Informe técnico de la plataforma

Versión: 0.11 Fecha: 2019-05-20

Resumen Ejecutivo

Chromia es una nueva plataforma blockchain para aplicaciones descentralizadas, concebida en respuesta a las deficiencias de las plataformas existentes y diseñada para permitir que una nueva generación de dapps escale más allá de lo que actualmente es posible. Chromia anteriormente se llamaba Chromapolis.

Motivación

Si bien plataformas como Ethereum permiten implementar cualquier tipo de aplicación en teoría, en *la* práctica tienen muchas limitaciones: mala experiencia de usuario, tarifas altas, experiencia de desarrollador frustrante, seguridad deficiente. Esto evita que las aplicaciones descentralizadas (dapps) se generalicen.

Creemos que para abordar estos problemas correctamente necesitamos repensar seriamente la arquitectura blockchain y el modelo de programación con las necesidades de las aplicaciones descentralizadas en mente. Nuestras prioridades son:

- Permitir que las dapps escalen a millones de usuarios.
- Mejorar la experiencia del usuario de dapps para lograr la paridad con las aplicaciones centralizadas.
- Permitir a los desarrolladores crear aplicaciones seguras con paradigmas conocidos.

Diseño técnico y características

Creemos que una cadena de bloques cumple el papel de una **base de datos compartida** dentro de un ecosistema de aplicaciones descentralizadas: almacena datos de aplicaciones y se asegura de que las adiciones, actualizaciones y transformaciones de datos estén autorizadas y sean consistentes con las reglas de la aplicación. Por esta razón, Chromia está diseñado y optimizado para cumplir el rol de una base de datos compartida de la mejor manera posible. Se implementa utilizando el marco postchain[1] existente desarrollado por ChromaWay, y cuenta con:

- Un modelo relacional[2]: Los datos de blockchain y el estado de la aplicación se almacenan en una base de datos relacional. Este modelo se considera el mejor de su clase en términos de flexibilidad, versatilidad y consistencia.
- Un lenguaje de programación relacional: Los backends de Chromia dapp están escritos en un lenguaje especializado que está profundamente integrado con el modelo relacional. Este modelo aumenta la productividad del programador y garantiza la coherencia de las aplicaciones.
- Escalado horizontal: Cada dapp tiene su propia(s) cadena(s) de bloques. Debido a que cada cadena de bloques es administrada por un subconjunto de nodos, es posible aumentar el rendimiento total aumentando el número de nodos.
- Indexación y consulta enriquecidas: las Dapps pueden recuperar rápidamente la información que necesitan directamente de los nodos que ejecutan la aplicación. La lógica de la cadena de bloques de Dapp puede realizar consultas complejas sin una degradación grave del rendimiento.
- Alto rendimiento de E/S: las consultas y actualizaciones de datos se delegan en una base de datos relacional muy optimizada, lo que permite a las dapps realizar un gran número de consultas y operaciones de actualización de datos.
- Consenso estilo PBFT[3]: Las transacciones se pueden confirmar en cuestión de segundos.
- Dapps de primera clase: Las dapps no surgen de “contratos inteligentes" en Chromia, sino que se consideran entidades de primera clase. Chromia ofrece a los desarrolladores de dapps un alto grado de flexibilidad y control.

---

1 https://chromaway.com/productos/postchain/

2 Codd, E.F (1970). "A Relational Model of Data for Large Shared Data Banks". Comunicaciones de la ACM. Clásicos. 13 (6): 377–87; https://dl.acm.org/citation.cfm?doid=362384,362685

- Aprovisionamiento de nivel de Dapp: la asignación de recursos a dapps en lugar de contratos da a los desarrolladores la libertad de crear sus propias políticas de uso de recursos y tarifas.

Chromia ofrece el mismo nivel de apertura, transparencia y descentralización que otras cadenas de bloques públicas. En Chromia los mineros son *reemplazados* por proveedores. Proveedores tienen[4] nodos que producen bloques. Se ha sugerido que los cuatro grupos de minería más grandes de Bitcoin[5] y Ethereum[6] podrían ejercer un control significativo sobre esas redes si se coludieron. Nuestro objetivo es garantizar que el número mínimo de proveedores de nodos cuya colusión sería necesaria para ejercer dicho control sobre Chromia supere este número significativamente. Por lo tanto, se puede decir que el modelo Chromia no tiende a la centralización más que las cadenas de bloques públicas más antiguas y confiables.

El consenso al estilo PBFT de Chromia se endurece aún más al anclar[7] cadenas de Chromia a una cadena de bloques de prueba de trabajo (PoW), probablemente Bitcoin o Ethereum. Esto asegura que la finalidad, la garantía de que las transacciones confirmadas no se pueden cambiar, es al menos tan fuerte como la de la cadena de anclaje elegida. Para alterar la historia de una porción anclada del historial de bloques de Chromia, sería necesario combinar la reorganización de la cadena de bloques PoW con una colusión maliciosa de un número suficiente de nodos de Chromia. La probabilidad de que cualquier atacante tenga los recursos para montar con éxito un ataque de este tipo es extremadamente bajo.

Usos

Chromia es una plataforma de propósito general que es adecuada para casi todo tipo de dapps. Es especialmente adecuado para casos que requieren una alta capacidad de E/S o una gestión de la mitad de conjuntos de datos complejos. Los juegos multijugador masivos en línea (MMOGs) son un ejemplo de tal caso.

Los juegos de blockchain se están volviendo cada vez más populares, pero los MMOGs están actualmente fuera de su alcance porque ninguna plataforma blockchain existente puede soportarlos. Chromia es capaz de alojar todo tipo de mundos de juego en la cadena de bloques, asegurándose de que evolucionan de acuerdo con las reglas predeterminadas y asegurando que nadie pueda hacer trampa. Creemos que la implementación de un MMOG será la mejor manera de mostrar las capacidades de Chromia. Los MMOGs tienen un conjunto muy exigente de requisitos; la capacidad de ejecutar MMOGs implica que Chromia es adecuado para dapps exigentes y complejos de todo tipo.

Justificación del diseño

Descripción general de los problemas con las plataformas existentes

Ethereum fue la primera cadena de bloques en ofrecer una plataforma para el desarrollo descentralizado de aplicaciones. Se crearon muchos prototipos de aplicación, pero los desarrolladores se enfrentaron a los siguientes problemas:

---

[3] Castro, M.; Liskov, B. (2002). "Practical Byzantine Fault Tolerance and Proactive Recovery". ACM Transactions on Computer Systems. Association for Computing Machinery. 20 (4): 398–461. https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.127.6130

[4] Decimos "propio" en lugar de control, porque el sistema de incentivos para los proveedores es de carácter económico: poseen recursos de Chromia y obtienen ganancias de ellos. Existe cierta tensión entre las nociones de propiedad y de control de esos recursos, ya que el control de los recursos de Chromia es posiblemente un criterio más relevante para evaluar la diversidad del grupo de proveedores.

[5] https://blockchain.info/pools

[6] https://www.etherchain.org/charts/topMiners

[7] Originalmente describimos el anclaje como “cadenas laterales" https://bitcointalk.org/index.php?topic=313347; una discusión más formal del anclaje se puede encontrar en el libro blanco de BitFury “On Blockchain Auditability". https://bitfury.com/content/downloads/bitfury_white_paper_on_blockchain_auditability.pdf

- Capacidad limitada. Debido a que la capacidad de la red es limitada y las tarifas de uso son proporcionales a la carga, las tarifas de transacción pueden ser de $ 1 o más para aplicaciones complejas. Este costo, que normalmente se paga por cada interacción con una aplicación, hace que la mayoría de las aplicaciones sean demasiado costosas para ser prácticas.
- Operaciones de E/S prohibitivamente costosas, por la misma razón. Por ejemplo, un contrato no puede recorrer en iteración una lista de usuarios, ya que el coste de esta acción podría superar el límite de gas en bloque. Por lo tanto, los desarrolladores tienen que saltar a través de aros para implementar algo tan simple como un pago de intereses a una lista de usuarios.
- Herramientas de modelado de datos deficientes y soporte deficiente para consultas. Los desarrolladores de aplicaciones tienen que recurrir a las capas de indexación y almacenamiento en caché d, o utilizar servicios de terceros que no proporcionan las mismas garantías de seguridad que la capa base.
- Lenguaje de contrato propenso a errores que ha dado lugar a muchos atracos de alto perfil.
- No hay provisión para actualizaciones de contrato en el nivel de plataforma, esta funcionalidad tiene que implementarse como una capa separada que aumenta aún más la complejidad.
- Los usuarios están obligados a pagar una tarifa por cada interacción y las confirmaciones son lentas. Esto da como resultado una experiencia de usuario (UX) deficiente.
- Pobre soporte al cliente ligero. Tres años después de comenzar los esfuerzos de desarrollo, la Fundación Ethereum todavía estaba luchando para ofrecer una billetera ligera de calidad de producción [8].

Las aplicaciones diseñadas con una gran audiencia en mente deben ser flexibles y receptivas. Se requiere una plataforma que permita al desarrollador asignar recursos de una manera que se adapte a sus usuarios. Incluso si Ethereum y otras plataformas actualmente en desarrollo abordan problemas de escalabilidad, no podrán proporcionar un grado suficiente de autonomía del desarrollador y seguirán siendo un entorno algo hostil para las dapps.

Creemos que para abordar estos problemas necesitamos repensar seriamente la arquitectura blockchain y el modelo de programación con las necesidades de las aplicaciones descentralizadas en mente.

Blockchain como base de datos

El papel principal de una cadena de bloques en un contexto de aplicación descentralizada es administrar los datos de una manera segura y consistente. Por lo tanto, una cadena de bloques se puede entender como una base de datos, más específicamente como una base de datos descentralizada segura. El papel principal de una cadena de bloques es la prevención del doble gasto, pero este es un caso especial de restricciones de coherencia de datos.

Las cadenas de bloques que están optimizadas para pagos, como Bitcoin, pueden adoptar modelos de datos altamente especializados (y optimizados). Pero una plataforma diseñada para hospedar diversas aplicaciones descentralizadas necesita un modelo de datos de propósito general.

La mayoría de las plataformas blockchain hoy en día utilizan almacenes de datos clave-valor (ejemplos: Ethereum, NEO, Fabric). Este modelo es, en teoría, completo y permite el uso de almacenes de datos de alto rendimiento como LevelDB. Sin embargo, este modelo es de muy bajo nivel y requiere que los desarrolladores de aplicaciones implementen funcionalidades básicas como la serialización y la indexación, un desafío desalentador.

---

8 el cliente ligero comenzó a ser algo utilizable alrededor de febrero de 2018.

Para agravar esto, las plataformas de cadena de bloques normalmente no exponen la funcionalidad completa de los almacenes de clave-valor, como la capacidad de usar claves de tamaño arbitrario e iterar a través de las claves. Por ejemplo, en la máquina virtual Ethereum (EVM) todas las claves son enteros de 256 bits y es imposible iterar a través de las claves almacenadas. Por estas razones, la implementación de un acceso adecuado a los datos indexados en el EVM es difícil e ineficiente.

Modelo relacional

El modelo relacional ha sido el estándar de oro para la gestión de bases de datos durante las últimas cinco décadas. Arraigado en las matemáticas y la lógica, se sabe que es capaz de modelar datos complejos de una manera eficiente. Por esta razón, y las razones indicadas anteriormente, consideramos que el modelo de datos relacionales es el eje de nuestra plataforma blockchain.

A medida que las aplicaciones descentralizadas se ocupan de estructuras de datos cada vez más complejas, las ventajas del modelo relacional se hacen cada vez más evidentes. Además, la mayoría de los ingenieros de software ya están familiarizados con él, por lo que no tendrán que aprender nuevos conceptos para implementar una aplicación.

Un modelo relacional también nos permite aprovechar el poder de los sistemas de administración de bases de datos SQL (DBMS) que se han optimizado durante décadas. En lugar de código dapp que atraviesa las celdas de memoria una por una, podemos enviar una consulta al DBMS y dejar que use su sofisticada planificación de consultas, estructuras de datos y capacidades de almacenamiento en caché para llevar a cabo la consulta lo más rápido posible.

Por supuesto, la elección del modelo de datos es una compensación. El modelo relacional puede tener las siguientes desventajas:

- El rendimiento es difícil de predecir y depende del planificador de consultas. Esto no es una desventaja significativa en el contexto de Chromia porque cada dapp se ejecutará de manera aislada; las consultas lentas afectarán sólo a la dapp que las realiza en lugar de al sistema en su conjunto.
- Es imposible imponer límites duros en el tiempo de ejecución de la consulta. Una vez más, esto no es un problema en Chromia porque afecta sólo al rendimiento de la aplicación que emite consultas lentas.
- La paralelización de la base de datos SQLs es un área compleja de investigación activa. Por lo que sabemos, ninguna plataforma blockchain ofrece una paralelización 100% totalmente automática a escala masiva. Por lo tanto, no hay evidencia de que un modelo relacional sea peor que otros modelos. Además, creemos que el modelo relacional hará que el particionamiento lógico y los mecanismos de cadena lateral sean más fáciles de implementar.

Aplicaciones descentralizadas de primera clase

En Ethereum todo el código vive en "contratos". No distingue entre contratos de billetera individuales y contratos multiusuario complejos: todos usan el mismo modelo de programación y medición de recursos. Una dapp basada en Ethereum usará uno o más contratos (posiblemente un contrato para cada usuario) y componentes interfaz. De hecho, muchas aplicaciones de Ethereum hacen uso del almacenamiento en caché centralizado, lo que hace que sus credenciales “descentralizadas” sean algo dudosas.

Si bien este enfoque es bastante elegante y puede escalar a diferentes tipos de aplicaciones, es muy inconveniente para las aplicaciones diseñadas para uso masivo. Los usuarios finales tienen que pagar por cada interacción con su aplicación, en proporción a las necesidades informáticas y de almacenamiento necesarias para su transacción. En otras palabras, Ethereum no da a las aplicaciones descentralizadas la flexibilidad para administrar los recursos por sí mismas. Por ejemplo, un modelo de negocio "freemium" es totalmente imposible. Esto crea una barrera para la adopción descentralizada de aplicaciones: la mayoría de los usuarios no están listos para pagar por cada clic. [9]

Chromia resuelve este problema aprovisionando recursos en el nivel de aplicación descentralizada:

- Cada dapp tiene su propia cadena de bloques (cadena lateral)
- Las tarifas (recaudadas para mantener los nodos) son pagadas por la dapp en su conjunto, no por los usuarios finales directamente

En consecuencia, las dapps son libres de implementar sus propias políticas de gestión de recursos, que pueden estar alineadas con las necesidades económicas en lugar de las técnicas.

Cada blockchain necesita un mecanismo antispam, pero este mecanismo no tiene que estar vinculado a las tarifas. Por ejemplo, una dapp podría permitir solo 1 acción de un usuario cada 15 segundos, por lo que un solo usuario no podrá enviar spam a la cadena de bloques con miles de millones de transacciones. Una dapp también puede mitigar los ataques de Sybil limitando el registro de nuevos usuarios a una tarifa razonable y / o requiriendo una invitación o un depósito.

En este modelo, no es necesario medir los recursos utilizados por cada operación. En su lugar, aprovisionamos recursos a la aplicación como un todo: cada cadena de bloques de dapp se ejecutará en un conjunto específico de nodos y, por lo general, tendrá su propio subproceso de CPU dedicado. Si una dapp necesita más de un subproceso de ejecución, puede constar de varias particiones, cada una de las cuales será una cadena lateral.

Esto elimina la sobrecarga de medición de recursos (ya no nos importa cuántas instrucciones se ejecutaron, ya que una aplicación no puede usar más recursos de los que se le dieron), lo que permite que las aplicaciones dapps funcionen más rápido y se escalen mejor.

Además de la programación, tener dapps como ciudadanos de primera clase en la plataforma permite que la economía de tokens se integre con el modelo de tarifas, es decir, las tarifas se toman de las ganancias "obtenidas" por una aplicación. También admite mecanismos para el gobierno y las actualizaciones de dapp que están integradas en la plataforma. Estas características se discuten con más detalle más adelante en este documento.

Modelo de programación

El marco Postchain en el que se basa Chromia nos permite utilizar el software de base de datos SQL de código abierto existente (específicamente, PostgreSQL) para capacidades de consulta y almacenamiento de datos. Sin embargo, no podemos permitir que las dapps realicen consultas SQL arbitrarias, ya que dichas consultas podrían ser inseguras, ambiguas o conducir a un uso excesivo de recursos.

La mayoría de las plataformas de blockchain dapp utilizan máquinas virtuales de varios tipos. Pero un arquitecto de máquina virtual tradicional no funciona muy bien con el modelo de datos relacionales Chromia, ya que necesitamos una forma de codificar consultas y operaciones. Por esta razón, estamos adoptando un enfoque más centrado en el lenguaje: se utilizará un nuevo lenguaje llamado Rell (**Rel**ational **language**) para la programación de dapp. Este lenguaje permite a los programadores describir el modelo/esquema de datos, las consultas y el procedimiento de desarrollo de código.

El código rell se compila en un formato binario intermedio que se puede entender como código para una máquina virtual especializada. Los nodos de Chromia luego traducirán las consultas contenidas en este código a SQL (mientras se aseguran de que esta traducción sea segura) y ejecutarán el código según sea necesario utilizando un intérprete o compilador.

Rell tendrá las siguientes características:

---

[9] de hecho, el hecho de que el costo para el usuario aumente a medida que más personas se unen a la red (mayor congestión de la red- > tarifas más altas) está totalmente en desacuerdo con las economías de escala sobre las que prosperan las tecnologías informáticas

- Seguridad de tipografía / comprobaciones de tipografía estáticas. Es muy importante detectar errores de programación en la fase de compilación para evitar pérdidas financieras. Rell tendrá mucha más seguridad de tipos que SQL y se asegurará de que los tipos devueltos por las consultas coincidan con los tipos utilizados en el código de procedimiento.
- Optimizado para la seguridad. Las operaciones aritméticas son seguras desde el momento, los programadores no necesitan preocuparse por los desbordamientos. Las comprobaciones de autorización son explícitamente necesarias.
- Conciso, expresivo y conveniente. A muchos desarrolladores no les gusta SQL porque es altamente verboso. Rell no molesta a los desarrolladores con detalles que se pueden derivar automáticamente. Como lenguaje de definición de datos, Rell es hasta 7 veces más compacto que SQL.
- Permite la meta-programación. No queremos que los desarrolladores de aplicaciones implementen los conceptos básicos desde cero para cada dapp. Rell permitirá el uso de plantillas.

Nuestra investigación indicó que ningún lenguaje o entorno existente tiene este conjunto de características, y por lo tanto el desarrollo de un nuevo lenguaje era absolutamente necesario.
Diseñamos Rell de tal manera que es fácil de aprender para los programadores:

- Los programadores pueden usar términos de programación relacional con los que ya están familiarizados. Sin embargo, no tienen que salir de su camino para expresar todo a través del álgebra relacional: Rell puede combinar sin problemas construcciones relacionales con programación de procedimientos.
- El lenguaje es deliberadamente similar a los lenguajes de programación modernos como JavaScript y Kotlin. Un lenguaje familiar es más fácil de adaptar, y nuestras pruebas internas muestran que los programadores pueden llegar a ser competentes en Rell en cuestión de días. En contraste, la sintaxis de estilo ALGOL de PL/SQL generalmente se siente poco intuitiva para los desarrolladores modernos.

El modelo de programación de Ethereum se describe normalmente como muy propenso a errores. Los errores en los contratos inteligentes de Ethereum han resultado en pérdidas por un total de cientos de millones de dólares[10]. En Chromia, nuestro objetivo es eliminar las fuentes más comunes de problemas a través de un mejor modelo de programación (sin interacciones extrañas entre diferentes contratos inteligentes como en el caso DAO [1112)] y lenguajes más seguros.

Dado que el código de Ethereum es inmutable, a menudo es imposible para un desarrollador arreglar su dapp a menos que conserve el control total, lo que lo hace no del todo descentralizado. En Chromia, las actualizaciones se pueden implementar a través de un mecanismo de gobierno y transición integrada.

Consenso y nodos

Información general del modelo

Está claro que el modelo de nodo completo no se escala particularmente. Si requerimos que los usuarios ejecuten un nodo completo que tenga una copia completa del estado del sistema, entonces las dapps están severamente limitadas en cuanto a los cálculos y recursos de almacenamiento que pueden usar.

Con el objetivo de lograr un mejor rendimiento a escala, proponemos un modelo en el que las dapps individuales se alojan en un subconjunto de nodos validadores que establecen un consenso sobre cualquier modificación en el estado de la dapp y manejan las consultas de los clientes. El sistema debe permitir a cualquier usuario ejecutar un nodo de réplica completo si lo desea, pero el sistema no debe depender de estos nodos de réplica para las operaciones. []

Mecanismo de control de Sybil

La investigación realizada por nuestro equipo indica que los mecanismos de control de Sybil comúnmente utilizados como PoW y Proof of Stake (PoS) son insatisfactorios[13][14][15]: ninguno de ellos garantiza un nivel suficiente de mitigación de ataques de Sybil, o incluso una medida particularmente buena de descentralización. La evidencia indica que la mayoría de las cadenas de bloques basadas en PoW, incluido Bitcoin, podrían estar controladas de facto por un pequeño grupo de entidades. Este problema es particularmente malo para las criptomonedas más pequeñas que no.

Sin embargo, tienen un ecosistema minero independiente. PoS también viene sin garantías de descentralización, y DPoS[16] en particular es propenso a la formación de cárteles y sobornos.

Por lo tanto, en lugar de seguir los enfoques comúnmente utilizados, diseñaremos el consenso de Chromia y los mecanismos de control de Sybil a partir de los primeros principios.

Lo que Chromia está tratando de lograr se puede comparar con la computación en la nube: una aplicación que utiliza redundantemente múltiples proveedores de alojamiento en la nube se puede considerar una aplicación descentralizada, en el sentido de que el fracaso o la censura de un solo proveedor de alojamiento cloud no resulta en un cierre de toda la aplicación. Un modelo de computación en la nube también permite a los usuarios usar clientes ligeros en lugar de hospedar una réplica completa del back-end de la aplicación en su dispositivo personal.

Los roles esenciales en el modelo de Chromia se definen de la siguiente manera. El software Chromia se ejecuta en nodos, instancias físicas o virtuales de potencia de computación. Los nodos están controlados o tal vez son propiedad de algún tipo de individuo, organización o colectivo al que nos referimos como proveedor. *Los usuarios* se conectan a estos nodos para publicar transacciones, consultar datos o sincronizar sus réplicas privadas.

Una red bizantina tolerante a fallos se distingue de una red meramente tolerante a fallos por su capacidad para tolerar comportamientos *arbitrarios y potencialmente maliciosos* por parte de los participantes de la red. El concepto de nodos es suficiente para diseñar una red tolerante a fallos, pero para apuntar a la tolerancia fallos bizantina adecuada debemos tener en cuenta las entidades proveedoras conscientes con el potencial de coordinar varios nodos.

Crucialmente, para mantener una dapp descentralizada, necesitamos asegurarnos de que los nodos que ejecutan sus blockchain (s) pertenecen a proveedores diferentes y no *coludidos.* En ese caso, la aplicación puede tolerar que un subconjunto de proveedores experimente errores, se vea comprometido o realice acciones hostiles.
Para que esto funcione, los participantes de la red necesitan i) saber qué nodos controla cada proveedor y ii) asegurarse de que los proveedores son *realmente* distintos. Esto último no se puede hacer mecánicamente, pero se puede hacer socialmente. Hay mucha evidencia de que Microsoft y Google son proveedores diferentes, pero no hay una forma mecánica de demostrarlo.
Creemos que *todo* consenso descentralizado depende en última instancia del "consenso social". Los sistemas descentralizados totalmente automatizados son una fantasía, al final son las personas las que determinan las reglas del sistema. Chromia reconoce esto, y lo incluye como un principio de diseño fundamental. En la práctica, la distinción del proveedor se logrará de la siguiente manera:

---

[10] Una lista de las vulnerabilidades más graves de Ethereum se puede encontrar aquí: https://www.dasp.co/

[11] https://blog.ethereum.org/2016/06/17/critical-update-re-dao-vulnerability/

[12] https://www.wired.com/2016/06/50-million-hack-just-showed-dao-human/

[13] Bentov, I., Gabizon, A., & Mizrahi, A. (01 de enero de 2016). Criptomonedas sin prueba de trabajo. https://arxiv.org/abs/1406.5694.

[14] Bentov, I., Lee, C., Mizrahi, A., & Rosenfeld, M. (8 de diciembre de 2014). Prueba de actividad: Extender la prueba de trabajo de Bitcoin a través de la prueba de participación. https://eprint.iacr.org/2014/452.pdf

[15] https://download.wpsoftware.net/bitcoin/pos.pdf

1. Inicialmente, ChromaWay seleccionará un conjunto de proveedores distintos. Creemos que nuestro amplio conocimiento de blockchain y la industria de TI nos permitirá elegir bien, y estamos incentivados a seleccionar proveedores que los usuarios aceptarán. Los usuarios que están preocupados por la singularidad del proveedor son bienvenidos a hacer su propia investigación y contribuir al proceso de toma de decisiones.
2. Eventualmente, una vez que el sistema tenga un conjunto suficientemente diverso de proveedores, permitiremos que los propios proveedores voten para agregar nuevos proveedores y el sistema ya no dependerá de ChromaWay como guardián.

Consenso

Cada cadena de bloques dentro de Chromia se asociará con un conjunto de nodos de validación que es un subconjunto de todos los nodos que pertenecen a Chromia. Este subconjunto de nodos ejecutará un algoritmo de consenso BFT. Dado que el tamaño del conjunto es limitado, los algoritmos similares esa PBFT son la opción optima l - están bien investigados, funcionan bien con conjuntos más pequeños de validadores y proporcionan una finalidad definitiva, lo que hace imposible la reorganización.

Sin embargo, hay dos riesgos sistémicos con el consenso basado en firmas de este tipo que deben ser considerados:

1. La posibilidad de colusión entre proveedores.
2. La posibilidad de que la mayoría de los nodos podrían verse comprometidos a través de un exploit de "día cero" de algún tipo.
3. El primer riesgo es extremadamente sutil, y se discute en cierta medida en otra parte de este documento. Este último es generalmente difícil de defender contra, el mejor enfoque es fomentar una amplia gama de software y hardware en el ecosistema de proveedores. Incluso con las estrategias de mitigación en su lugar, la amenaza se ve agravada por el comportamiento de los comisos basados en firmas en condiciones de fallo. Se ha demostrado que es propenso a un fallo catastrófico[17], lo que significa que una ruptura en el consenso puede corromper la cadena hasta el punto de que se vuelve muy difícil de recuperar.
4. Por esta razón, decidimos implementar una capa adicional de protección anclando bloques en una cadena de bloques basada en PoW, como Bitcoin o Ethereum. Esto se puede hacer a bajo precio, una sola transacción de Bitcoin anclando la totalidad de Chromia cada poco bloque cuesta muy poco, y garantizará que la fuerza de confirmación de Chromia será al menos tan fuerte *como Bitcoin* para los bloques que están anclados. Por ejemplo, un usuario que prefiere confiar en la seguridad de Bitcoin puede esperar hasta que se confirme un pago entrante a través del anclaje de Bitcoin antes de enviar productos.

Compensación de nodos

Las Dapps requieren recursos computacionales y almacenamiento y deberían poder pagar a los proveedores por ellas. Se debe incentivar a los proveedores a ofrecer servicios de alta calidad a los dapps a precios competitivos. Chromia establecerá un mercado donde los desarrolladores de dapp y los proveedores de nodos pueden comprar y vender recursos.

ChromaWay actuará como un proveedor de nodos clave en las primeras etapas. Nuevos proveedores se unirán a medida que el ecosistema cobre impulso, con precios más bajos de los recursos estimulando las dapps y precios más altos estimulando a los proveedores. Con el tiempo se logrará el equilibrio del mercado. Estimamos que, a largo plazo, el costo de usar los recursos de nodo coincidirá aproximadamente con el costo de las plataformas de computación en la nube como AWS EC2.

---

16 Prueba de participación delegada, https://bitshares.org/technology/delegated-proof-of-stake-consensus/

Características misceláneas

Creemos que para cumplir con los requisitos de las aplicaciones descentralizadas de alto rendimiento Chromia tiene que cumplir con los siguientes requisitos:

- Tiempo de confirmación: ~ 1 segundo (necesario para una buena experiencia de usuario, interactividad del usuario en tiempo real ...)
- Tasa de transacción: >500 TPS por cadena lateral. La tasa general en todo el sistema es ilimitada.
- Capacidad de E/S: actualizaciones y lecturas de >100k por segundo

Las pruebas preliminares del marco Postchain demuestran que es posible cumplir y superar estos requisitos.

Chromia también vendrá con un SDK de cliente que admite el desarrollo por parte del cliente de aplicaciones descentralizadas. Se ofrecerán SDK para JavaScript (para habilitar aplicaciones basadas en navegador), Java y otros lenguajes. El SDK también permitirá el inicio de sesión electrónico único (single-sign-on) en toda la plataforma y una billetera para la administración de claves para ahorrar a los usuarios la molestia de registrarse en cada aplicación por separado.

Aplicaciones descentralizadas

Suponemos que el lector de este documento ya está familiarizado con el concepto de aplicación descentralizada. Nevertheless, tiene sentido aclarar a qué nos referimos exactamente, ya que está

---

[17] https://download.wpsoftware.net/bitcoin/pos.pdf

íntimamente conectado con el objetivo de la plataforma. Por 'aplicación descentralizada' nos referimos a una aplicación multiusuario que se aloja y proporciona de forma descentralizada. Es decir, ninguna entidad debe tener control sobre la funcionalidad de una aplicación de este tipo.

Los problemas potenciales con el control central son que la entidad controladora puede:

- Cerrar la aplicación
- Denegar el servicio a determinadas categorías de usuarios
- Monetizar a los usuarios violando su privacidad
- Quitar la funcionalidad que son valoradas por los usuarios

El software de código abierto y de punto a punto abordó el problema del control centralizado para ciertas categorías de aplicaciones, como el software de oficina y el uso compartido de archivos, pero el software que se basa en bases de datos alojadas en el servidor es mucho más desafiante. Bitcoin fue posiblemente el primero en lograr precisamente esto, creando una base de datos compartida descentralizada segura de transacciones financieras y habilitando aplicaciones de pago fuera del control de las entidades centralizadas.

Sin embargo, la "base de datos" de Bitcoin es extremadamente primitiva. Una base de datos descentralizada más avanzada permite descentralizar aún más aplicaciones y, probablemente, crear tipos de aplicaciones completamente nuevos que antes eran inconcebibles.

Las aplicaciones descentralizadas tienen los siguientes rasgos deseables:

- No controlado por una sola entidad.
- Idealmente, controlado por la comunidad de usuarios.
- No se puede apagar
- Resistente a la censura: el servicio no se puede negar
- Transparente, los usuarios pueden ver lo que está pasando

- Privacidad: los usuarios tienen control sobre sus datos
- Alta disponibilidad

No esperamos que las aplicaciones descentralizadas tengan *todas* estas características. De hecho, algunas características pueden contradecirse entre sí. Por ejemplo, una dapp puede permitir a la mayoría de los usuarios restringir el acceso a una minoría, en cuyo caso la dapp está controlada por los usuarios, pero no es resistente a la censura. En la práctica, los desarrolladores de aplicaciones aspiran a una relación razonable entre la descentralización y otras prioridades.

Aplicaciones transparentes

Algunas aplicaciones solo están parcialmente descentralizadas: solo los datos que son críticos para la transparencia se alojan en la cadena de bloques, mientras que el resto de la aplicación está centralizada. Estas aplicaciones se describen mejor como aplicaciones transparentes (tapps) que aplicaciones descentralizadas.

Muchas aplicaciones que se comercializan como dapps son, de hecho, tapps. Por ejemplo, CryptoKitties[18] almacena información de cadera de los propietarios de gatitos en la cadena de bloques de Ethereum. Puede ser cerrado unilateralmente por la empresa que lo controla, y por lo tanto no puede ser llamado descentralizado en un sentido significativo. Se puede cerrar de varias maneras diferentes:

- Cerrar el sitio web. Dado que el código de cliente no es de código abierto, sin el sitio web CryptoKitties se vuelve imposible jugar el juego.
- Cierre de contratos. La compañía detrás de CryptoKitties puede cerrar los contratos alojados en la cadena de bloques Ethereum.

Por lo tanto, en la práctica lo único que diferencia CryptoKitties de una aplicación centralizada es la transparencia.

Modelo de token

Los modelos tradicionales de financiación y monetización no funcionan bien para aplicaciones descentralizadas. El cálculo del valor realizado en un modelo de financiación tradicional se basa en el control de la "propiedad" centralizada, como los datos, la base de usuarios, la propiedad intelectual y las patentes. Una aplicación descentralizada idealmente pertenece a sus usuarios, un grupo diverso de partes interesadas que forman algún tipo de equilibrio mutuamente beneficioso. No hay una parte central para poseer activos, agregar valor y beneficiarse de esa actividad. Es por eso que necesitamos un tipo diferente de modelo de financiación que sea más compatible con la propiedad distribuida. Para que la propiedad se distribuya, es necesario denotar la propiedad de participación en el sistema con algún tipo de activo líquido o semilíquido. Esto permite cuantificar la proporción de participación de un actor determinado, le permite agregar valor sin controlar o someterse al control, e intercambiar ese valor de forma segura. Por lo general, esto se logra con tokens.
El modelo básico de token ICO se ve más o menos así:

1. Emitir tokens.
2. Vender tokens a los inversores.
3. Haz lo que quieras con el dinero.

En lugar de eso, Chromia proporcionará mecánicos que equilibran los intereses de los desarrolladores y los usuarios. Esencial para esto es el meta-token de Chromia llamado Chroma. Los tokens de las dapps se pueden respaldar automáticamente con Chroma, proporcionando liquidez y valor que es independiente de la inversión en la dapp en cuestión. Los inversores de dapps pueden ser compensados en Chroma a través de un contrato de participación en los beneficios. Para los desarrolladores, Chromia ofrece la oportunidad de obtener ingresos de dapps. Esto incentiva la creación y el mantenimiento de dapps de alta calidad porque mejores dapps generan más ingresos y crean más demanda de tokens propiedad del desarrollador. El modelo Chromia está diseñado para apoyar economías circulares sostenibles y fomentar

una relación mutuamente beneficiosa entre desarrolladores, usuarios e inversores.

El papel de Chromia

Chromia pretende ser la base de datos descentralizada para las aplicaciones descentralizadas. Una combinación de una base de datos descentralizada y código, que se ejecuta en dispositivos de usuario final (por ejemplo, aplicación móvil o de navegador), normalmente comprenderá toda la aplicación descentralizada. Veamos cómo Chromia habilita las funciones de dapp:

No controlado por una sola entidad.

Suponemos que después de crear una dapp, los desarrolladores harían que tanto el front-end como el back-end (es decir, partes que se ejecutan en Chromia) sean de código abierto. Esto permite que la aplicación se use y desarrolle sin involucrar necesariamente al desarrollador original.
Los datos que pertenecen a la aplicación serán alojados por Chromia. Esto se hace en dos niveles:

1. El sistema Chromia root consiste en un conjunto diverso de nodos que ejecutan blockchains de aplicaciones, administran conversiones de tokens, asignan compensación de nodos y otras funcionalidades principales.
2. Cada dapp seleccionará un conjunto de nodos igualmente diverso para administrar sus datos.

Ambos niveles son sistemas criptoeconómicos descentralizados, y por lo tanto podemos decir que la aplicación no está controlada por una sola entidad. Normalmente, los usuarios pagarán por los recursos necesarios para hospedar la aplicación. Un problema potencial es que el código de la aplicación podría conceder el control a alguna entidad centralizada. Lo ideal es que los usuarios exijan una revisión independiente y utilicen la aplicación solo si las estructuras de control son razonables.

Controlado por la comunidad de usuarios.

Chromia incluirá mecanismos de gobierno opcionales que permitirán a los usuarios controlar varios aspectos de la funcionalidad de dapp. Por ejemplo, las actualizaciones de código.

No se puede cerrar.

Como se mencionó anteriormente, Chromia permite el alojamiento descentralizado de aplicaciones, esto garantiza que una sola entidad no pueda cerrar una aplicación. Pero no podemos garantizar que una aplicación no pueda cerrarse mediante acciones legales, ya que las estructuras de raíces de Chromia serán vigiladas por pocas empresas (al menos dentro de los primeros años de su existencia) que tengan que cumplir las leyes. Por lo tanto, una aplicación podría tener que ser desalojada de Chromia.
Debemos tener en cuenta, sin embargo, que la aplicación pertenece fundamentalmente a los usuarios. Chromia es una plataforma de alojamiento público y completamente de código abierto. Si los usuarios no están de acuerdo con una decisión del gobierno de cerrar la aplicación, simplemente pueden mover sus datos a otro lugar, es decir, pueden configurar un Polis diferente (similar a una bifurcación en una cadena de bloques tradicional) en una jurisdicción diferente. Mientras los usuarios necesiten una aplicación y estén dispuestos a admitirla, no se puede cerrar.

Resistente a la censura.

En el modelo Chromia, los desarrolladores de aplicaciones normalmente delegarán las operaciones a los nodos. Los nodos procesan las solicitudes de usuario mediante un mecanismo de consenso. Por lo tanto, ni los desarrolladores ni los nodos tienen la capacidad de implementar la censura por capricho.

En teoría, es posible que varios nodos puedan coludirse para implementar la censura, pero luego los usuarios pueden exigir que la aplicación se mueva a otros nodos. Por supuesto, es posible que una aplicación tenga algunos componentes de censura (antispam, antiabuso, etc.) como features. Lo que es razonable depende de la aplicación en particular. Si los usuarios creen que la censura no está justificada, pueden bifurcar la aplicación y alojar una versión actualizada.

---

[18] Un juego popular que permite a los jugadores comprar, recoger, criar y vender varios tipos de gatos virtuales. https://www.cryptokitties.co/

Transparente.

Los datos de la aplicación se alojarán en varios nodos y el consenso de blockchain los hace inmutables una vez que se finalicen. Creemos que muchas aplicaciones tendrán la transparencia como única característica. Chromia es un proveedor de tecnología neutral, no hace cumplir por sí mismo la descentralización. En muchos casos, la transparencia ya es una gran mejora con respecto al statu quo.

Privacidad.

La privacidad es un tema complejo. Los datos de aplicación descentralizados suelen ser públicos, por lo que la aplicación debe diseñarse teniendo eso en cuenta. Por ejemplo, podría utilizar identidades seudónimas, construcciones criptográficas como hash, pruebas de conocimiento cero, etc.

Creemos que este enfoque es mejor que un enfoque tradicional basado en la confianza y el secreto de los proveedores de aplicaciones. En un modelo centralizado, si se viola la seguridad de un proveedor, la privacidad se ve 100% comprometida. En nuestro modelo, dado que los datos son públicos en primer lugar, no se pueden comprometer.

Chromia planea ofrecer funciones que mejoren la privacidad (para su uso en dapps) en el futuro.

Alta disponibilidad.

Chromia está diseñado para soportar fallas en los nodos. El número de fallos que puede soportar es un parámetro configurable. El número mínimo de nodos es de cuatro, en ese momento puede soportar un error de nodo. Si se desea una mayor disponibilidad, se puede utilizar un mayor número de nodos.

Calidad de la descentralización

Chromia pretende ser una plataforma técnica neutral en lugar de una autoridad moral, por lo que permitirá que las aplicaciones se alojen independientemente de su nivel de descentralización.

Sin embargo, creemos que la descentralización es importante, y es importante que los usuarios conozcan las características de la aplicación que están utilizando. Por esta razón, planeamos desarrollar directrices y criterios de evaluación. Las empresas independientes podrán clasificar las solicitudes según estos criterios. También animamos a los usuarios a exigir una auditoría de código independiente.

Arquitectura de la plataforma

En esta sección describimos la arquitectura de la plataforma, ampliando la sección “Justificación del diseño".

Postchain

Chromia se basa en el marco Postchain[19.] Postchain define interfaces entre los componentes de un sistema basado en blockchain y proporciona una serie de bloques de construcción para la creación de redes, consenso, criptografía, etc.

La principal diferencia entre Postchain y otros marcos de blockchain es que Postchain está diseñado para almacenar datos de blockchain (tanto el contenido de blockchain sin procesar como el estado de la aplicación) en una base de datos relacional. No solo eso, Postchain permite que la lógica de transacciones y el consenso estén completamente alineados con una base de datos relacional; por ejemplo, las transacciones que violan las restricciones en la base de datos son rechazadas y excluidas del consenso, no resultan en errores fatales de ningún tipo.

Postchain se implementa en gran parte en Kotlin y se ejecuta en la máquina virtual Java (JVM). La JVM es una de las máquinas virtuales más utilizadas, está orientada a casos de uso de servidores y tiene un gran número de bibliotecas disponibles. La JVM proporciona protección inherente contra vulnerabilidades como saturaciones de buffer, fugas de datos, etc.: controla el acceso a los objetos, realiza la comprobación de los límites de la matriz y no expone características propensas a errores, como los punteros sin procesar. Por lo tanto, las aplicaciones implementadas en la JVM suelen estar libres de problemas como la ejecución remota de código, incluso cuando contienen errores. Esto es muy importante para el software blockchain ya que la ejecución remota de código puede conducir a grandes pérdidas.

Kotlin endurece aún más las comprobaciones de tipo y, en particular, garantiza la seguridad nula dentro del código escrito en Kotlin. Creemos que el uso de un lenguaje de programación moderno diseñado para la

seguridad puede reducir el número de defectos y ayudar a asegurarse de que los defectos restantes no conduzcan a consecuencias drásticas.

Postchain permite que varias cadenas de bloques se alojen en una sola base de datos y permite que una cadena de bloques "vea" los datos que pertenecen a otra cadena de bloques cuando esos datos son finales (confirmados). Esto simplifica la implementación de la interacción entre cadenas de bloques, ya que las cadenas de bloques pueden referirse a datos compartidos sin ninguna sobrecarga o complejidad adicional. En particular, esto se puede utilizar para las transferencias de activos entre cadenas de bloques.

Cadenas

Chromia se divide en múltiples cadenas de bloques con el fin de lograr la escalabilidad horizontal. En este modelo, cada nodo solo necesita trabajar con datos relacionados con sus cadenas de bloques correspondientes. Esta arquitectura aumenta la escalabilidad y simplifica las actualizaciones, ya que una actualización de una sola cadena de bloques no tendrá ningún efecto en los demás.

El sistema general consiste en una serie de cadenas de bloques de "sistema" que son esenciales para la funcionalidad de Chromia y una serie de cadenas de bloques de aplicaciones que son específicas de aplicaciones particulares.

Cadenas del sistema:

Cadena de raíces.

Validadores: nodos raíz.

Propósito: realizar un seguimiento de la lista de nodos raíz.

Descripción: La cadena raíz es necesaria para que los clientes ligeros puedan validar cualquier dato dentro de Chromia sin descargar toda la cadena de bloques.

Cadena de directorios.

Validadores: nodos raíz.

Propósito: realizar un seguimiento de todos los proveedores, nodos, blockchains de aplicaciones y sus validadores. *Descripción:* La cadena de directorios es responsable de realizar un seguimiento de toda la información crítica y orquestar las operaciones del sistema.

Cadena raíz de tokens.

Validadores: como se define en el directorio.

Propósito: realizar un seguimiento de los tokens Chroma.

Descripción: la cadena raíz de tokens realiza un seguimiento de la distribución de tokens entre otras cadenas.

Cadena de anclaje.

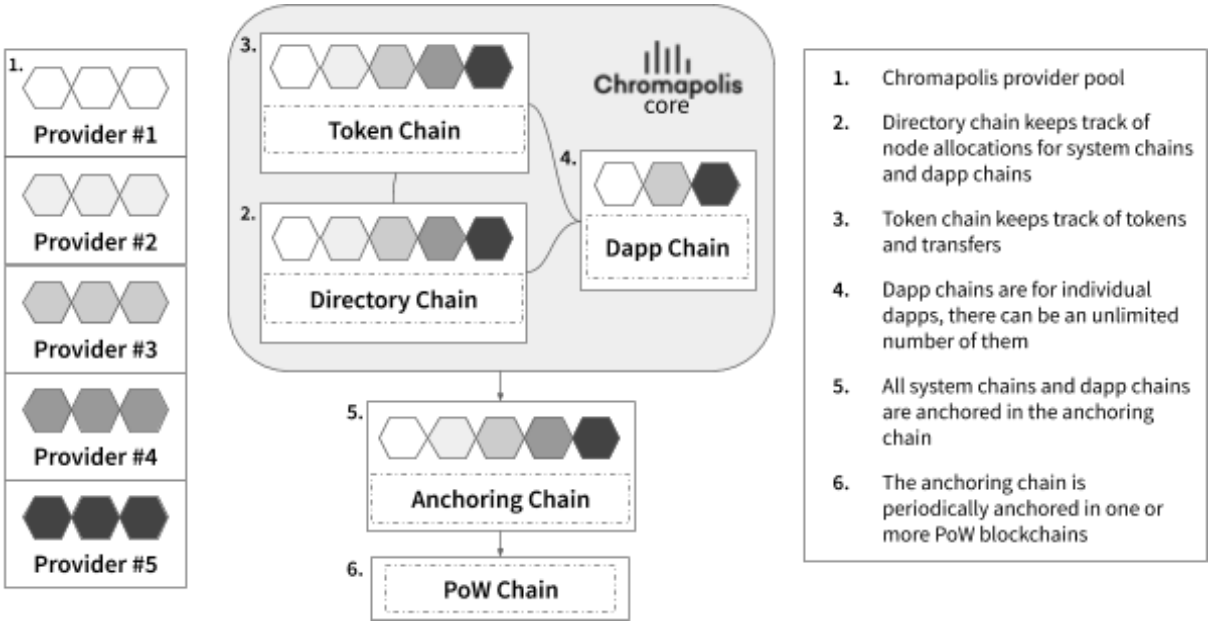
Validadores: como se define en el directorio.

Propósito: Defenderse contra ataques a un subconjunto de nodos.

Descripción: La cadena de anclaje registra hashes de bloques de otras cadenas. Esto permite detectar fallos de consenso. En caso de un error de consenso, los bloques anclados en la cadena de anclaje tienen prioridad sobre otras versiones de bloques. La cadena de anclaje está anclada en Bitcoin & Ethereum.

---

19 El código fuente se puede encontrar en https://bitbucket.org/chromawallet/postchain2/



(Las consideraciones de seguridad relacionadas con el mantenimiento de varias cadenas se explican en una sección independiente).

Implementación de nodos

El modelo de datos and operaciones necesarias para la funcionalidad del sistema, como la selección de nodos y recompensas, se puede implementar en Rell. El uso de un lenguaje declarativo de alto nivel puede simplificar la implementación y reducir la posibilidad de defectos.

Interacción con otras cadenas de bloques

La interacción con las cadenas de bloques Bitcoin y Ethereum es necesaria para el anclaje. La interacción de Ethereum es necesaria para permitir que ETH se use para pagos dentro de Chromia y para Chroma como un token ERC20. Esta funcionalidad se puede implementar a través de indexadores: los nodos que tienen que interactuar con Ethereum necesitan ejecutar un nodo Ethereum en paralelo al nodo Chromia e importar información de la cadena de bloques Ethereum a la base de datos Chromia.

Componentes

La siguiente es una lista de componentes de software que planeamos implementar para la versión Chromia MVP:

1. Compilador y entorno de tiempo de ejecución de Rell
2. Rell IDE: herramientas que facilitan el desarrollo
3. Client SDK: permite que el front-end (aplicación web o móvil) se conecte e interactúe con Chromia.
4. Nodo de Chromia, cadenas del sistema
5. Bitcoin & Ethereum soporte necesario para el anclaje
6. Contrato Chroma ERC20, puerta de enlace en el lado de Chromia
7. Contrato inteligente de conversión automática en Ethereum

Gobernanza

Chromia soporta diferentes estructuras de gobierno en los niveles de sistema y aplicación.

Gobernanza del sistema de croma

La gobernanza a nivel de sistema cubre los siguientes temas:

1. Actualizaciones del sistema, es decir, actualizaciones de las estructuras de la cadena de bloques del sistema, sus reglas, etc.
2. Ajuste de parámetros como el precio de ejecución de una dapp de acuerdo con las realidades económicas.
3. Aceptación de nuevos miembros en el sistema.
4. Exclusión de malos actores.

Obviamente, la gobernanza debe ser descentralizada, una sola entidad no debe tener control sobre el sistema. Creemos que los proveedores están en la mejor posición para realizar tareas de gobernanza:

- Pueden revisar profesionalmente las propuestas.
- Están motivados para mantener Chromia interesante tanto para los usuarios como para los desarrolladores de aplicaciones. Una mala decisión de gobierno afectará los ingresos y ganancias recaudados por los proveedores.

Por lo tanto, podemos requerir que 2/3 de los proveedores voten a favor de una propuesta de gobierno para aprobarla.

Centralización inicial

El lanzamiento inicial de Chromia MVP probablemente no tendrá una cantidad suficiente de proveedores independientes. Por lo tanto, en la etapa inicial, la gobernanza estará centralizada: todas las decisiones serán tomadas por ChromaWay en consulta con las partes interesadas del sistema. La transición a una gobernanza adecuada descentralizada ocurrirá cuando el sistema esté listo desde una perspectiva técnica y el ecosistema de proveedores esté saludable.

Alternativas rechazadas

Sin gobernanza formal

Algunas criptomonedas, por ejemplo, Bitcoin, se enorgullecen de no tener un gobierno formal. Podría funcionar bien si todo lo que quieren es "oro digital" -- después de todo, el oro en sí mismo no tiene gobierno. Pero Chromia es más complejo, y necesita ser capaz de responder a los desafíos de manera oportuna y coordinada, por lo que Chromia necesita un sistema de gobierno formal.

Skate

Un modelo de gobernanza generalizado en las cadenas de bloques que tienen gobernanza en cadena es el voto de las partes interesadas. Esto es particularmente común en las cadenas de bloques de DPoS, ya que la votación de las partes interesadas es una parte esencial de los mecanismos de control y consenso de Sybil. Hemos considerado a fondo este modelo y lo hemos rechazado por los siguientes motivos:

1. Por lo general, no es posible controlar la descentralización de la participación, es decir, los tokens pueden concentrarse en unas pocas manos, por lo tanto, no puede garantizar la gobernanza descentralizada.
2. No es justo en el sentido de que las partes interesadas ricas tienen más poder.
3. Muchos usuarios mantienen sus tokens en los exchanges, esencialmente permitiendo que los exchanges voten por ellos.
4. La votación al estilo DPoS parece ser particularmente propensa a problemas con sobornos, cárteles y centralización. Estos problemas se han observado en la naturaleza.
5. Incluso si los tokens se distribuyeron de manera más o menos uniforme, pocos usuarios realmente pasan por una molestia de votación, pocos usuarios pueden entender las propuestas, etc. Esto se demostró en el caso dado.

Usuarios Únicos

Es tentador dar un voto a cada usuario, lo que hace que la gobernanza sea más justa que "votar con dinero". Pero es imposible identificar usuarios únicos en un entorno descentralizado, y todavía se aplican

muchos problemas relacionados con el staking. En particular, es posible que los usuarios no estén lo suficientemente informados como para tomar buenas decisiones.

Aun así, planeamos experimentar con este tipo de gobernanza: nuestro plan es identificar un conjunto de usuarios que quieran participar activamente en la gobernanza: "ciudadanos de Chromia". El control de Sybil se puede implementar haciendo un seguimiento del gráfico social. No tenemos un plan inmediato para dar a estos usuarios ningún poder de gobierno formal, pero pueden emitir votos consultivos.

Gobierno de aplicaciones

Las diferentes aplicaciones tienen diferentes necesidades de gobierno:

1. Algunos están diseñados para ser *inmutables* y, por lo tanto, no requerirían gobernanza en absoluto.
2. Otros podrían ejercer la democracia directa y dar a cada usuario el derecho de voto.
3. Otra opción es implementar el voto ponderado, por ejemplo, proporcional a los tokens que uno tiene.
4. Los desarrolladores de Dapp también pueden desempeñar un papel en la gobernanza, y ya sea:
    a. Mantener el control total
    b. Trabajar junto con los usuarios a través de la votación, por ejemplo, el desarrollador hace propuestas que los usuarios pueden aprobar o rechazar

Queremos dar a los desarrolladores y usuarios la capacidad de decidir por sí mismos y experimentar con diferentes formas de gobierno a su ante plazca. Sin embargo, queremos asegurarnos de que los usuarios siempre tengan ciertas libertades:

1. La libertad de acceder y copiar los datos de la aplicación. Esto es una propiedad inherente de una cadena de bloques pública.
2. La libertad de bifurcar la aplicación. Esta es una propiedad inherente del software libre y de código abierto y los datos públicos: cualquiera puede hacer una copia modificada del software y ejecutarla en una copia de los datos.

Por lo tanto, no imponemos ninguna restricción que no sea una propiedad inherente de las aplicaciones que se ejecutan en cadenas de bloques públicas.

Chromia proporcionará herramientas que darían a los usuarios la capacidad de bifurcar una aplicación si están disgustados con su gobierno o simplemente quieren experimentar con algo diferente. Nuestro objetivo es asegurarnos de que esta bifurcación se pueda hacer de una manera suave y civilizada.

Usos

Chromia es una plataforma de propósito general adecuada para una amplia gama de aplicaciones. Sin embargo, la competitividad entre las diferentes cadenas de bloques es alta, por lo que tiene sentido centrarse en las fortalezas de Chromia:

- Chromia está centrado en la base de datos, como tal, es particularmente adecuado para aplicaciones que son similares a las bases de datos en su naturaleza, o que tratan con esquemas de datos complejos, consultas complejas, indexación, etc.
- Chromia tiene una excelente capacidad de lectura y escritura de datos, por lo que es especialmente adecuado para aplicaciones que requieren operar con grandes cantidades de datos.
- Chromia permite tanto consultas rápidas como confirmaciones rápidas. Por lo tanto, es adecuado para aplicaciones interactivas donde los datos deben mostrarse y actualizarse en cuestión de segundos.

- Chromia es muy flexible en términos de políticas de uso de recursos, por lo que puede adaptarse a modelos de negocio diferentes que no funcionan en otras cadenas de bloques anteriores.

Tokens

Los tokens son una parte esencial de las cadenas de bloques.

- Alta capacidad: nuestro objetivo es admitir 50 millones de transferencias de tokens por día por blockchain en la versión MVP del software. Esto no es un récord mundial, pero debería ser suficiente para admitir grandes bases de usuarios. La capacidad de transferencia de tokens se puede mejorar aún más en futuras versiones.
- Baja latencia: las transferencias se pueden confirmar en 2 segundos, lo que debería ser suficiente para admitir pagos en persona.
- Flexibilidad: la implementación del token es totalmente programable, se puede implementar cualquier característica imaginable.
- Políticas de tarifas personalizadas: la política de tarifas se decide por dapp. Esto significa que las transferencias pueden ser gratuitas, o sujetas a una tarifa plana, o una tarifa proporcional a la cantidad de comercio.
- Soporte nativo de múltiples tokens e intercambio atómico: el intercambio de tokens sin confianza se implementa en el nivel de formato de transacción, ni siquiera requiere ningún soporte especial en la dapp.
- Transferencia entre cadenas de bloques: los tokens se pueden mover entre diferentes cadenas de bloques dentro de Chromia. Las cadenas de bloques que no son de Chromia se pueden admitir en el futuro.
- Soporte de billetera delgada: una billetera delgada (por ejemplo, una billetera móvil o del navegador) puede validar las transferencias en cuestión de segundos, sin sincronizarse con una cadena de bloques.

Juegos

Los juegos basados en blockchain son un sector de rápido crecimiento de la economía criptográfica, pero la tecnología blockchain actual limita severamente lo que los juegos pueden ofrecer. Por lo general, una cadena de bloques se utiliza solo para alojar tokens negociables, mientras que el juego real ocurre fuera de la cadena de bloques.

Chromia puede permitir tipos de juegos mucho más avanzados donde todo el mundo del juego puede ser alojado dentro de la cadena de bloques, evolucionando con el tiempo de acuerdo con las reglas predefinidas. La actualización del estado del juego cada ciclo requiere un número de operaciones de lectura y escritura proporcionales al número de unidades del juego. Esto significa que las cadenas de bloques que no tienen una alta capacidad de lectura / escritura pueden admitir relativamente pocas unidades / jugadores.

En el EVM, cargar y almacenar una celda de memoria que ya no está vacía cuesta 5200 gas. El límite de gas de bloque en el momento de escribir es de 8.000.000. Por lo tanto, Ethereum puede hacer como máximo 1500 operaciones de lectura / escritura por bloque. Si toda la cadena de bloques de Ethereum estuviera dedicada a un solo juego, como máximo se podrían actualizar 6000 unidades (por ejemplo, mover) por

minuto. Una cadena de bloques pública basada en prueba-de-autoridad o “ejemplo, mover” llamada GoChain ofrece 136500000 gas por bloque y un intervalo entre bloques de 5 segundos. Esto implica 5250 actualizaciones de celda por segundo.

Para Chromia, apuntamos a al menos 100,000 actualizaciones de celdas por segundo en la versión MVP, ofreciendo una capacidad que es veinte veces mayor que las mejores cadenas públicas disponibles basadas en EVM. Planeamos aumentar este número en el futuro con el almacenamiento de estado de blockchain en memoria optimizada.

Aquí hay una lista de los beneficios de Chromia para aplicaciones de juegos:

- Carga rápida del cliente del juego (gracias a la capacidad de consulta avanzada, todo el estado del juego relevante para el usuario se puede transferir al cliente en cuestión de segundos)
- Interactividad: las actualizaciones se pueden confirmar en cuestión de segundos, los datos se pueden recuperar de blockchain en cuestión de segundos
- Alta capacidad de lectura y escritura (más de 100k actualizaciones por segundo)
- Buen soporte para esquemas de datos complejos necesarios para admitir mundos de juego
- Capacidad de actualizar el código a lo largo del tiempo
- Viene con contratos que pueden aportar liquidez automática para tokens del juego. El uso de tokens en juegos se tratará con más detalle en la sección “Token".

Usos empresariales

Basándonos en nuestra experiencia con aplicaciones de blockchain empresariales, creemos que Chromia se puede usar en aplicaciones donde los datos están abiertos o se pueden alojar abiertamente en forma encriptada, o solo los compromisos (hashes) deben revelarse. Esto puede ser particularmente relevante en las aplicaciones que están relacionadas con la transparencia. De hecho, la publicación de datos a través de una cadena de bloques privada difícilmente hace que las cosas sean más transparentes.

ChromaWay planea ofrecer la opción de almacenamiento basado en Chromia para su plataforma de contrato comercial Esplix, lo que permite a las empresas utilizar contratos Esplín sin la molestia de ejecutar sus propios nodos blockchain.

Tokens e incentivos

De manera similar a cómo se usan los tokens en Ethereum para pagar tarifas de transacción y compensar a los productores de bloques, los tokens Chroma se utilizan en Chromia para compensar los nodos productores de bloques.
Pero hay una diferencia: en el modelo Ethereum, las tarifas son pagadas directamente por los usuarios que realizan transacciones. En Chromia, las tarifas son pagadas por las dapps, que a su vez pueden cobrar tarifas a los usuarios. Esto se describe con más detalle en la siguiente sección.

Honorarios

Modelos de tasas de solicitud

En Chromia, los usuarios pagan tarifas indirectamente:

1. La dapp paga a los nodos que alojan las tarifas. La tarifa se paga diariamente desde la cuenta de token dapp y depende de los recursos computacionales solicitados por la aplicación y el volumen de datos utilizados.

2. La propia dapp puede cobrar tarifas de los usuarios de acuerdo con sus propias políticas.

Esto significa que no hay una política de tarifas para todo el sistema para los usuarios. Los desarrolladores de Dapp son libres de implementar cualquier política que deseen. Creemos que los siguientes modelos de

tarifas podrían ser relevantes:

1. Modelo clásico: se pagan tarifas por cada acción realizada. A diferencia de Bitcoin y Ethereum, el precio se puede fijar, las tarifas no necesitan estar basadas en la demanda.

2. Modelo de suscripción: el usuario paga una suscripción y luego puede realizar acciones sin pagos adicionales, sin embargo, estas acciones deben estar limitadas para evitar el abuso. Por ejemplo, en un servicio similar a Twitter, un usuario podría estar restringido a 50 mensajes por día.

3. Modelo Freemium: ciertas acciones pueden realizarse de forma gratuita, pero otras acciones pueden requerir una suscripción de pago. El modelo premium es muy común para las empresas de Internet.

4. Modelo subsidiado: una aplicación podría no cobrar tarifas de los usuarios, y en su lugar depender de una cuenta pre-financiada proporcionada por un patrocinador. Esto puede funcionar bien cuando los patrocinadores obtienen beneficios de usuarios fuera de blockchain, por ejemplo, la dapp podría estar disponible solo para nosotros que compramos un producto físico. Este modelo podría funcionar bien con los fabricantes de dispositivos IoT que patrocinan a los usuarios que compraron el dispositivo para el uso de una dapp relacionada.

5. Modelo basado en donaciones: los donantes ricos podrían donar tokens para proporcionar servicios a los usuarios de forma gratuita.

6. Conectado al juego: el usuario puede pagar tarifas indirectamente cuando realiza acciones en el juego:
    a. Comprar artículos en el juego, tierra, etc.
    b. Convertir fichas a “oro del juego"
    c. Artículos comerciales
    d. Pagar impuestos en el juego

Tarifas de alojamiento

En general, las tarifas de alojamiento de Chromia dapp no dependen de los recursos *consumidos* por una aplicación, sino de los recursos asignados para una aplicación. Esto es similar a cómo funciona el alojamiento de servidores dedicados y "privados virtuales": a la empresa de alojamiento no le importa lo que el servidor está haciendo realmente, quiere ser compensada por proporcionar un servidor. Este es también el modelo utilizado por AWS EC2, Google Cloud Compute Engine y servicios similares. En el espacio blockchain, EOS utiliza un modelo similar.

Las necesidades de las aplicaciones pueden ser muy diferentes. Algunas aplicaciones requieren una gran cantidad de recursos computacionales, algunas necesitan procesar un gran número de transacciones, algunas necesitan más espacio de almacenamiento, algunas necesitan una pequeña cantidad de almacenamiento muy rápido. El tipo de hardware que es óptimo para una aplicación depende de sus requisitos.

Por esta razón, introducimos diferentes clases de nodo. Es probable que los requisitos de clase evolucionen con el tiempo en función de las necesidades de las aplicaciones, la capacidad del proveedor, la disponibilidad del hardware, etc. A condición de que en el lanzamiento de MVP queramos introducir tres clases:

A. La clase más rápida para aplicaciones que requieren una alta tasa de transacciones o un procesamiento costoso. Especificaciones: CPU de 3 + GHz, dos hilos de hardware por blockchain, almacenamiento NVMe.

B. Clase media. Especificaciones: CPU de 2 + GHz, 1.5 hilos de hardware por blockchain, almacenamiento SSD.

C. Clase económica. Especificaciones: CPU de 1 + GHz, equivalente a un solo subproceso de hardware de 1 GHz por blockchain, almacenamiento SSD.

La tarifa de alojamiento de aplicaciones que se paga diariamente se divide en varios componentes:

1. Porcentaje de tiempo de procesamiento.
2. Número de transacciones.
3. Almacenamiento.

Chromia no tiene los medios para medir con precisión los recursos computacionales "consumidos" por una aplicación, ya que esto depende de una variedad de factores complejos que están fuera del control del código de Chromia (cachés de CPU, canalización de CPU, sobrecarga de cambio de contexto del sistema operativo, optimizaciones del motor de base de datos, etc.). En su lugar, Chromia medirá el tiempo medio necesario para procesar un bloque según lo informado por los nodos productores de bloques.

Cuando un subproceso asignado para una aplicación nunca está inactivo (es decir, continuamente se compila o aplica bloques), la aplicación está utilizando el 100% del tiempo de procesamiento. En ese caso, paga un precio completo por un día de alojamiento para una clase en particular.

Cuando una aplicación utiliza menos del 100% del tiempo de procesamiento, es elegible para un descuento. Para los nodos de clase A y B, el descuento está limitado al 50%. Incluso si la aplicación está completamente inactiva, todavía tiene que pagar la mitad del precio de alojamiento del día. Esto es necesario porque los recursos físicos reales se asignan a una aplicación si los utiliza o no. Se proporciona un descuento limitado porque queremos fomentar que las aplicaciones sean lo más eficientes posible. El tiempo de inactividad puede aumentar la capacidad disponible para otras aplicaciones, disminuir el consumo de energía y el desgaste del hardware.

Para el alojamiento de nodos de clase C no hay límite para el descuento y las aplicaciones que no construyen bloques no pagarán nada en costos de alojamiento. Además, la clase C permite a las aplicaciones especificar el “energía y”. Una aplicación que no quiere pagar más del 50% de la tarifa diaria de alojamiento se puede limitar para usar no más del 50% del tiempo de procesamiento. Los nodos de clase C utilizarán algoritmos especiales que permiten el co-alojamiento eficiente de un gran número de blockchains. Como resultado de esto, los nodos de clase C tienen como destino en lugar de garantizar su capacidad publicada.

Los costos de almacenamiento y los costos por transacción también dependen de la clase de nodos utilizados por una aplicación. Hospedar 1 GB de datos en nodos de clase C será mucho más barato que hospedar la misma cantidad de datos en nodos de clase A.

El precio del alojamiento se estandariza seleccionando la mediana de los precios presentados por todos los proveedores. En el futuro se desarrollará un mercado más sofisticado que permita a los proveedores subastar la capacidad sobrante una vez que el número de proveedores supere las necesidades de descentralización.

Incentivos de nodo

El proceso de construcción de bloques debe ser incentivado adecuadamente. Es decir, no debería ser rentable para los nodos descuidar sus deberes, por ejemplo, haciendo sólo bloques vacíos o ningún bloque en absoluto.

En teoría el colectivo de proveedores tiene interés en ofrecer un gran servicio a todas las aplicaciones. Si las aplicaciones se mueven a otras plataformas blockchain, los proveedores dejan de ganar dinero. Sin embargo, también tenemos que considerar a los proveedores que podrían intentar engañar al sistema para obtener un beneficio individual. Más allá del incentivo básico para no crear bloques no válidos o historiales

en conflicto (que pueden ser detectados y castigados automáticamente excluyendo automáticamente un nodo, y posiblemente su proveedor, del sistema), el sistema puede rastrear los siguientes datos:

1. Número de bloques construidos por un nodo para una cadena de bloques en particular como primario (el papel de primario se rota con el tiempo).
2. Número de transacciones en bloques creados por un nodo como principal.
3. Número de mensajes de confirmación enviados.

Estos datos se pueden utilizar para detectar nodos que descuidan su deber como principal o no son lo suficientemente rápidos como para enviar firmas de confirmación. Los nodos que sistemáticamente tienen un rendimiento inferior pueden ser excluidos automáticamente o a través del voto de los proveedores.

Tenga en cuenta que los nodos de en su conjunto tienen interés en aceptar tantas transacciones como sea posible y almacenar tantos datos como sea posible, ya que se pagan por el número de transacciones y el almacenamiento utilizado.

Otro recurso que otros sistemas de cadena de bloques suelen descuidar es la capacidad de un nodo para responder a las consultas. De hecho, si los nodos se compensan solo por la cantidad de datos procesados, se les incentiva ignorar las consultas y solo procesar transacciones. Pero si los usuarios ejecutan clientes ligeros, las consultas son absolutamente cruciales. Hemos desarrollado un mecanismo que crea un incentivo para que los nodos respondan a las consultas. Se explica en detalle en el Apéndice. En pocas palabras, al recibir una respuesta de un nodo, un cliente puede descubrir que esta respuesta es "afortunada" a través de un mecanismo similar a PoW. Sólo una fracción de todas las respuestas (por ejemplo, 1 en un millón) es "afortunada". Una respuesta afortunada se publica en una determinada cadena de bloques y produce una pequeña recompensa tanto al usuario como al nodo que produjo la respuesta. Se hacen disposiciones especiales (cubiertas en el Apéndice) para desalentar a los nodos de cultivar respuestas afortunadas por su cuenta.

Apuestas de nodo

Para alentar a los proveedores a proteger sus nodos, se les pedirá que coloquen tokens de Chroma en una cuenta separada que represente la participación del proveedor en la economía de Chromia y se use como garantía que se pierde cuando los nodos propiedad de un proveedor no se comportan de manera adecuada.

Los proveedores pueden agrupar nodos en unidades con diferentes niveles de participación: alto, medio, bajo. Los nodos de alto riesgo deben estar más completamente protegidos, ya que se pueden usar para aplicaciones altamente sensibles a la seguridad, como la ejecución de blockchains del sistema y dapps financieras de alto nivel. Los nodos de apuestas bajas se pueden usar para aplicaciones menos sensibles, como juegos simples. Cada dapp puede especificar una apuesta mínima que se requiere para los nodos que la ejecutan. El nivel de participación necesario para las cadenas de bloques del sistema es establecido por un consejo de proveedores.

Uso de tokens en juegos

La generación actual de juegos de blockchain se basan en elementos coleccionables y no ofrecen un juego rico en funcionalidades. Imaginamos una nueva generación de juegos multijugador en línea masiva con extensos mundos de juego alojados dentro de las cadenas de bloques de Chromia, y economías de mercado basadas en tokens y elementos de juegos comerciables.

Para este tipo de juego, Chromia puede ofrecer un conjunto de contratos inteligentes que hacen que los tokens de juego sean líquidos y valiosos. Esto permitiría a los desarrolladores de juegos arrancar rápidamente las economías de juego. Para los usuarios de juegos, los contratos inteligentes prefabricados ofrecen un cierto grado de estabilidad: pueden estar seguros de que los tokens de juego que ganan no perderán todo su valor de la noche a la mañana debido a una estructura de tokens codificada de manera deficiente.

En el corazón de los contratos inteligentes del juego Chromia, es un algoritmo de creación de mercado /

conversión de tokens similar a uno ampliamente conocido “algoritmo de Bancor" (un algoritmo similar fue descubierto por el equipo de Chromia antes de Bancor). Cuando los tokens Chroma se convierten en tokens de juego (por ejemplo, tokens de "oro" de juego), se crean nuevos tokens de juego. Los tokens Chroma se colocan en las reservas de contratos inteligentes y el precio se ajusta. Los ajustes de precios funcionan de tal manera que una mayor demanda (más personas comprando fichas de juego que vendiendo) resulta en un precio más alto. Cuando los tokens de juego se convierten de nuevo a Chroma, el precio se reduce. El algoritmo se puede configurar para permitir un movimiento de precio suave, por lo que el precio del token de juego contra Chroma no puede caer significativamente a menos que la gran mayoría de los usuarios abandonen el juego y conviertan sus tokens a Chroma.

Se puede cobrar una tarifa en el momento de la conversión ajustando el precio de compra / venta. Por ejemplo, se puede sacar una tarifa del 1% de la cantidad de Chroma y usarse para:

- Pague las tarifas de alojamiento de dapp de juegos (es decir, se transfiere a la cuenta de alojamiento de dapp)
- Pagar al desarrollador del juego y, posiblemente, a los inversores

El aumento del precio del token de juego con la demanda significa que los jugadores tienen un incentivo para invertir en oro de juego. De hecho, tienen un incentivo para descubrir nuevos juegos interesantes que van a crecer en popularidad con el tiempo. Indirectamente también tienen incentivos para promover y compartir los juegos que juegan. Este conjunto de incentivos puede resultar en una dinámica de juego saludable.

La lista completa de características de Chromia desarrolladas específicamente para su uso en aplicaciones de juegos se publicará en un documento separado.

Economía del token de croma

En resumen, el token de Chroma tiene las siguientes funciones en Chromia:

- Es utilizado por dapps para pagar tarifas de alojamiento, compensando así a los nodos.
- Se utiliza como una moneda “estándar” dentro de la economía de Chromia, ya que los dapps pueden cobrarlo como tarifas, o usar como reservas para pegar sus propios tokens, etc.
- Se utiliza para asegurarse de que los proveedores tienen una participación en el ecosistema de Chromia, compensando así los incentivos por coludirse.

Dado que los tokens Chroma se utilizan para fines de “stake” y reserva, esperamos que una cantidad significativa sea sacada de la circulación y “bloqueada” para este tipo de uso.

Cuentas del sistema

Chromia tiene varias cuentas de token Chroma especiales que se utilizan para fines de todo el sistema:

- Referencia de tokens ERC20: los tokens Chroma en esta cuenta pertenecen a los propietarios de tokens Chroma ERC20 que permiten cierta interoperabilidad con la cadena de bloques Ethereum. Esta cuenta es administrada por la cadena de bloques de la puerta de enlace Ethereum.
- Grupo de compensación de nodos del sistema: los nodos que ejecutan cadenas de bloques de dapp son compensados por dapps. Pero los nodos que ejecutan blockchains del sistema también necesitan ganar dinero. Por esta razón, un cierto porcentaje (decidido por el consejo de proveedores) se saca de las tarifas de alojamiento y se envía al grupo de compensación de nodos del sistema, que luego se usa para compensar a los nodos por las cadenas de bloques del sistema de alojamiento. En otras palabras, Chromia en sí puede ser visto como una dapp que gestiona a otras dapps.

- Grupo de desarrollo futuro: Inicialmente ChromaWay y sus subsidiarias desarrollarán Chromia, pero eventualmente este rol debería ser descentralizado. Una vez que la economía está suficientemente descentralizada, el "grupo de desarrollo futuro" se puede desbloquear y utilizar de acuerdo con el voto de los proveedores para mejorar Chromia en su conjunto.

- Parte destinada a caridad: en ciertas situaciones donde se sacrifican tokens (se explica a continuación) una fracción de dichos tokens se puede desviar a un grupo de caridad. Los fondos de esta cuenta se pueden utilizar para donar a organizaciones benéficas de acuerdo con los votos de los usuarios. Esto puede promover Chromia como una cadena de bloques ética y socialmente consciente.

Cuenta de bien público

En ciertas situaciones, los tokens deben ser "sacrificados" (irreparablemente destruidos o quemados) para evitar un conflicto de intereses o la posibilidad de abuso. Estas situaciones incluyen los mecanismos de control de Sybil, el castigo de las partes que se comportan mal o una “acción neutral" en caso de desacuerdo entre dos o más partes.

Chromia ofrece una alternativa a la destrucción irreparable en forma de cuenta de bien público. Esta es una cuenta virtual que distribuye automáticamente los tokens recibidos en 4 cuentas diferentes:

- El 25% de los tokens se queman; la grabación de tokens beneficia indirectamente a todos los titulares de tokens de Chroma, ya que los tokens se eliminan permanentemente de la circulación
- El 25% de los tokens se colocan en el "Grupo de compensación de nodos del sistema"
- El 25% de los tokens se colocan en el "Grupo de desarrollo futuro"
- El 25% de los tokens se colocan en la parte destinada a caridad

Por lo tanto, todos los usuarios de Chromia se benefician indirectamente de la cuenta de bien público a largo plazo. Es muy poco probable que se abuse de la cuenta de bien público, ya que los fondos de control en ella son colectivamente coinvertidos y no son fácilmente accesibles. Por lo tanto, es una alternativa viable y productiva a la simple “quema".

Los fondos se enviarán a la cuenta de bien público en los siguientes casos:

- Un usuario tendría que enviar 10 tokens Chroma a la cuenta de bien público para convertirse en un "ciudadano de Chromia". Esto confirma el compromiso del usuario con Chromia y da ciertas ventajas como la capacidad de votar, servicios prioritarios, la capacidad de participar en el programa de recompensas "solicitud de suerte", etc. (Los detalles sobre este programa está cubiertos en el Apéndice.)
- La parte del stake perdida por los nodos que se comportan mal se envía a la cuenta de bien público.
- El 0,1% de las tarifas de alojamiento de aplicaciones se envían a la cuenta de bien público Además, alentamos a las dapps a usar la cuenta de bien público cuando el destino de los tokens no esté claro por alguna razón, o si los tokens deben destruirse por razones teóricas del juego. Por ejemplo, "Burnable Payments" es un simple mecanismo teórico del juego que asegura que ni el comprador ni el vendedor tengan un incentivo para hacer trampa: si el comprador no está de acuerdo con el vendedor, puede quemar los fondos.

Distribución de tokens

Mil millones de tokens se crearán en el momento del lanzamiento del sistema. Eso constituye el límite de suministro de tokens, no se creará ningún token en el futuro. Distribución inicial de tokens:

- 70% propiedad de ChromaWay a través de su filial Chromia Devcenter Oü para ser vendido,

adjudicado a los miembros del equipo, invertido o utilizado de cualquier otra manera
- 3% puesto en un contrato de conversión automática en Ethereum blockchain para habilitar la conversión Chroma<->ETH
- 2% puesto en el grupo de compensación de nodo del sistema
- El 25% se destina a uso promocional: se dará a los usuarios para que prueben aplicaciones alojadas en Chromia

Dentro de la asignación de ChromaWay, hasta el 25% (de todos los tokens) se venderá inicialmente a socios seleccionados. El resto se bloqueará y se liberará lentamente con el tiempo. Hasta un 17% se desbloqueará durante el primer oído y después del lanzamiento, luego hasta un 12% por año. ChromaWay y sus subsidiarias mantendrán tokens durante al menos tres años. Esto crea incentivos a largo plazo para el desarrollo de Chromia. Después de tres años, el desarrollo y la gobernanza de Chromia deben pasar a un modelo descentralizado.

Los tokens promocionales también se bloquearán inicialmente y se desbloquearán a una tasa del 0.5% por mes. Por lo tanto, el porcentaje de tokens en circulación cambia con el tiempo:

1. Al inicio: hasta un 30%
2. Después de 1 año: hasta el 53%
3. Después de 2 años: hasta un 71%
4. Después de 3 años: hasta el 89%
5. Después de 4,5 años: 100%

Fondo de token promocional

El uso del fondo de token promocional será controlado inicialmente por Chromia Devcenter. Su propósito es fomentar el uso de la plataforma Chromia y dapps alojadas en Chromia. Los tokens de este fondo deben ser solo para los usuarios finales, no deben usarse para financiar el desarrollo de proyectos.

La razón de este fondo es que es difícil para un usuario promedio de Internet adquirir tokens: necesitan registrarse en intercambios criptográficos, lo que es una gran molestia. También las personas son generalmente reacias a gastar dinero sólo para probar una nueva aplicación (que podría no ser tan grande).

Por lo tanto, es necesario regalar tokens de forma gratuita para construir una base de usuarios convencionales. Sin embargo, esto debe hacerse con precaución. Obviamente, se deben usar las medidas de control de Sybil: es ciertamente posible que alguien intente hacerse pasar por varios usuarios para adquirir un gran número de tokens de forma gratuita. Una forma posible de mitigar el abuso es requerir algún tipo de identificación (por ejemplo, cuenta de Facebook).

Los tokens también se pueden dar para su uso dentro de una aplicación o juego específico. Los tokens del fondo promocional se liberan gradualmente para:

- Usuarios incorporados a medida que el sistema crece
- Supervisar la situación y experimentar con diferentes formas de distribuir tokens
- Evitar afectar el valor del token Chroma

1% por mes es una tasa *de* distribución máxima. Si el uso promocional se considera ineficiente, los tokens podrían reservarse para su uso posterior o enviarse a la cuenta de bien público.

Descentralización

Centralización necesaria al principio

Chromia será una verdadera plataforma descentralizada para aplicaciones descentralizadas: no controlada por nadie, abierta a la innovación sin permiso.

La descentralización no es un punto de partida, sino un objetivo. La descentralización adecuada requiere

una comunidad fuerte con un número largo de participantes independientes que están comprometidos con Chromia. Pero construir una comunidad lleva tiempo. La plataforma necesita probarse a sí misma antes de que se considere lo suficientemente interesante como para contribuir a ella.

Por lo tanto, Chromia será centralizada al principio; creemos que es mejor adoptar esto y usar un modelo centralizado de desarrollo y gobernanza para acelerar el desarrollo.

Por esta razón, ChromaWay abrió una empresa con fines de lucro llamada Chromia Devcenter Oü que actuará como centro de desarrollo de Chromia en las etapas iniciales. Como el mayor poseedor de tokens Chroma que están bloqueados durante más de 3 años, Chromia Decentar está motivada en aumentar el valor de Chromia como sistema, ya que probablemente también aumentará el valor de sus activos.

Después de observar el ecosistema de criptomonedas durante 7 años, creemos que un modelo con fines de lucro es la forma óptima de escalar el desarrollo en las etapas iniciales. Estos son algunos ejemplos de fallas de modelos basados en la comunidad más descentralizados:

- El proyecto "Colored coins" sufrió de un lento desarrollo y fragmentación. Incluso las recompensas monetarias no ayudaron a atraer a una base de desarrolladores persistentes [20]. Desarrolladores que se unieron al proyecto produciendo temporalmente código de baja calidad y luego fueron a otra cosa.
- El proceso impulsado por recompensas del proyecto Master Coín (ahora conocido como Omni) produjo tres implementaciones incompatibles. Con el tiempo cambiaron a un proceso centralizado y lograron mejores resultados.
- La Fundación Ethereum no pudo crear una billetera que funcionase adecuadamente durante tres años. Como resultado, los usuarios tuvieron que confiar en billeteras web inseguras, o luchar con mantener su billetera de nodo completo sincronizado.

Como empresa con fines de lucro, Chromia Devcenter podrá establecer objetivos concretos y centrarse en ellos; en particular, concentrarse en las características que son esenciales para la adopción de la plataforma Chromia y el crecimiento de la base de usuarios.

Más allá del desarrollo, Chromia Devcenter también puede

- Organizar eventos promocionales
- Ayudar a las empresas a crear dapps en Chromia
- Colaborar en proyectos con otras empresas
- Invertir en el ecosistema dapp

Creemos que estas actividades se realizan mejor sobre una base comercial con fines de lucro. Los modelos de fundaciones sin fines de lucro pueden resultar en un uso ineficiente de los fondos, abuso, corrupción, etc.

Es importante destacar que Chromia Devcenter **no** es Chromia. Una vez lanzada, Chromia como red tendrá un cierto grado de autonomía. Chromia Devcenter no puede obligar a las personas a ejecutar una versión particular del software. Tampoco puede modificar ningún registro de cadena de bloques o estado más allá de lo que se le concedió acceso explícitamente. Por lo tanto, no se le puede hacer responsable de lo que sucede dentro de la red.

---

[20] Ver entrevista con ChromaWay CTO, en ese momento liderando el proyecto de monedas de colores, en Coindexo en 2013| https://www.coindesk.com/colored-coins-paint-sophisticated-future-for-bitcoin/

Con respecto a la red Chromia, el papel de Chromia Devcenter es el siguiente:

- Producir software libre y de código abierto, que se puede inspeccionar y modificar de forma independiente según sea necesario.

- Controle ciertos parámetros, como el precio de los recursos y la selección de proveedores, hasta que la red sea lo suficientemente grande y descentralizada como para controlar estos parámetros por sí sola.

Hay dos riesgos asociados con este rol:

- El software de código abierto (u otro software relevante) tendrá una puerta trasera u otra amenaza de seguridad.

Mitigación: Alentamos a los proveedores y usuarios a revisar el software antes de ejecutarlo.

- Los parámetros del sistema o la selección del proveedor se pueden establecer en valores que interrumpan el sistema. *Mitigación*: Limitaremos la tasa de cambio a través de las reglas de blockchain aplicadas por los nodos. En el peor de los casos, los proveedores/usuarios pueden bifurcar la red para evitar configuraciones disruptivas.

Descentralización a través de un conjunto diverso de proveedores

Una vez que el ecosistema de proveedores está lo suficientemente maduro, la gobernanza puede pasar a un grupo de proveedores. ¿Cómo se compara esto con la calidad de la descentralización vista en otras cadenas de bloques?

Bitcoin

Satoshi originalmente describió Bitcoin como "1 CPU = 1 voto" tipo de sistema. La base de usuarios original consistía principalmente en usuarios comunes de Internet interesados en sistemas P2P, y la producción de bloques era extremadamente descentralizada. Aun así, Satoshi era esencialmente el dictador y podía cambiar el código como quisiera. Podría, en principio, hacer una actualización que robaría monedas de otros usuarios.

Con el tiempo, la situación con las actualizaciones de código se volvió mejor: se revisa todo el código que entra en el software del nodo Bitcoin, los binarios del nodo Bitcoin se construyen utilizando un proceso que permite a varias partes verificar que el código en el repositorio corresponde a los binarios, esto significa que los usuarios finales pueden confiar en un grupo descentralizado de desarrolladores para controlar las posibles puertas traseras y otros problemas.

Por otro lado, la situación con la producción de bloques empeoró con el tiempo. En primer lugar, los usuarios se unieron a los "grupos de minería" para hacer que las recompensas sean más predecibles. Como resultado, ya no producen bloques, sino que alquilan su poder a una piscina que produce bloques. Esto significa que un grupo de minería puede, en principio, producir una cadena maliciosa de bloques. En teoría, los usuarios deben notar esto y cambiar a un grupo diferente, pero tomará algún tiempo. En un momento determinado, un solo grupo (GHASH.io) tenía >50% del hashpower total, y los usuarios no hicieron nada.

Otro problema vino con el advenimiento de la minería ASIC: las empresas manufactureras de ASIC comenzaron a explotar por su cuenta. Las empresas que tenían chips más eficientes obtuvieron mayores ganancias y pudieron reinvertirlos en la expansión. Las economías de escala crean un bucle de retroalimentación positiva donde la producción de chips de minería y la minería en sí se vuelve cada vez más centralizada.

Esto culminó con Bitmain enviando más del 70% de todo el equipo de minería, y los grupos de minería afiliados a Bitmain que tenían más del 50% del hashrate total. Bitmain no reporta ninguna estadística, pero tenemos todas las razones para creer que los almacenes con el logotipo de Bitmain en ellos llenos de mineros Bitmain en realidad pertenecen a Bitmain y son la fuente de un enorme hashrate. En cualquier caso, los tres grupos mineros más grandes de la actualidad pueden controlar la red, y dos de ellos están

afiliados a Bitmain.

También es innegable que la mayor parte de la potencia de hash se aloja dentro de China, gracias a la energía barata, instalaciones baratas, etc. Esto le permite al gobierno chino controlar Bitcoin. Potencialmente podría apoderarse de las instalaciones y ejecutar un ataque del 51%, o un tenedor suave para introducir la censura.

La centralización de PoW resultó en actualizaciones de red retrasadas y Bitcoin se convirtió en prácticamente inusable para los pagos debido a tarifas extremadamente altas. Resumen: mientras que el desarrollo de Bitcoin está descentralizado, la producción de bloques está fuertemente centralizada.

DPoS

Se observó que las cadenas de bloques basadas en DPoS - BitShares, Lisk, ARK, STEEM, EOS - tienen un gran grado de centralización de participación, lo que significa que pocos grandes titulares de tokens pueden controlar eficazmente la red. Los problemas con la centralización de DPoS están completamente explicados por Vitalik Buterin[21].

Ethereum

La producción de bloques de Ethereum está actualmente basada en PoW y, por lo tanto, tiene aproximadamente los mismos problemas que Bitcoin (los tres grupos más grandes pueden controlar la producción de bloques).

Está destinado a eventualmente hacer la transición a la prueba de participación o “proof-of-stake”. Eso no significa que cada parte interesada tenga la oportunidad de producir un bloque. En su lugar, el número de productores de bloques se restringirá a unas 1000 entidades, por lo que los propietarios de tokens más pequeños tienen que delégate producción de bloques a los grupos para poder participar.

Chromia

Parece que ningún proyecto existente da el control de la red a un conjunto muy grande de personas. Tampoco parece ser un enfoque particularmente útil, la mayoría de las personas no tienen suficiente conocimiento técnico ni motivación para mantener la red segura. Una persona que ejecuta software sin inspeccionarlo cuidadosamente es esencialmente solo un proxy para la entidad que decidió qué software lanzar.

Por esta razón creemos que el modelo de Chromia donde la red está controlada por un grupo limitado de proveedores no es un impedimento para la descentralización. Mientras estos proveedores sean verdaderamente independientes, persigan sus propios objetivos (es decir, se beneficien de las dapps de alojamiento) y operen en muchos países diferentes, el sistema puede considerarse descentralizado.

Inicialmente planeamos obtener al menos doce proveedores. A largo plazo, el número puede llegar a miles, a la par con el esquema propuesto para Ethereum.

Otra forma de verlo es una barrera de entrada. Un minero Bitcoin ASIC se puede comprar por varios miles de dólares, pero un usuario no será capaz de generar ningún bloque por su cuenta. Para convertirse en un jugador importante se necesitan cientos de millones de dólares en capitalización para adquirir hardware y construir instalaciones.

Por otro lado, cualquier empresa de alojamiento profesional puede convertirse en un proveedor de Chromia y participar en la creación de bloques. Por lo tanto, creemos que la barrera de entrada es en realidad más baja que la observada en otras cadenas de bloques.

Número de nodos completos

Las cadenas de bloques públicas como Bitcoin y Ethereum cuentan con un gran número de nodos completos, que se estima que están en el rango de 5000-10000. Si bien un gran número de nodos completos aumenta potencialmente la resistencia de la red, también tiene desventajas: la red no puede ser más rápida que su nodo más lento. Por lo tanto, ambos.

Bitcoin y Ethereum limitan severamente el número de transacciones, así como los recursos computacionales necesarios para procesar las transacciones.

Chromia toma un enfoque diferente: el número de nodos completos podría estar limitado al número de productores de bloques, que normalmente estará en la escala de 10-100 nodos por blockchain. ¿Esto da como resultado una menor resistencia de la red? Analicemos diferentes amenazas:

1. **Errores de hardware de nodo:** suponiendo que los errores sean aleatorios, es extremadamente improbable que 10 nodos fallen al mismo tiempo, antes de que se puedan realizar nuevas réplicas.
2. **Red DoS**: Mientras que en ciertos escenarios un mayor número de nodos es útil, una red se puede desactivar de manera efectiva dirigiéndose específicamente a los productores de bloques, y el número de productores de nodos independientes podría ser en realidad mayor en caso de chromia.
3. **Particiones de red**: Las redes basadas en el consenso de PoW normalmente no hacen nada para detectar particiones de red, por lo que simplemente pueden trabajar a través de interrupciones menores. Pero en caso de una interrupción importante, podría resultar en gastos dobles en los diferentes lados de la partición. El hecho de que una red de estilo Chromia se detenga en caso de una partición es en realidad una característica, no un error.

Cabe señalar que Chromia no desalienta a los usuarios de ejecutar nodos completos. Cada cadena de bloques que se ejecuta en Chromia debe ser pública, por lo tanto, cualquier usar que desee ejecutar un nodo de observación completo para una cadena de bloques en particular debe ser capaz de hacerlo, siempre y cuando tenga acceso al hardware moderno.

De hecho, si comparamos Chromia con Ethereum, se puede decir que la arquitectura Chromia facilita la ejecución de un nodo completo: un usuario de Ethereum se ve obligado a descargar datos de todas las dapps y todos los usuarios. El usuario de Chromia puede elegir qué dapps le interesan y sincronizar solo los datos de blockchain correspondientes.

El número de nodos completos de Chromia podría ser menor no porque sea más difícil ejecutar un nodo, sino porque con un cliente ligero que funcione correctamente no es necesario, y esperamos que menos aficionados se preocupen más por las dapps específicas que por “la computadora del mundo".

Seguridad

Cadena de bloques

El papel de la cadena de bloques es asegurarse de que hay un solo estado de aplicación visto por todos los usuarios, y que los ataques de doble gasto y repetición no son posibles.

En un modelo de seguridad de cliente ligero, los nodos de cadena de bloques también asumen la responsabilidad de validar las transiciones de estado. Discutiremos la seguridad del cliente en una sección separada; en esta sección nos centraremos únicamente en los aspectos de seguridad del modelo de “nodo completo".

---

21 https://vitalik.ca/general/2018/03/28/plutocracy.html

La amenaza más básica contra la que estamos protegiendo es un solo nodo que viola deliberadamente las reglas del sistema. Esto puede ocurrir porque quien lo controla se ha corrompido por alguna razón[22], o porque ha sido comprometido por un atacante externo. Los sistemas centralizados construidos usando una arquitectura de software tradicional no tienen protección contra eso - un solo servidor comprometido puede resultar en la modificación arbitraria de datos, que en caso de datos financieros puede conducir a pérdidas arbitrarias. En particular, esto podría afectar a los siguientes escenarios:

- Intrusión externa mediante la explotación de una vulnerabilidad de software o hardware.
- Empleado no autorizado: el administrador del sistema u otra persona que tenga acceso al servidor puede explotarlo para beneficio personal.
- Manipulación del proveedor de hospedaje: el acceso físico al servidor permite al proveedor modificar los datos.
- La propia empresa puede cambiar arbitrariamente los datos o las reglas para su propio beneficio.

La primera capa de protección contra estos escenarios es la lógica de aplicación que requiere autorización criptográfica y un modelo de cálculo determinista. Cuando los nodos de un usuario tienen datos completos, pueden detectar casos en los que se infringen las reglas y, por lo tanto, rechazar un estado de aplicación falso. En Chromia esto se logra al requerir que las aplicaciones se desarrollen en Rell: Rell tiene un modelo de computación determinista y facilita la implementación de la autorización criptográfica para todas las mutaciones de datos. La arquitectura general también permite a los nodos de usuario recibir datos de entrada completos (bloques y transacciones) y calcular de forma independiente el estado de la aplicación.

Un atacante más sofisticado puede aprovechar situaciones en las que pueden existir varios estados de aplicación válidos al mismo tiempo. Este ataque se suele describir como doble gasto, por ejemplo:

1. Un atacante produce un estado de aplicación en el que un comerciante sea pagado para enviar algunos bienes.
2. El mercante envía las mercancías.
3. El atacante reemplaza el estado de la aplicación por otro en el que no se paga al comerciante, sino que los fondos se dirigen de vuelta a la cuenta del atacante.
4. Ahora el atacante tiene tanto los bienes como el dinero.

Existen muchas variaciones de este ataque. Por ejemplo, se puede hacer utilizando diferentes tipos de tokens y el comerciante puede ser un intercambio. Para protegerse contra este ataque, no se permite que existan estados de aplicación mutuamente incompatibles con el sistema.

Esto se puede hacer utilizando un algoritmo de consenso bizantino tolerante a errores (BFT) que "confirma" un solo estado de aplicación y rechaza todos los estados incompatibles después de eso.

Se ha demostrado que en una red asíncrona (es decir, sin confirmación de entrega de paquetes) un algoritmo de consenso BFT puede tolerar hasta el 33% de los fallos de nodo. Estrictamente hablando, 2/3 además de un nodo debe seguir siendo honesto. Por ejemplo, un sistema con 10 nodos puede tolerar hasta 3 fallos, es decir, seguirá funcionando cuando 3 nodos se vean comprometidos.

Chromia utiliza un algoritmo de consenso de estilo PBFT para construir el blockchain. Cuando el número de nodos validadores de blockchain es 3f + 1, un bloque debe recibir 2f + 1 "votos" para ser confirmado (es decir, más de 2/ 3 de todos los votos). Los nodos de los usuarios solo tratan con bloques que están confirmados.

Por lo tanto, Chromia puede tolerar la corrupción arbitraria de una minoría (menos de 1/3) de los nodos validadores de blockchain sin consecuencias drásticas, excepto una posible desaceleración. Chromia intentará

garantizar que a cualquier cadena de bloques se le asignen nodos de diferentes proveedores para que un solo error no pueda resultar en la corrupción de la cadena de bloques. Los requisitos serán especialmente estrictos para las cadenas del sistema.

Esta es la suposición fundamental de Chromia - nodos individuales (así como proveedores individuales) pueden y fallarán, pero no debería tener ningún efecto en los usuarios de Chromia.

Pero también tenemos que considerar situaciones en las que más del 33% de los validadores de una cadena de bloques en particular fallan. Consideramos que esto es poco probable, pero posible. Si bien no podemos garantizar un funcionamiento sin problemas en caso de un "ataque del 34%", podemos tratar de minimizar el daño y permitir una recuperación rápida. En particular, Chromia necesita características para:

- Dificultar que los atacantes se beneficien del ataque.
- Hacer posible que Chromia detecte el ataque lo antes posible, para que se puedan tomar medidas de recuperación.
- Que sea posible para los usuarios de Chromia para detectar el ataque tan pronto como sea posible, por lo que pueden abstenerse de las operaciones que podrían resultar en pérdidas financieras.
- Permita a los usuarios de Chromia esperar confirmaciones más fuertes para transacciones de alto valor si lo desean.

---

[22] la corrupción aquí abarca una serie de escenarios posibles en los que el proveedor del nodo tiene incentivos para actuar de una manera que es perjudicial para los objetivos del colectivo en el que participa. Ganancia financiera, coerción, engaño, inestabilidad mental; hay muchas razones por las que un operador de nodo puede

La herramienta más poderosa a nuestra disposición es el anclaje, una forma de aumentar la fuerza de confirmación de una cadena de bloques usando otra. Consideremos el esquema de anclaje más simple. Supongamos que queremos anclar bloques de blockchain X en blockchain Y. Para hacer eso:

1. Cuando se confirma **la X_i** de bloques en blockchain X, uno de los productores de bloques publicará una tupla **(X, i**, **hash(X_i))** en blockchain Y
2. Una vez que esa publicación se confirma en blockchain Y, el nodo de un usuario (que sigue tanto blockchain X como blockchain Y) puede encontrar la primera tupla de forma **(X, i, *)** que se publica en blockchain Y
3. Se dice que X_i de bloques está anclado cuando **(X, i**, **hash(X_i))** es la primera tupla de este tipo.
4. Si el consenso sobre blockchain X falla y se produce un bloque X diferente **_i,** el bloque anclado **X_i** debe tener prioridad. Es decir, en caso de una recuperación, la cadena de bloques debe incluir el último bloque anclado, y los bloques incompatibles con él deben eliminarse.

Es fácil ver cómo el comerciante puede usar el anclaje para aumentar la fuerza de confirmación. Supongamos que el comerciante esperará hasta que el bloque **X_i** que contiene un pago a él se confirme y ancle antes de que envíe los bienes. En este caso si el consenso de blockchain X falla (por ejemplo, los nodos de X se ven comprometidos y producen varios historiales incompatibles), pero blockchain Y se mantiene correcto, el comerciante no sufre una pérdida: una vez que se reinicie blockchain X (por ejemplo, con nuevos validadores) se incluirá el bloque X_i y, por lo tanto, el comerciante recibirá el dinero.

Las implementaciones técnicas de anclaje pueden diferir en:

1. Lo que se está publicando (por ejemplo, sólo un compromiso)
2. Quién puede publicar información

3. Si las pruebas de cliente son posibles
4. Lo fácil que es para un nodo detectar un error de anclaje

Chromia hará uso de anclaje multinivel, es decir, los bloques de una cadena de bloques dapp se anclarán en una cadena de anclaje especial mantenida por otro conjunto de nodos.

Veamos un ejemplo. En primer lugar consideramos la situación sin anclaje. Supongamos que la blockchain A de la dappes ejecutada por 10 nodos validadores, todos los cuales están comprometidos. Si los tokens de esta dappchain se negocian en un intercambio centralizado, los nodos comprometidos podrían be utilizados para realizar un ataque:

1. Los nodos prepararán dos versiones de un bloque a la misma altura: el bloque **X_i** contiene un pago del atacante al intercambio y el bloque **X'_i** no
2. El intercambio ve bloquear **X_i** y acredita tokens a la cuenta del atacante.
3. El atacante vende sus tokens por bitcoins y retira bitcoins del intercambio.
4. El _i bloque **X** se revela a todos los demás nodos y los bloques posteriores se construyen sobre él.
5. No es posible saber si el **X_i** de bloques o _i bloque **X** debe tener prioridad. Obviamente, el **X_i** de bloque es mejor para el intercambio, pero el _i bloque X podría incluir otros pagos importantes.
6. Por lo tanto, el intercambio podría sufrir una pérdida incluso después de que se reemplacen los nodos defectuosos, ya que la cadena de bloques podría construirse en el bloque **X'_i.**

En una situación con anclaje, el intercambio puede protegerse de este riesgo. Debe esperar hasta que el pago esté anclado en bloque X**_i** antes de crear el dinero. En este caso, incluso si los nodos intentan construir un bloque alternativo **X'_i,** ese bloque no se incluirá en una cadena de bloques después de que se reemplacen los nodos, ya que no está anclado.

El comerciante puede sufrir una pérdida sólo cuando la cadena de anclaje en sí se ve comprometida. Sin embargo, la cadena de anclaje Chromia incluirá un mayor número de nodos validadores de diferentes proveedores, por ejemplo, cien. Podría ser suficiente comprometer 4 nodos para compromiso de la cadena de bloques dapp, sin embargo, se necesitará al menos 34 nodos / proveedores para ser comprometidos para anclar dos bloques incompatibles. Es decir, requiere una colusión a gran escala.

Sin embargo, no podemos descartar completamente esta situación. Por esta razón, la cadena de anclaje se anclará en las cadenas de bloques PoW: Bitcoin y Ethereum. Una billetera de intercambio en modo de alta seguridad puede esperar hasta **que el X_i** de bloque esté anclado en el bloque **A_j,** y el bloque **A_j** esté anclado en el B_k de bloques de Bitcoin. En este caso, para revertir un pago, se necesitaría comprometer un número significativo de daños

Nodos de croma y, además de eso, realizar una reorganización de la cadena de bloques de Bitcoin. Creemos que esta situación que estapcionalmente improbable.

La fuerza de confirmación se puede aumentar aún más mediante el anclaje a múltiples cadenas de bloques. En particular, consideramos establecer una red de notarios e instituciones de gran reputación en múltiples países. Si anclamos la cadena de anclaje de Chromia en esta cadena de notarios, será imposible revertir los bloques de Chromia sin una conspiración mundial.

Seguridad de nodo

Creemos que una colusión entre los proveedores de Chromia es poco probable, ya que la pérdida de participación, ganancias y posibles acciones legales sirve como un elemento disuasorio. Sin embargo, si se descubriera un exploit de software que permite a un atacante ejecutar código arbitrario en un nodo de Chromia, varios nodos de Chromia podrían verse comprometidos al mismo tiempo.

Las causas más comunes de vulnerabilidades explotables de forma remota son los errores de corrupción de memoria dentro del código de la aplicación. Por esta razón, Chromia se implementa utilizando un lenguaje

seguro que protege contra la corrupción de memoria (Kotlin) y se ejecuta en la JVM que proporciona seguridad de memoria.

Otra posible fuente de vulnerabilidades es el código dapp. Las dapps de Chromia se implementarán en Rell, que es en sí mismo un lenguaje seguro para la memoria; además de eso, el entorno de ejecución de Rell se implementa en Kotlin y se ejecuta dentro de la JVM, por lo que para que una aplicación salga del sandbox tendrá que derrotar los mecanismos de seguridad de Rell y JVM, lo que creemos que es prácticamente imposible.

La fuente restante de vulnerabilidad es el código escrito en C, es decir, el sistema operativo host (por ejemplo, Linux) y DBMS (por ejemplo, PostgreSQL). Las vulnerabilidades explotables en el propio kernel de Linux parecen ser extremadamente raras, y el acceso a PostgreSQL será mediado por Rell, lo que limita la posibilidad de ataques.

Sin embargo, investigaremos más opciones para reducir la superficie de ataque posible:

- Ejecutar la distribución de Linux orientada a la seguridad con todos los componentes no esenciales deshabilitados
- Considere el uso de un sistema operativo que reduzca aún más la huella o la superficie de ataque (por ejemplo, OSv[23])
- Considere la posibilidad de cambiar a un motor de base de datos basado en JVM o implementar un nuevo motor de base de datos específicamente para Chromia

Otro posible vector de ataque es el hardware y el firmware. Por ejemplo, el Motor de administración Intel está presente en la gran mayoría de los productos de Intel, y efectivamente runs un runas sistema operativo separado que potencialmente puede verse comprometido. Esto podría proporcionar un vector para poner en peligro el nodo que se ejecuta en la misma CPU. Para mitigar este vector de ataque, recomendaremos a los proveedores que diversifiquen y usen hardware de diferentes proveedores. También aconsejaremos a los proveedores que limiten su exposición a los proveedores de nube. Si la mayor parte de los nodos de Chromia se ejecutan en, por ejemplo, AWS, Amazon tiene la potencia de bifurcar o apagar la red.

Seguridad de gobierno

La gobernanza puede ser una fuente de problemas de seguridad. Podemos tomar un ejemplo del mundo corporativo: mientras que la propia CEO podría no ser capaz de manipular los servidores directamente, puede reemplazar al administrador del sistema por uno que, por ejemplo, eliminará algunos datos cruciales.

Por lo tanto, los mecanismos de gobernanza de Chromia también deben diseñarse teniendo en cuenta la seguridad. En particular:

1. No debería ser posible introducir cambios que se puedan usar para bifurcar o destruir teniendo en.
2. Todos los cambios deben aplicarse con retraso para que puedan ser revisados y, si es necesario, se puedan tomar medidas de mitigación, en los casos más graves esto podría ser una bifurcación dura de emergencia.
3. La tasa de cambios debe ser limitada.

Seguridad ligera del cliente

La mayoría de los usuarios de Chromia utilizarán clientes ligeros que no procesan la totalidad de los datos de blockchain. Tendrán que confiar en los nodos de Chromia para consultar datos del estado de la cadena de bloques y proporcionar el estado de confirmación de las transacciones y los pagos. ¿Cómo pueden los usuarios del cliente ligero autenticar estos datos? En resumen, tienen que confiar en los nodos validadores. Cada bloque está firmado por una mayoría de BFT [24] de todos los nodos de validación. Por lo tanto, para confirmar una transacción no válida, más de dos tercios de los nodos de validación tendrían que estar en peligro.

La seguridad del cliente ligero no es significativamente peor que la seguridad del nodo completo. Poco más de un tercio de los nodos deben verse comprometidos para producir una bifurcación, pero más de dos tercios deben verse comprometidos para producir un bloque no válido. El primer escenario es mucho más parecido a ley, y los clientes ligeros están protegidos contra él en la misma medida que los nodos completos.

Los clientes ligeros también pueden aprovechar el anclaje, incluido el anclaje en blockchains PoW. Los métodos de anclaje utilizados en Chromia pueden producir pruebas compactas, esto significa que un usuario puede beneficiarse del anclaje sin necesidad de ejecutar un nodo completo.

En algunos casos, los datos recuperados de los nodos no son importantes y no es necesario autenticarlos. En escenarios donde es necesario autenticar los datos, se pueden utilizar diferentes estructuras de datos en función de la naturaleza de los datos:

1. Transacción Merkle tree: se puede utilizar para comprobar que una transacción está confirmada y es válida. Por ejemplo, esto se puede utilizar para verificar un pago. La raíz del árbol de Merkle de la transacción está presente en el encabezado del bloque y está firmada por los nodos como parte del algoritmo de consenso.
2. Árbol de Merkle de compromiso de estado: Normalmente, un encabezado de bloque se comprometerá con el conjunto de filas que representan el estado de la cadena de bloques. Esto permite a un cliente ligero asegurarse de que una determinada fila devuelta en respuesta a una consulta está presente en el último estado de la cadena de bloques.

Los compromisos se pueden desactivar en las cadenas de bloques de alto rendimiento a medida que aumentan la sobrecarga de la cadena de bloques. Una raíz de Merkle de compromiso de estado está presente en el encabezado de bloque y, por lo tanto, se firma de la misma manera que la raíz de Merkle de transacción.

1. Aserciones e indexadores: se pueden usar estructuras de datos especiales para demostrar que toda la respuesta de la consulta es correcta y no omite ningún dato. Si están presentes, se firman de la misma manera en el encabezado del bloque.
2. Respuestas de consulta firmadas: cuando una respuesta de consulta es importante pero no se puede probar a través de indexadores, un cliente ligero puede enviar solicitudes a varios nodos y recibir respuestas firmadas.

Un cliente ligero puede autenticar datos a través de firmas de nodo validador solo en él conoce las claves públicas del nodo validador. Las pubkeys del nodo validador se pueden obtener de la cadena de directorios. La propia cadena de directorios se puede validar utilizando la cadena raíz. El proceso de resolución es el siguiente:

1. Un cliente ligero viene con un hash incorporado y codificado del bloque de génesis de la cadena de bloques raíz, así como una lista inicial de claves públicas de nodo raíz.
2. Un cliente ligero descarga toda la cadena de bloques raíz para obtener una lista actualizada de nodos raíz. La cadena de raíces es extremadamente escasa con solo un bloque por día, y por lo tanto esta operación no es una gran carga incluso para un cliente ligero.
3. Un cliente ligero puede consultar cualquier réplica de la cadena de directorios para recuperar una lista de validadores para la cadena de bloques que le interesa y validarlos con un mecanismo de compromiso de estado, es decir, comprobar las firmas de los nodos raíz.

[23] http://osv.io/

4. Los resultados de la consulta a la cadena de directorios también deben confirmarse a través de la cadena de anclaje y el anclaje PoW.[25]

Seguridad del cliente y la billetera de Dapp

El equipo de Chromia desarrollará ChromaWallet, una billetera que se puede usar para contener tokens de Chroma, así como cualquier token en cualquier cadena de bloques de Chromia que siga el estándar FlexiblFlexible TokensTambién proporcionará la capacidad de interactuar con dapps utilizando una interfaz simple basada en formularios y administrar cuentas de dapp. ChromaWallet se proporcionará en formatos de aplicaciones de escritorio, móviles y web y se centrará en la integración de carteras de hardware.

En una versión futura, ChromaWallet podrá funcionar como un navegador dapp de propósito general, sandboxing dapp UI ejecución de código y ofreciendo representación de interfaz gráfica en una pila de tecnología web. El navegador dapp podrá descargar el código de interfaz de usuario de dapp desde una cadena de bloques de Chromia. Por otra parte, no podrá garantizar que este código esté libre de errores o defectos de seguridad, pero sí podrá garantizar que el código solo se pueda actualizar junto con la propia dapp y que todos los usuarios ejecuten código idéntico (es decir, el código no se puede interceptar específicamente para un usuario). Tenga en cuenta que la funcionalidad del explorador dapp no estará presente en la versión MVP.

En su lugar, las dapps que requieren una interfaz de usuario compleja, como los juegos, se pueden implementar mediante un cliente independiente entregado como una aplicación web o móvil. En este caso, la seguridad se puede controlar mediante el uso de subcuentas. El cliente dapp recibirá una clave privada de una subcuenta que pertenece a un usuario y podrá firmar la transacción en nombre del usuario. Esto significa que el usuario puede realizar acciones de juego de una manera natural, similar a cómo funcionan los juegos "normales". No habrá diálogos de confirmación molestando a los usuarios por cada acción que realice en un juego, como se ve en Ethereum MetaMask y EOS Scatter.

24 2/3+1 de nodos validadores totales

Sin embargo, las acciones que son sensibles, como una transferencia de una gran suma de tokens, pueden requerir confirmación utilizando una subcuenta diferente administrada por ChromaWallet. Esto significa que el código de una dapp maliciosa no será capaz de hacer un daño significativo. Esto también significa que las transacciones de gran valor pueden beneficiarse de la integración de billetera 2FA o hardware implementada en ChromaWallet.

[25] el paso #4 es necesario para asegurarse de que una colusión de nodos raíz no pueda comprometer ninguna otra cadena de bloques.