de duda.

Orchard resolvió este problema utilizando Halo 2, un sistema de pruebas que no requiere configuración de confianza. Esto significa que: no hubo ceremonia, no existe residuos tóxicos, no hay riesgo de que alguien haya conservado secretos. Ahora, los parámetros provienen de datos públicos y verificables.

El pool protegido de Zcash ahora es sin necesidad de confianza, al igual que Bitcoin, y su seguridad está garantizada por las matemáticas criptográficas, no por la confianza en los participantes de una ceremonia.

### 8.5 "Hubo Un Preminado"

Este concepto erróneo es fundamentalmente incorrecto. No existían monedas antes del bloque génesis, por lo que no hubo ningún preminado.

La confusión surge por la Recompensa de los Fundadores, ya que durante los primeros cuatro años de Zcash, el 20 % de las recompensas de bloque se destinó a fundadores, inversionistas, empleados y a la Fundación Zcash. Sin embargo, esto no fue un preminado; simplemente era una porción de la emisión continua, creada mediante minería, igual que todas las demás monedas.

Esta distinción es importante. Un preminado habría creado monedas antes de que cualquiera pudiera participar, mientras que la Recompensa de los Fundadores generaba monedas al mismo ritmo que las recompensas para los mineros, pero las distribuía de forma diferente. Los mineros recibían el 80 % de cada bloque, y los fundadores el 20 % restante. Lo importante es que ambos provenían del mismo calendario de emisión.

Las condiciones de la Recompensa de los Fundadores fueron completamente divulgadas antes del lanzamiento. Tanto el documento técnico como el sitio web explicaban su propósito y su funcionamiento. Por lo tanto, cualquier persona que minara o comprara ZEC en 2016 sabía exactamente cómo funcionaba su distribución, y no existía ninguna asignación oculta, reserva secreta ni monedas que aparecieran de la nada.

La Recompensa de los Fundadores terminó con el primer halving en noviembre de 2020. En ese momento, cada beneficiario había recibido exactamente lo que se había prometido públicamente, y nada más.

### 8.6 "Los Desarrolladores Reciben el 20 % de las Recompensas de Minería"

Este concepto erróneo confunde dos programas distintos y a sus respectivos beneficiarios.

La Recompensa de los Fundadores funcionó de 2016 a 2020, destinando el 20 % de las recompensas de

bloque a fundadores, inversionistas tempranos, empleados y a la Zcash Foundation, y terminó con el primer halving. Por lo tanto, los fundadores no han recibido recompensas del protocolo desde 2020.

El Fondo de Desarrollo reemplazó a la Recompensa de los Fundadores y funcionó de 2020 a 2024. Este fondo también asigna el 20 % de las recompensas de bloque, pero a diferentes destinatarios: ECC recibe 7 % para el desarrollo del protocolo, Fundación Zcash recibe 5 % para infraestructura y subvenciones, subvenciones comunitarias, reciben 8 %, administrado por una comunidad independiente para financiar proyectos del ecosistema.

Contrario a algunos malentendidos, el Fondo de Desarrollo no existe para el enriquecimiento personal. Su objetivo es financiar organizaciones que emplean desarrolladores, mantienen infraestructura y otorgan subvenciones a proyectos del ecosistema. En otras palabras, el fondo paga por la mejora continua de Zcash.

La alternativa sería el modelo de Bitcoin, que depende principalmente de donaciones y patrocinio corporativo, un enfoque que también tiene sus propios compromisos. Zcash, en cambio, adoptó financiamiento a nivel de protocolo para apoyar un desarrollo sostenible, y casi nueve años de mejoras continuas sugieren que esta decisión ha sido justificada.

### 8.7 "La Fundación Zcash Controla Zcash"

Este concepto erróneo parte de la idea equivocada de que una sola entidad controla Zcash, lo cual no es cierto.

En realidad, cuatro organizaciones independientes contribuyen al protocolo: Electric Coin Company (ECC) desarrolla la implementación de referencia y la billetera digital Zashi. Fundación Zcash mantiene Zebra, una implementación independiente de nodo, y administra subvenciones. Shielded Labs realiza investigación desde Suiza. El equipo Tachyon, liderado por Sean Bowe, desarrolla infraestructura de escalabilidad.

Estas organizaciones operan en distintas jurisdicciones, con diferentes fuentes de financiamiento y mandatos distintos. Aunque colaboran en el desarrollo del protocolo, pueden discrepar entre sí y no responden a una autoridad común.

Esta separación fue diseñada deliberadamente. Si alguna organización fuera presionada, capturada o comprometida, las otras podrían seguir funcionando y mantener el sistema. El protocolo no depende de un solo equipo, y la existencia de dos implementaciones independientes de nodos significa que no hay una única base de código autoritativa.

En términos de gobernanza, Zcash está más descentralizado que muchos proyectos de criptomonedas. Incluso podría argumentarse que es más descentralizado que Bitcoin, ya que este último está dominado por una sola implementación de mecanismo y un pequeño grupo de administradores que deciden qué cambios se integran al código.

### 8.8 "El Mossad está detrás de Zcash"

Esta idea errónea no es más que una teoría conspirativa, sin pruebas que la respalden.

La conspiración apunta al hecho de que algunos fundadores tienen conexiones con Israel o al hecho de que hay criptógrafos académicos involucrados en el proyecto. Según esta lógica, cualquier tecnología desarrollada en parte por personas vinculadas a cualquier país está controlada por los servicios de inteligencia de ese país.

Zcash es de código abierto, literalmente cada línea de código es pública y auditable; su criptografía es matemática publicada, revisada por pares y escrutada por investigadores de todo el mundo. Si hubiera una puerta trasera, sería visible en el código y en las pruebas.

Además, cuatro organizaciones independientes, con sede en varios países, contribuyen al protocolo. La comunidad incluye desarrolladores, investigadores y usuarios de todos los continentes. Es simplemente irracional creer que una agencia de inteligencia controla un proyecto de código abierto distribuido a nivel mundial debido al origen de cualquiera de los primeros colaboradores.

El mismo pensamiento conspirativo podría utilizarse para atacar cualquier tecnología. Signal se desarrolló en parte gracias a subvenciones del Gobierno de los Estados Unidos, ¿significa eso que la CIA está detrás de Signal? Linux cuenta con colaboradores de todos los principales gobiernos y empresas, ¿significa eso que varios gobiernos lo han comprometido?

El código es de código abierto y las matemáticas son públicas, lo que invalida la conspiración.

### 8.9 "Los delincuentes utilizan Monero por una razón"

Esta idea errónea implica que los delincuentes habrían identificado necesariamente la tecnología de

privacidad más sólida para ocultar sus delitos, pero esto les da demasiado crédito.

Los delincuentes no son criptógrafos. No evalúan las implementaciones de curvas elípticas ni comparan las construcciones de conjuntos de anonimato. En cambio, utilizan lo que les resulta familiar y lo que ya tiene reputación en sus comunidades.

Monero construyó su marca en torno a ser la "moneda del crimen" y, por lo tanto, atrajo a los delincuentes. Esto demuestra un patrón de refuerzo entre la marca Monero y el uso de Monero por parte de los delincuentes, no la superioridad técnica de Monero.

La comparación de las capacidades de privacidad de Monero y Zcash favorece a Zcash. Monero oculta las transacciones entre 16 señuelos, mientras que Zcash oculta las notas entre más de millones de otras. Los señuelos de Monero pueden eliminarse mediante el análisis de la cadena con el tiempo, mientras que la indistinguibilidad criptográfica de Zcash hace imposible el descifrado de dicho análisis de la cadena. Monero no puede ser la elección de los delincuentes por razones de privacidad cuando las fuerzas del orden han logrado rastrear las transacciones de Monero, como lo demuestra el caso japonés mencionado anteriormente.

Los delincuentes también utilizan dinero en efectivo, teléfonos prepagos e incluso el correo electrónico estándar en sus negocios, pero nadie sostiene que se utilicen porque sean las opciones más seguras disponibles. Más bien, estos medios se utilizan porque son las opciones más accesibles y familiares.

Las elecciones de los delincuentes revelan decisiones basadas en el marketing y los efectos de red, no decisiones razonadas basadas en la solidez criptográfica.

### 8.10 "Monero es más privado porque todas las transacciones son privadas"

Este concepto erróneo sostiene que la privacidad obligatoria de Monero significa de alguna manera que es más seguro que el modelo de privacidad opcional de Zcash. La confusión surge al no distinguir entre los valores predeterminados del diseño y la solidez criptográfica.

Como se ha mencionado anteriormente, Zcash también es privado por defecto, ya que las billeteras digitales modernas aplican el blindaje. La ruta predeterminada está totalmente encriptada.

Incluso si las rutas predeterminadas de Monero y Zcash fueran diferentes, la distinción no determinaría su nivel de privacidad.

El mecanismo es más importante que la configuración.

Mecanismo de Monero: firmas en anillo con 16 señuelos, que ocultan su transacción entre 16 posibles remitentes. Dado que los señuelos pueden eliminarse con el tiempo mediante el análisis de la cadena, el conjunto de anonimato se reduce de forma retroactiva y se puede rastrear la conexión.

Mecanismo de Zcash: pruebas de conocimiento cero sobre un árbol Merkle de más de un millón de notas. Tu transacción podría haber gastado cualquiera de las notas, y no hay ningún proceso de eliminación para rastrear el origen. El conjunto solo crece y la indistinguibilidad criptográfica es permanente.

La configuración predeterminada de cerraduras débiles en todas las puertas no es preferible a la configuración predeterminada de una cerradura fuerte en las puertas que importan y la opción de añadir cerraduras a las demás. La privacidad débil obligatoria es simplemente privacidad débil, y la privacidad fuerte opcional es simplemente privacidad fuerte.

La pregunta correcta no es si la privacidad es la opción predeterminada, sino si la privacidad se mantiene bajo un análisis contradictorio. La privacidad de Zcash sí lo hace, la de Monero no.

Figure 9: El equipo de Zcash en sus inicios, con Zooko Wilcox-O'Hearn, cofundador de Zcash, y Jay Graber, entonces desarrollador junior del equipo de Zcash y que más tarde se convertiría en director ejecutivo de Bluesky, entre otros.

## 9. El camino por delante

### 9.1 Proyecto Tachyon

Tachyon aborda tres cuellos de botella en la escalabilidad de Zcash: la prevención del doble gasto, el escaneo de la cadena de bloques y el tamaño de las transacciones. La prevención del doble gasto es la más difícil de las tres, y su solución revela lo que hace que Tachyon sea un auténtico avance en lugar de otra optimización incremental.

**El problema de los anuladores** Zcash evita el doble gasto mediante anuladores: cuando se gasta

una nota, se revela un anulador, una cadena de caracteres aleatoria que funciona como un token de revocación. Los anuladores no pueden vincularse a las notas que revocan, pero si se intenta gastar la misma nota dos veces, se revela el mismo anulador y la red sabe que debe rechazar el duplicado.

El problema de los anuladores es que cada nodo validador debe almacenar todos los anuladores revelados, para siempre. No es seguro eliminar los anuladores antiguos, ya que alguien podría decidir volver a gastar una nota antigua. A cien transacciones por segundo, esto generaría aproximadamente un gigabyte de crecimiento de estado al día. Si no estás familiarizado con el tema, se trata de una cantidad extrema en comparación con la mayoría de las cadenas de bloques, incluidas las cadenas de alto rendimiento como Solana.

**Por qué fracasan las soluciones simples** La comunidad criptográfica sabe desde hace años que las pruebas recursivas podrían resolver este problema. En lugar de que la red rastree los anuladores, las pruebas recursivas permitirían a los usuarios demostrar que no han gastado dos veces. Se adjunta la prueba a la transacción y los validadores verifican la prueba y luego eliminan los anuladores antiguos.

El problema está en los detalles.

- **Enfoque 1:** Descargar el historial completo de la cadena a su billetera digital y construir la prueba localmente. Esto funciona criptográficamente, pero falla en la práctica, ya que su billetera digital soporta el ancho de banda y el costo computacional de cada transacción que realizan los demás, y los teléfonos no pueden hacer esto.
- **Enfoque 2:** Añadir un servicio intermediario. Envíe su transacción al servicio, deje que construya la prueba utilizando el historial completo de la cadena y, a continuación, difúndala. Esto funciona, pero introduce una latencia masiva. El servicio debe procesar toda la cadena para cada transacción y requiere que usted confíe en el servicio con sus datos de transacción.
- **Enfoque 3:** Envíe sus anuladores al servicio por adelantado, reciba las pruebas y, más tarde, adjunte esas pruebas a sus transacciones y las transmita. Esto puede parecer inteligente, pero tiene un defecto fatal: el servicio puede observar qué anuladores está preparando para gastar y, por lo tanto, puede vincular sus transacciones entre sí, revelando su primacía al intermediario.

**Sincronización inconsciente** Esta es la solución de Tachyon: un servicio que demuestra que no ha realizado un doble gasto realizando el cálculo, sin ver lo que aparece en la transacción final y sin saber qué anuladores está gastando. El servicio no puede distinguir sus transacciones de las de cualquier otra persona.

Técnicamente, esto se define como un servicio "desconocido". El servicio es ciego a los datos reales que procesa en su nombre, por lo que usted obtiene la ayuda computacional sin confiar en el ayudante.

El resultado son validadores que no almacenan el historial completo de anuladores. Por lo tanto, los usuarios no están expuestos a costes que varían en función de la actividad total de la red, y la indistinguibilidad del libro mayor, la propiedad de privacidad fundamental de Zcash, permanece intacta.

**Los otros cuellos de botella Blockchain** El escaneo, el proceso de identificar qué transacciones le pertenecen, se resuelve mediante cambios en el diseño del protocolo en lugar de una nueva criptografía. El requisito actual de descifrar cada transacción se sustituye por un protocolo de pago más eficiente.

El tamaño de la transacción y el tiempo de verificación utilizan las mismas técnicas de prueba recursiva. El tamaño marginal de la transacción y el tiempo de verificación se reducen aproximadamente a la escala de Bitcoin. Por lo tanto, una transacción Zcash totalmente privada acaba teniendo aproximadamente el mismo tamaño y velocidad que una transacción Bitcoin transparente.

**Qué permite esto** Una vez implementado Tachyon, las limitaciones de escalabilidad de Zcash serán las mismas que las de otras blockchains: ancho de banda y latencia. La sobrecarga criptográfica que encarecía la privacidad desaparece e incluso un teléfono puede realizar transacciones privadas sin procesar toda la cadena. Un nodo puede validar sin almacenar gigabytes de estado anulador.

La disyuntiva entre privacidad y escalabilidad, que durante mucho tiempo se ha considerado fundamental para el dinero encriptado, resulta ser un problema de ingeniería con una solución criptográfica.

### 9.2 Mecanismo de sostenibilidad de la red (NSM)

Bitcoin se enfrenta a un problema inminente: a medida que las recompensas por bloque se reducen a la

mitad hasta llegar a cero, las comisiones por transacción deben compensar a los mineros por mantener la seguridad de la red. Si las comisiones serán suficientes sigue siendo una pregunta sin respuesta, pero las alternativas previstas, como las emisiones residuales, van a romper el límite de 21 millones.

Zcash hereda este problema, pero el Mecanismo de Sostenibilidad de la Red lo resuelve sin romper el límite.

**El mecanismo** El NSM permite quemar ZEC del suministro circulante y reintroducirlos como futuras recompensas por bloque. Quemar 1 ZEC ahora provoca que se emitan 0,5 ZEC adicionales durante los próximos cuatro años, 0,25 durante los cuatro años siguientes, y así sucesivamente. La emisión sigue un modelo de decaimiento exponencial que se aproxima al calendario de reducción a la mitad de cuatro años existente.

A corto plazo, esto se traduce en una reducción del suministro circulante y un aumento de la escasez. A largo plazo, habrá más ZEC disponibles para las recompensas por bloque más adelante en la curva de emisión, lo que mantendrá los incentivos para los mineros sin superar el límite de 21 millones.

**Tres ZIP** ZIP 233 establece la quema voluntaria, lo que significa que los usuarios pueden donar directamente a la red Zcash en lugar de a organizaciones o individuos. Las billeteras digitales podrían ofrecer la opción de quemar ZEC al realizar transacciones. Las quemas verificables permiten crear comunidades con acceso restringido por tokens o insignias de identidad que demuestran la contribución a la sostenibilidad de la red.

ZIP 234 suaviza la curva de emisión, de modo que, en lugar de reducciones abruptas a la mitad, las emisiones disminuyen de forma continua. Esto proporciona un mecanismo predecible para reintroducir las monedas quemadas sin shocks repentinos en el suministro.

ZIP 235 quema el 60 % de las comisiones por transacción. Actualmente, esto supone aproximadamente 210 ZEC al año, lo que es una cantidad insignificante. El objetivo es establecer el mecanismo mientras las comisiones son bajas y los mineros no tienen ningún incentivo económico para oponerse a él. Las futuras estructuras de comisiones seguirán siendo una decisión de la comunidad que se tomará una vez que el NSM esté operativo.

**Aplicaciones futuras** El NSM crea una infraestructura para los casos de uso que la comunidad encontrará en el futuro:

- **Comisiones ZSA:** la acuñación, la transacción o el puenteo de activos protegidos de Zcash podrían quemar una parte para compensar a los titulares de ZEC.
- **Comisiones de soporte heredadas:** los usuarios que almacenan fondos en pools más antiguos podrían pagar comisiones, lo que incentivaría la migración a pools más nuevos y seguros.
- **Tarifas de incentivo a la privacidad:** el uso transparente de direcciones podría incurrir en tarifas para compensar la reducción del conjunto de anonimato.
- **Tarifas dinámicas:** Shielded Labs está desarrollando un sistema de tarifas basado en el mercado que sustituye la tarifa fija de 10 000 zatoshi por acción. El mecanismo calcula una tarifa marginal basada en la mediana de los 50 bloques anteriores, redondea a potencias de diez para preservar la privacidad y ofrece un carril prioritario 10 veces más rápido durante la congestión.

**¿Por qué ahora?** Actualmente, las tarifas de transacción son mínimas, por lo que implementar el mecanismo de quema ahora evita la dificultad política a la que se enfrentó Ethereum con EIP-1559, donde los mineros tenían fuertes incentivos para oponerse a la quema de tarifas. Si se implementa ahora, el precedente existirá para cuando las tarifas de Zcash se vuelvan significativas.

El NSM puede continuar la tradición de Zcash de mejorar el diseño de Bitcoin. La privacidad y el Fondo de Desarrollo ya diferencian a Zcash, y esta actualización añadiría una tercera diferenciación: un mecanismo para la sostenibilidad a largo plazo de la red que no está presente en Bitcoin.

### 9.3 Resistencia cuántica

La relación de Zcash con la computación cuántica es más matizada que la de la mayoría de las demás criptomonedas. El protocolo ya proporciona importantes protecciones de privacidad poscuánticas en escenarios comunes, como resultado de decisiones de diseño deliberadas tomadas desde el inicio del proyecto.

**Lo que ya está protegido** Los adversarios cuánticos no pueden comprometer el anonimato en cadena. Los anuladores de Zcash, el mecanismo que evita el doble gasto, utilizan funciones pseudoaleatorias

con clave basadas en criptografía simétrica, y estas primitivas siguen siendo seguras frente a los ataques cuánticos. Por lo tanto, los esquemas de compromiso están perfectamente ocultos, y el encriptado simétrico utiliza tamaños de clave diseñados para la seguridad poscuántica.

Esto contrasta fuertemente con otras criptomonedas privadas. Las imágenes clave de Monero, equivalentes a los anuladores, se volverían transparentes para un adversario cuántico, y el gráfico de transacciones quedaría al descubierto. La estructura de Zcash evita por completo esta vulnerabilidad.

**Las dos amenazas** Las computadoras cuánticas amenazan dos propiedades distintas: la privacidad y la solidez.

Las preocupaciones sobre la privacidad se centran en los ataques del tipo "recoger ahora, descifrar después". Un adversario podría recopilar datos de transacciones encriptados hoy y descifrarlos más tarde, una vez que lleguen los ordenadores cuánticos. Esto afecta principalmente a la distribución secreta en banda, el mecanismo para transmitir los detalles de las transacciones a los destinatarios. El diseño de Tachyon elimina por completo la distribución secreta en banda, protegiendo contra esta amenaza futura.

Las preocupaciones sobre la solidez se centran en la criptografía de curva elíptica, que podría ser descifrada por los computadores cuánticos. Aunque esto permitiría la falsificación o el robo, no comprometería la privacidad. Por lo tanto, las amenazas difieren en su urgencia, ya que las violaciones de la privacidad son retroactivas (las transacciones pasadas se vuelven vulnerables), mientras que las violaciones de la solidez a menudo no lo son (se puede reaccionar cuando aparecen los ordenadores cuánticos).

**Recuperabilidad cuántica** ECC ha desarrollado técnicas para la "recuperabilidad cuántica" en Orchard. Tras los próximos cambios en las billeteras digitales, y suponiendo que aparezcan las computadoras cuánticas, los usuarios podrán recuperar sus fondos a través de un mecanismo especial que impide que los adversarios cuánticos los roben, un mecanismo que también protege la privacidad.

El plazo para la integración de las billeteras digitales es que se lancen en 2026, por lo que los usuarios que protejan sus monedas y esperen estas mejoras estarán protegidos.

**Mejores prácticas actuales** Proteja sus monedas. El diseño del pool protegido ya proporciona una resistencia cuántica sustancial para la privacidad en cadena. Trate las direcciones como secretos siempre que sea posible. Los torniquetes siguen siendo la última defensa: incluso si se produjera una falsificación, esta acabaría siendo detectable cuando los fondos salieran de los pools protegidos.

Los criptógrafos de Zcash seguirán estando a la vanguardia de los avances, ya que el diseño modular del protocolo, que aísla los primitivos vulnerables, permite futuras actualizaciones sin necesidad de revisiones.

Figure 10: "El hombre del tanque" frente a una columna de tanques cerca de la plaza de Tiananmen en Pekín, el 5 de junio de 1989.

## 10. Conclusión

En conclusión, este artículo comenzó con una simple observación: a menos que utilices dinero en efectivo, todas tus compras se registran y almacenan indefinidamente. Bitcoin podría haber solucionado esto, pero no lo hizo. El blockchain que se suponía que nos liberaría de la vigilancia financiera se convirtió en la herramienta de vigilancia más completa jamás implementada.

Zcash tomó un camino diferente. En lugar de la transparencia por defecto, con la privacidad añadida como una idea de último momento, primero resolvió el problema más difícil: ¿cómo se verifican las transacciones sin verlas?

La respuesta requería pruebas de conocimiento cero, compromisos que ocultaran los importes, anuladores que impidieran el doble gasto sin vincular las transacciones y un modelo de notas que rompiera por completo el gráfico de transacciones. Siguieron nueve años de evolución del protocolo: Sprout demostró que la privacidad era posible, Sapling la hizo práctica y ahora Orchard la ha convertido en fiable.

El resultado de esta evolución es la imposibilidad de distinguir los registros. Dos transacciones protegidas no pueden diferenciarse, ni por observadores, ni por validadores, ni por estados nacionales con recursos ilimitados. Los datos no solo se ocultan o se mezclan con señuelos, sino que se cifran. Lo que ve la red es matemáticamente indistinguible del ruido aleatorio. Una auténtica caja fuerte suiza.

El camino por delante sigue siendo exigente. Tachyon elimina los cuellos de botella que limitan la escala. El NSM crea una economía sostenible. La resistencia cuántica es un problema solucionable en el que ya se está trabajando. Los cimientos están construidos. La criptografía funciona. La privacidad es real.

Ante nosotros se presentan dos futuros: uno en el que todas las transacciones son visibles, controlables y reversibles por quienquiera que tenga el poder, y otro en el que el dinero es tan privado como los pensamientos.

Zcash es la forma en que el dinero sigue siendo libre.

---

**Further Reading**

- El Individuo Soberano (The Sovereign Individual) de James Dale Davidson
  - Mi Tesis de inversion en Zcash (My Zcash Investment Thesis) de Frank Braun
  - Especificación del protocolo Zcash de Daira-Emma Hopwood et al.
  - Zcash: Guía para pasar de cero a héroe (Zcash: A Zero to Hero's Guide) de Arjun Khemani
  - Entendiendo Zcash: Visión general completa (Understanding Zcash: A Comprehensive Overview) de Youssef Haidar
  - Dentro de Zcash: dinero encriptado a escala planetaria (Inside Zcash: Encrypted Money at Plane-tary Scale) de CoinDesk Research
  - Argumentos a favor de una pequeña asignación a ZEC (The Case for a Small Allocation to ZEC) de Sacha
  - Dinero Libre de Arjun Khemani
  - Libro blanco de Bitcoin (Bitcoin Whitepaper) de Satoshi Nakamoto