zkBTC: Das vollständig private, vertrauensfreie Bitcoin

TaprootFreak

www.zkcoins.com

Zusammenfassung. Eine rein zwischen Peers laufende Form privaten elektronischen Geldes, gedeckt durch Bitcoin, würde Zahlungen erlauben, ohne Beträge oder den Transaktionsgraphen preiszugeben und ohne Verwahrer. Digitale Signaturen und Zero-Knowledge-Gültigkeitsbeweise lösen einen Teil, aber der Nutzen geht verloren, wenn eine vertrauenswürdige Partei Transaktionen ordnen oder das Deckungs-Bitcoin halten muss. Wir schlagen ein System vor, das Bitcoin nur als Ordnungsschicht nutzt. zkCoins ist ein clientseitig geprüftes Transferprotokoll: jeder Zustandsübergang trägt einen rekursiven Gültigkeitsbeweis, und der einzige On-Chain-Abdruck ist ein Nullifier fester Grösse von 64 Byte, der Doppelausgaben verhindert, ohne ein globales Betragsbuch. zkBTC ist ein eins-zu-eins durch Bitcoin gedecktes Token auf diesem Protokoll. Die Reserve liegt in Tresoren, die nur entlang vorsignierter BitVM2-Betrugsnachweis-Pfade ausgebbar sind. Jeder darf in die Operator-Menge; diese Menge wächst nur; jeder neue Tresor wird von allen aktuellen Operatoren unterschrieben, sodass ein Inhaber, der sich vor dem ersten Mint (vor dem Einrichten dieses Tresors) anmeldet, der ehrliche Unterzeichner jedes Tresors ist und den eigenen Ausgang bedienen kann. Ein Gatekeeper, wenn bestimmt, darf neue Einlagen prüfen und kann nicht einfrieren, beschlagnahmen oder den Ausgang blockieren. Unter den genannten Ehrlichkeits- und Lebendigkeitsannahmen (1-aus-N beim Einrichten, mindestens ein ehrlicher lebender Challenger in jedem Widerspruchsfenster, und einwandfreie Circuit- und Graph-Kryptographie) kann kein Operator die Reserve stehlen. zkBTC ist in diesem Sinn effektiv vertrauensfrei: du bist Operator; du vertraust niemandem sonst dein Bitcoin an.

1. Einleitung

Der Internethandel siedelt noch immer über Finanzinstitute als vertrauenswürdige Dritte. Bitcoin hat dieses Vertrauen für die Abwicklung entfernt, doch jede Zahlung auf seinem Buch sendet Zahler, Empfänger und Betrag an die Welt. Vertraulichkeit ist im gewöhnlichen Handel nötig; die üblichen Mittel geben auf, was Bitcoin wertvoll macht. Eigene Privatsphäre-Ketten ersetzen Bitcoins Sicherheitsmodell durch ein neues. Verwahrte oder föderierte Wrapping-Token führen genau den Dritten wieder ein, den Bitcoin entfernt hat.

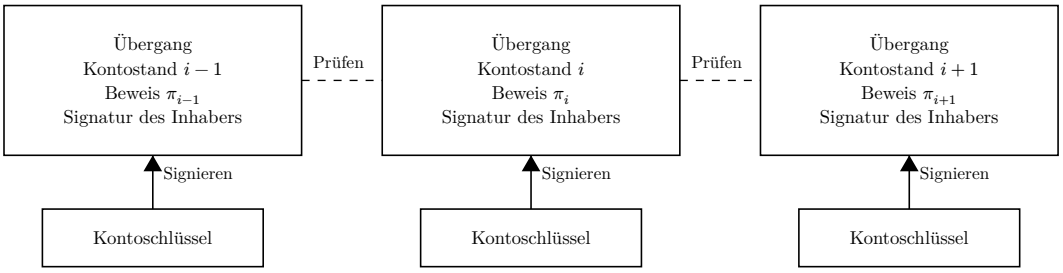
Gebraucht wird elektronisches Geld, das Bitcoin als einzige Abwicklungs- und einzige Deckungsgrundlage behält, Beträge und den Graphen kryptographisch verbirgt und jedem Inhaber den Ausgang zurück auf On-Chain-Bitcoin erlaubt, ohne dass jemand zustimmen muss. Dieses Paper schlägt ein solches System vor. zkCoins ist ein clientseitig geprüftes Transferprotokoll, dessen einziger On-Chain-Abdruck 64 Byte je Übergang ist. zkBTC ist ein durch Bitcoin gedecktes Token darauf, dessen Reserve durch Betrugsnachweise gesichert ist, nicht durch Verwahrung. Ehrliche Mehrheit der Hashrate ordnet die Nullifier. Operatoren können die Reserve nicht stehlen, solange

ein ehrlicher Operator je Deckungsgruppe seinen Epoch-Signaturschlüssel beim Einrichten löscht, mindestens ein ehrlicher lebender Challenger in jedem Fenster handelt und Circuit sowie BitVM2-Graph einwandfrei sind. Der Inhaber, der vor dem ersten Mint in die kumulative Operator-Menge eintritt, *ist* dieser ehrliche Operator für jeden Tresor. Kanonizität des Mints (ein Mint gegen eine private Gabel) ist ein eigenes Residuum; ein bestimmter Gatekeeper kann es nur zur Mint-Zeit prüfen. Ein Gatekeeper ist nicht das, was zkBTC vertrauensfrei macht.

2. Transaktionen

Wir definieren eine elektronische Münze als Kette beweis tragender Kontozustands-Übergänge. Jedes Konto ist eine Folge von Zuständen. Jeder Übergang (Senden, Empfangen, Mint, Einlösen) trägt einen Zero-Knowledge-Beweis, dass er den Regeln gegenüber seinem Vorgänger genügt, und eine BIP-340-Signatur des aktuellen Kontoschlüssels. Der Empfänger prüft den Beweis. Die Prüfung ist clientseitig: niemand sonst muss Beträge oder Parteien sehen. Der Empfänger prüft die Beweiskette so, wie ein Bitcoin-Empfänger die Signaturkette prüft.

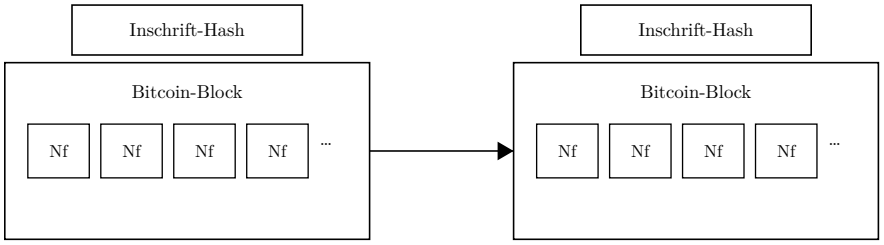
Das Problem ist, dass der Empfänger nicht sehen kann, ob der Zahler einen zweiten, widersprüchlichen Übergang desselben Zustands erzeugt hat. Eine zentrale Prägeanstalt könnte ordnen, dann hängt das ganze Geldsystem an ihr, und Privatsphäre und Ausgang fallen in Verwahrung zurück. Der Empfänger muss wissen, dass kein früherer widersprüchlicher Übergang existiert. Dafür braucht es eine einzige vereinbarte Geschichte von Übergangsmarkern, öffentlich bekannt, ohne Beträge oder Parteien. Bei zkBTC ist der erste Übergang einer Münze ein Mint, gebunden an eine On-Chain-Tresor-Einlage (Abschnitt 11), der letzte ein Einlösen, das sie verbrennt; dazwischen sind Transfers gewöhnliche zkCoins-Übergänge, und die Deckung bewegt sich nicht. Der Lebenszyklus ist also Mint, Transfer, Einlösen: nur die Endpunkte berühren die Reserve, jeder innere Schritt bleibt vom Bitcoin-Buch weg ausser seinem Nullifier.



3. Die Nullifier-Kette

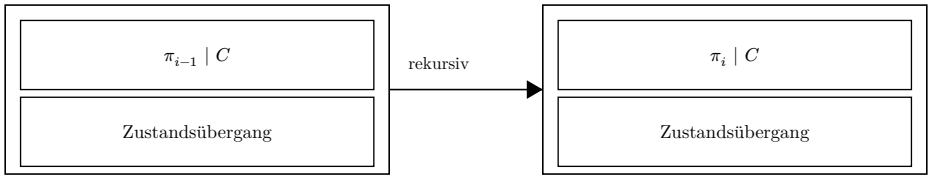
Die Lösung beginnt bei Bitcoin selbst als Ordnungsschicht. Bitcoin ist der Zeitstempelserver. Jeder zustandsfortschreibende Übergang veröffentlicht einen Nullifier fester Grösse (ein Paar (Pk_i, R_i), ein aus dem Kontoschlüssel abgeleiteter öffentlicher Schlüssel und eine Nonce-Festlegung der Signatur, 64 Byte) in Bitcoin-Blöcken über Inschriften, je Batch halbaggregiert. Der Nullifier verrät nichts über Beträge, Parteien oder den Graphen; er bedeutet nur denen etwas, die die Geschichte der Münze schon halten. Das erste Vorkommen eines gegebenen Schlüssels Pk_i auf der Kette zählt. Ein zweiter widersprüchlicher Übergang desselben Zustands trifft denselben Schlüssel und wird von jedem Prüfer abgewiesen. Die im Circuit verankerte Vorgängerbindung lässt jeden Übergang beweisen, dass der Nullifier seines Vorgängers schon on-chain ist, sodass keine Off-Chain-Gabel überlebt. Halbaggregation lässt einen Publisher viele Nullifier-Signaturen zu einer Inschrift mit

einem gemeinsamen Aggregatskalar verbinden, sodass die On-Chain-Kosten je Übergang beim 64-Byte-Paar bleiben, auch wenn ein Batch viele Marker hält.



4. Gültigkeitsbeweise

Wo Bitcoin Regeln mit Proof-of-Work durchsetzt, setzt zkCoins sie mit Zero-Knowledge-Gültigkeitsbeweisen durch. Ein Compliance-Prädikat C (ein einziger Circuit) prüft jeden Übergang: Werterhaltung je Asset (weite Summe, überlaufsicher, sodass kein Übergang Wert erzeugt ausser über einen ausdrücklichen Mint), Autorisierung durch den Kontoschlüssel, korrekter Vorgängerzustand, einmalige Verwendung jeder Münze und die Verankerung alles dessen, was der Prüfer nicht sehen kann, in verbergende Commitments. Beweise setzen sich rekursiv zusammen (proof-carrying data): wer den letzten Beweis prüft, prüft die ganze Geschichte bis zur Genesis, die Prüfkosten bleiben konstant. Eine Münze fälschen heisst, das Beweissystem zu brechen, nicht das Netz zu überrechnen. Der Circuit ist fest und sein Digest festgenagelt: alle prüfen gegen dasselbe Prädikat, wie jeder Bitcoin-Knoten dasselbe Proof-of-Work-Ziel prüft.



5. Netzwerk

Die Schritte, das Netz zu betreiben, sind:

1) Der Sender baut einen Zustandsübergang und seinen Gültigkeitsbeweis und sendet den Münzbeweis direkt an den Empfänger off-chain (gift-wrapped, NIP-17-artiger Transport).
2) Der Sender reicht seinen Nullifier an einen Publisher.
3) Publisher bündeln Nullifier und schreiben den Batch als Inschrift in einen Bitcoin-Block, gegen On-Chain-Gebühren.
4) Bitcoin ordnet die Inschriften mit Proof-of-Work.
5) Empfänger (ihre Knoten) scannen Blöcke und nehmen einen Übergang nur an, wenn das erste Vorkommen seines Nullifiers passt.
6) Wallets verlängern ihre Kontoketten auf verankerten Übergängen.

Knoten halten stets die Bitcoin-Kette mit der meisten kumulierten Arbeit für die richtige. Reorganisationen folgen Bitcoins eigener Longest-Chain-Regel: ein tiefer als die Finalitätstiefe begrabener Übergang ist gesiedelt. Erscheinen zwei Inschriften desselben Schlüssels, zählt nur

das erste Vorkommen auf der kanonischen Kette; das spätere ist eine Doppelausgabe und wird abgewiesen.

Publisher sind erlaubnisfrei und austauschbar. Jeder kann selbst veröffentlichen. Neue Nullifier-Ankündigungen vertragen verspätete Ausstrahlung: wer einen Batch verpasst, kann später gegen die Kette prüfen. Ein Nullifier muss nicht jeden Publisher erreichen; eine Aufnahme in einem bestätigten Batch genügt, und wer keinem Publisher traut, schreibt ihn selbst. Kein Publisher kann stehlen: sie berühren nie Münzen, nur 64-Byte-Marker. Einen bestimmten Nutzer zu zensieren kostet Gebühren, die ein Wettbewerber einnimmt. Fälschen liegt ausser Reichweite: die Marker tragen keinen ausgebbaren Wert, die Gültigkeit erzwingen die Beweise, die der Empfänger prüft. Gleichstände zwischen Bitcoin-Gabeln löst weiteres Proof-of-Work, wie in Bitcoin selbst; das erste Vorkommen eines Nullifiers wird immer auf der überlebenden kanonischen Kette nach Reorg bewertet.

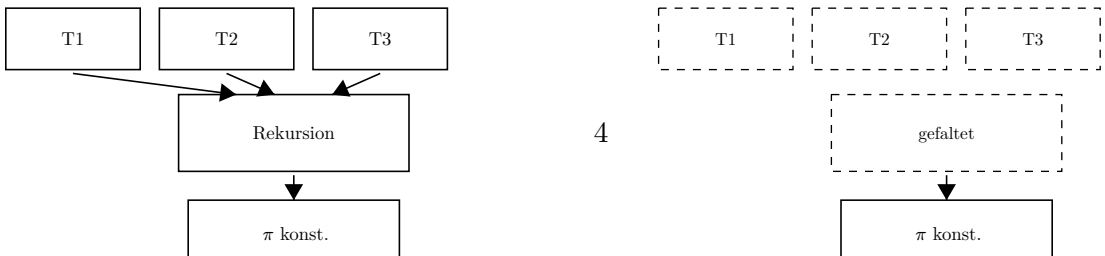
6. Anreiz

Das Protokoll definiert einen Gebührenmünz-Mechanismus, unter dem Publisher je eingeschriebenem Marker entlohnt werden. Sie legen die On-Chain-Kosten vor und werden in Gebührenmünzen an den Batch-Einträgen bezahlt. Die erste Protokollversion hält das Veröffentlichen gesponsert: jeder kann selbst veröffentlichen oder einen sponsernden Publisher nutzen. Der offene Gebührenmarkt ist ein definierter, aufgeschobener Mechanismus, kein Startmerkmal; es gibt keine neue Ausgabe für Publisher. Die Anordnung gleicht Minern, die Transaktionsgebühren einziehen, sobald die Blocksubvention sinkt.

Für die zkBTC-Reserve hinterlegen Operatoren Kautionen und verdienen Einlösegebühren (die `max_fee`-Decke des Inhabers) dafür, dass sie Bitcoin an Einlöser vorstrecken. Eine falsche Behauptung wird on-chain widerlegt, die Kaution des Operators wird gestrichen und der Anspruch ist nichtig, ein Betrüger verbrennt also seinen Einsatz für eine Forderung, die nichts zahlt. Widerspruch ist erlaubnisfrei; wie Challenger finanziert werden, ist eine wirtschaftliche Kalibrierung, die der Entwurf offenlässt. Ehrlichkeit ist die profitable Strategie. Ein Publisher gewinnt nichts durch Zurückhalten: Nutzer wechseln. Unter diesen Regeln und den genannten Annahmen (1-aus-N beim Einrichten, mindestens ein ehrlicher lebender Challenger im Fenster, einwandfreie Circuit- und Graph-Kryptographie) ist Betrug nicht profitabel. Der Anreiz ist Gebühreneinkommen für ehrlichen Dienst, nicht Ausgabe eines neuen Basisassets. Wer Kautionen anhäuft und viele Operatoren anmeldet, gewinnt keine direkte Ausgabemacht über die Tresore; Ausgaben existieren nur entlang des vorsignierten Graphen. Gewinnsuchendes Kapital ist besser damit eingesetzt, Einlösungen gegen Gebühr zu bedienen, als Kautionen auf widerlegbaren Behauptungen zu verbrennen. Die gierige Strategie ist ehrlicher Dienst, nicht abgesprochener Betrug.

7. Kompakter Zustand

Ist der letzte rekursive Beweis eines Kontos geprüft, müssen die früheren nicht behalten werden. Bitcoin beschneidet ausgegebene Transaktionen im Nachhinein; zkCoins materialisiert die globale Geschichte gar nicht. Rekursion faltet die ganze Übergangsgeschichte in einen Beweis fester Grösse. Die Kette trägt nur 64 Byte je Übergang. Knoten halten keine globale UTXO-Menge und keinen Nutzerzustand: ein nur anhängendes Log der Erstvorkommen reicht, und die Geschichte jedes Kontos faltet sich in einen konstanten rekursiven Beweis.



Eine grobe Kapazitätsgrenze folgt allein aus dem Blockraum. Mit 4-MB-Blöcken alle 10 Minuten und 64 Byte je Nullifier lässt Bitcoin wie er ist rund $4 \times 10^6/64/600 \approx 104$ Übergänge je Sekunde zu, wenn der ganze Block Nullifier wäre. Die Blockprüfung des Nullifier-Stroms ist billiger als Bitcoins eigene: keine Witness-Ausführung, kein UTXO-Lookup. Neue Knoten brauchen keinen Initial Block Download von Transaktionsdaten, nur die 64-Byte-Marker und das nur anhängende Erstvorkommen-Log. Jeder Knoten leitet dieses Log aus der Kette allein ab, Knoten können gehen und wiederkommen und die in ihrer Abwesenheit eingeschriebenen Nullifier als Beweis dessen nehmen, was geschah — es gibt sonst nichts, dem zu vertrauen wäre.

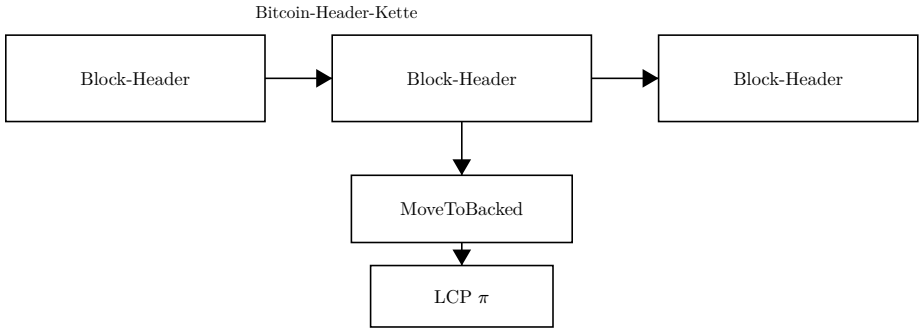
8. Vereinfachte Prüfung

Zahlungen lassen sich prüfen, ohne den vollen Beweisstapel zu betreiben. Zwei leichte Anordnungen gelten.

Eine dünne Wallet darf nur ihre Schlüssel und geheimen Blinds halten, während ein Knoten scannt und beweist. Der nächste Ausgabeschlüssel des Kontos ist über ein wallet-eigenes verbergendes Commitment gebunden, ein bösartiger Knoten kann die Schlüssel der Wallet nicht drehen. Er kann aber noch die Ausgaben wählen, die er beweist: Zahlungsabsicht in den Circuit zu binden ist dokumentierte offene Entwurfsarbeit, eine Wallet, die das Beweisen abgibt, legt darin Vertrauen in ihren Knoten.

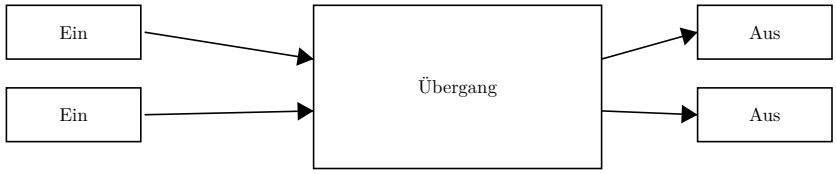
Getrennt davon bettet der Mint-Übergang einen rekursiven Bitcoin-Light-Client-Beweis (Header und Proof-of-Work-Tiefe) ein, der zeigt, dass die Deckungs-Transaktion mindestens D_{mint} Blöcke tief liegt. Das ist vereinfachte Zahlungsprüfung im Circuit. Wie Nakamoto für SPV feststellte, beweist Proof-of-Work-Tiefe Arbeit, nicht Kanonizität: eine private Gabel kann ebenfalls tief sein. Genau dieses Residual ist der Grund, warum der Mint-Pfad einen äusseren Canonical-View-Check tragen kann (der Gatekeeper aus Abschnitt 11).

Wer häufige oder hochwertige Zahlungen empfängt, wird weiter eigenen Knoten und Prover für unabhängige Prüfung vorziehen. Delegation ist eine Bequemlichkeit mit diesem Residual; beides selbst zu betreiben stellt den vollen clientseitigen Prüfpfad von Abschnitt 4 wieder her, ohne Dritte für Scan oder Beweis.



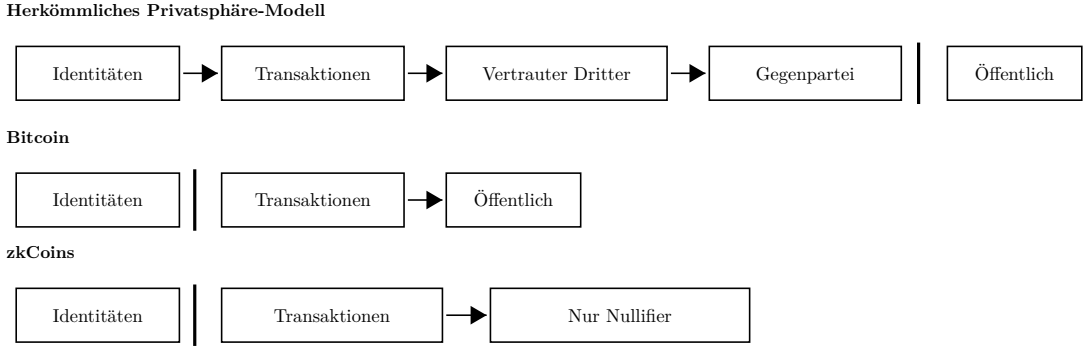
9. Zusammenfassen und Aufteilen von Wert

Münzen einzeln zu behandeln wäre möglich; Übergänge mit mehreren Eingängen und Ausgängen sind effizienter. Ein Übergang lässt bis zu acht Eingänge und acht Ausgänge zu; Erhaltung wird im Circuit erzwungen. Eine typische Zahlung nimmt eine Eingangsmünze und teilt sie in eine Zahlungs- und eine Wechselausgabe. Eine eigenständige Kopie der vollen Geschichte einer Münze braucht es nie; die Rekursion hat sie schon in den letzten Beweis gefaltet.



10. Privatsphäre

Banken schaffen Vertraulichkeit, indem sie Information bewachen: Zugang haben die Parteien und ihr Vermittler. Ein System, das Übergangsmarker öffentlich ansagen muss, muss Privatsphäre anderswo schaffen. Bitcoin bricht den Informationsfluss an den Identitäten, hält sie ausserhalb des öffentlichen Buchs, während Beträge und Graph voll öffentlich bleiben. zkCoins schiebt die Grenze weiter: nur die 64-Byte-Nullifier sind öffentlich. Beträge, Parteien und Graph bleiben zwischen Sender und Empfänger, geschützt durch verbergende Commitments mit frischen Blinds. Adresswiederverwendung verknüpft keine On-Chain-Marker; der Nullifier-Schlüssel jedes Übergangs ist einmalig.



Als zusätzliche Brandmauer wird je Übergang ein neuer Kontoschlüssel verwendet, und üblich ist ein neues Empfangskonto je Gegenpartei, wenn Unverkettbarkeit unter Gegenparteien nötig ist. Die ehrliche Grenze bleibt: Peg-in und Peg-out von zkBTC sind gewöhnliche öffentliche Bitcoin-Transaktionen, Beträge und Zeit am Rand sind beobachtbar und korrelierbar. Innere Transfers sind abgeschirmt.

11. Die Bitcoin-Reserve

Wir beschreiben jetzt das durch Bitcoin gedeckte Token. zkBTC ist Token-Standard 3 auf dem zkCoins-Transfersystem: ein eins-zu-eins-Anspruch auf Bitcoin in Tresoren, nicht in Verwahrung.

Der Tresor ist eine Taproot-Ausgabe mit NUMS-Internschlüssel, es gibt also keinen Key-Path-Spend. Seine einzigen Ausgabepfade sind ein geordneter, N-aus-N vorsignierter BitVM2-Transaktionsgraph (Assert, Challenge, Disprove, Payout), fest beim Einrichten der Einlage. Nach dem Einrichten werden die Epoch-Signaturschlüssel gelöscht; Identitätsschlüssel bleiben. Der Tresor hat

dann für diesen Epoch-Graphen keinen lebenden Unterzeichner. Unter ehrlichem 1-aus-N-Löschen des Epoch-Schlüssels kann keine Koalition etwas ausserhalb des Graphen unterschreiben.

Peg-in (Mint) läuft so. Einleger und Operatoren unterschreiben gemeinsam eine MoveToBacked-Transaktion, die die Einlage unter den Tresor legt. Der Mint-Übergang beweist im Circuit, über den Light-Client-Beweis aus Abschnitt 8, dass MoveToBacked mindestens D_{mint} Blöcke tief liegt (tiefe Finalität, in der Grössenordnung 2016 Blöcke) und dass die Tresorinstanz zu den Asset-Bedingungen passt: die Policy-Wurzel der Operator-Anmeldung, Betragsgleichheit mit der Tresorausgabe und ein Einmal-Mintschlüssel, sodass jeder Tresor genau einmal prägt. Der Umlauf ist dann eine bedingte Obergrenze gegen die öffentliche Tresormenge auf der kanonischen Kette, kein unbedingtes Invariant der Kette allein.

Die Operator-Menge ist offen, erlaubnisfrei und wächst nur. Jeder darf beitreten, indem er eine Kaution mit Beweis des Schlüsselbesitzes hinterlegt. Niemand geht. Jeder neue Tresor ist N-aus-N über die aktuelle volle Menge. Ein Inhaber, der sich vor dem ersten Mint anmeldet, ist der ehrliche Unterzeichner jedes Tresors: ein späterer Sybil-Club kann ihn nicht weglassen, und er unterschreibt keinen Graphen, der einen Tresor an einen Dieb zahlt. Das schliesst das Ablassen durch selbst kontrollierte Operatoren (Angriff B). Anmelden vor dem ersten Mint; früher geprägte Münzen sind ein untrusted Prefix, weil das Token fungibel ist.

Der Gatekeeper ist optional und je Asset. Reine Proof-of-Work-Tiefe im Circuit beweist keine Kanonizität (eine private Gabel kann tief sein). Das ist Angriff A, ein Residuum der Bitcoin-Klasse, gemildert durch tiefe Finalität in der Grössenordnung 2016 Blöcke, sauber geschlossen nur wenn ein bestimmter Gatekeeper seine Mint-Signatur zurückhält, bis er auf seiner eigenen kanonischen Bitcoin-Sicht die Deckungs-Transaktion und die Anmeldeverpflichtung bestätigt hat. Er hat keinen Schlüssel auf dem Tresor, keine Rolle bei Transfers oder Einlösen und kann nicht einfrieren, beschlagnahmen oder umleiten. Er kann nur neue Mints ablehnen. Er ist nicht der Schluss von Angriff B und nicht das, was zkBTC vertrauensfrei macht. Ein nachlässiger Gatekeeper, der die Kanonizitätsprüfung überspringt, ermöglicht Angriff A, nicht das Ablassen späterer Tresore.

Peg-out (Einlösen) ist zuerst verbrennen. Der Einlöse-Übergang des Inhabers zerstört die Münze und veröffentlicht eine Einlösekennung (ihren Übergangs-Nullifier), die Auszahladresse und eine `max_fee`-Decke in verbergender Form festlegt. Es gibt kein Ent-Verbrennen: eine zu niedrig gesetzte Decke kann eine verbrannte Münze unbedient lassen. Jeder angemeldete Operator streckt die Auszahlung aus eigenen Mitteln innerhalb `max_fee` vor und holt genau den Einlösebetrag aus dem Tresor zurück, indem er korrekten Dienst über den BitVM2-Graphen behauptet. Jeder Beobachter kann eine falsche Behauptung im Widerspruchsfenster anfechten; ein einziger erfolgreicher Disprove streicht die Kaution und macht den Anspruch nichtig. Die Erstattung ist anspruchnehmergebunden, allein finanziert und von einer Partei autorisiert: die Auszahltransaktion ist konstruktionsgemäss nicht übertragbar. Unter den genannten Annahmen (1-aus-N beim Einrichten, mindestens ein ehrlicher lebender Challenger im Fenster, einwandfreie Circuit- und Graph-Kryptographie) ist der schlimmste Fall ein Einfrieren (das Einlösen wartet auf einen lebenden Operator), nicht Diebstahl. Ein kritischer Circuit- oder Graph-Soundness-Fehler ist der Diebstahlfall. Einwandfreie Circuit- und Graph-Kryptographie bleibt eine stehende Annahme.

Ehrliche Residuen bleiben, auch wenn Diebstahl geschlossen ist. Die Mitunterschrift des Einlegers macht einen getresorten Beitrag zugestimmt; ein Gatekeeper, der seine Mint-Signatur zurückhält, nachdem MoveToBacked gefeuert hat, lässt diese Einlage einen unwiderruflichen Reservebeitrag, ein Verlustpfad auf der Einlegerseite. Ohne Rat und ohne Admin-Schlüssel kann ein Einfrieren

unter den genannten Annahmen dauerhaft sein, solange kein künftiges Covenant-Upgrade kommt. „Nicht Diebstahl„ heisst nicht „wieder holbar“.

Wir betrachten die Wahrscheinlichkeit eines Aufholens gegen die kanonische Kette in denselben Begriffen wie Nakamotos Angreiferanalyse [1]. Das Rennen zwischen ehrlicher Kette und Angreifer ist ein binomialer Random Walk. Die Wahrscheinlichkeit, dass ein Angreifer von z Blöcken Rückstand aufholt, wenn p die Wahrscheinlichkeit ist, dass ein ehrlicher Knoten den nächsten Block findet, und $q = 1 - p$ die des Angreifers, ist die Gambler's-Ruin-Wahrscheinlichkeit

$$q_z = \begin{cases} 1 & \text{falls } p \leq q \\ (q/p)^z & \text{falls } p > q. \end{cases}$$

Der Prüfer eines neuen Mints kann den Fortschritt des Angreifers auf einer privaten Gabel nicht kennen. Wie lange muss das Netz warten, bevor es die Deckung eines Mints als gesiedelt behandelt? Nimmt man an, ehrliche Blöcke brauchten die erwartete Zeit, ist der mögliche Fortschritt des Angreifers, während der Mint auf z Bestätigungen wartet, eine Poisson-Verteilung mit Erwartungswert

$$\lambda = z \cdot q/p.$$

Die Wahrscheinlichkeit, dass der Angreifer jetzt noch aufholen könnte, ergibt sich, indem man die Poisson-Dichte jedes Fortschritts mit der Aufholwahrscheinlichkeit von dort multipliziert:

$$\sum_{k=0}^{\infty} \left(\lambda^k e^{-\lambda}/k!\right) \cdot \begin{cases} (q/p)^{(z-k)} & \text{falls } k \leq z \\ 1 & \text{falls } k > z. \end{cases}$$

Umgestellt, damit man nicht den unendlichen Schwanz summiert:

$$1 - \sum_{k=0}^{z} \left(\lambda^k e^{-\lambda}/k!\right) \cdot \left(1 - (q/p)^{(z-k)}\right).$$

Als C-Code:

```
double AttackerSuccessProbability(double q, int z)
{
  double p = 1.0 - q;
  double lambda = z * (q / p);
  double sum = 1.0;
  int i, k;
  for (k = 0; k <= z; k++)
  {
    double poisson = exp(-lambda);
    for (i = 1; i <= k; i++)
      poisson *= lambda / i;
    sum -= poisson * (1 - pow(q / p, z - k));
  }
  return sum;
}
```

Einige Ergebnisse: die Wahrscheinlichkeit fällt exponentiell mit z:

```
q=0.1
z=0 P=1.0000000
z=1 P=0.2045873
z=2 P=0.0509779
z=3 P=0.0131722
z=4 P=0.0034552
z=5 P=0.0009137
z=6 P=0.0002428
z=7 P=0.0000647
z=8 P=0.0000173
z=9 P=0.0000046
z=10 P=0.0000012
```

```
q=0.3
z=0 P=1.0000000
z=5 P=0.1773523
z=10 P=0.0416605
z=15 P=0.0101008
z=20 P=0.0024804
z=25 P=0.0006132
z=30 P=0.0001522
z=35 P=0.0000379
z=40 P=0.0000095
z=45 P=0.0000024
z=50 P=0.0000006
```

Für P kleiner als 0,1 %:

```
P < 0.001
q=0.10 z=5
q=0.15 z=8
q=0.20 z=11
q=0.25 z=15
q=0.30 z=24
q=0.35 z=41
q=0.40 z=89
q=0.45 z=340
```

Selbst die Extremzeile ($q = 0.45$) braucht nur $z = 340$ für $P < 0.1\%$. Die Mint-Abrechnung wartet $z = D_{\text{mint}} \approx 2016$, wo die reine Aufholwahrscheinlichkeit weiter zusammenbricht. Die geschlossene Form $q_z = (q/p)^z$ für $p > q$ bei tiefer Finalität ergibt für drei Angreiferanteile:

```
q=0.10 (q/p = 1/9): z=2016 P ≈ 10^-1924
q=0.30 (q/p = 3/7): z=2016 P ≈ 10^-742
q=0.45 (q/p = 9/11): z=2016 P ≈ 10^-176
```

Bei tiefer Finalität ist das Aufholen gegen die ehrliche Kette selbst für einen 45-Prozent-Angreifer wirtschaftlich geschlossen: $z = 2016$ liefert Wahrscheinlichkeiten der Ordnung 10^{-176} und kleiner. Das ist nicht Angriff A: eine private Gabel kann tief sein, ohne die ehrliche Kette einzuholen. Das Residual-Mint-Risiko ist daher die Kanonizität der bewiesenen Kette, nicht rohes Proof-of-Work-Aufholen. Ein bestimmter Gatekeeper kann die Kanonizität zur Mint-Zeit prüfen. Das Sybil-Operator-Ablassen schliesst die nur-wachsende Operator-Menge, nicht der Gatekeeper.

12. Schluss

Wir haben ein System für privates elektronisches Geld vorgeschlagen, gedeckt durch Bitcoin, ohne Verwahrvertrauen. Münzen sind Ketten beweis tragender Zustandsübergänge. Bitcoin ordnet Nullifier fester Grösse, sodass Doppelausgaben öffentlich erkennbar sind, ohne Beträge preiszugeben. Gültigkeitsbeweise ersetzen die globale Prüfung eines Wertbuchs; der On-Chain-Abdruck bleibt konstant. Die Reserve ist durch vorsignierte Betrugsnachweis-Pfade mit offener, nur-wachsender Operator-Menge gesichert: ein Inhaber, der sich vor dem ersten Mint anmeldet, ist der ehrliche Unterzeichner jedes Tresors und kann den eigenen Ausgang bedienen. Unter 1-aus-N beim Einrichten, mindestens einem ehrlichen lebenden Challenger in jedem Fenster und einwandfreier Kryptographie können Operatoren nicht stehlen. In diesem Sinn ist zkBTC effektiv vertrauensfrei. Ein Gatekeeper, wenn einer bestimmt ist, handelt nur an der Mint-Grenze und kann Transfers oder Einlösen nicht einfrieren. Der Ausgang hängt an der Lebendigkeit eines angemeldeten Operators, nie an einer Erlaubnis. Die Garantien sind an die aufgezählten Annahmen gebunden. Dieses Paper ist eine informative Einführung; der Entwurf ist normativ in [8] festgelegt und wartet auf Umsetzung.

Literatur

[1] S. Nakamoto, „Bitcoin: A Peer-to-Peer Electronic Cash System,“ https://bitcoin.org/bitcoin.pdf, 2008.

[2] R. Linus, „zkCoins,“ https://gist.github.com/RobinLinus/d036511015caea5a28514259a1bab119, 2023.

[3] J. Nick, L. Eagen, R. Linus, „Shielded CSV: Private and Efficient Client-Side Validation,“ Cryptology ePrint Archive 2025/068, 2025.

[4] P. Todd, „Scalable Semi-Trustless Asset Transfer via Single-Use-Seals and Proof-of-Publication,“ https://petertodd.org/2017/scalable-single-use-seal-asset-transfer, 2017.

[5] R. Linus et al., „BitVM2: Bridging Bitcoin to Second Layers,“ https://bitvm.org/bitvm_bridge.pdf, 2024.

[6] E. Bal, L. Aumayr, A. İyidoğan, G. Scaffino, H. Karakuş, C. E. Aslan, O. S. Thyfronitis Litos, „Clementine: A Collateral-Efficient, Trust-Minimized, and Scalable Bitcoin Bridge,“ Cryptology ePrint Archive 2025/776, 2025.

[7] P. Wuille, J. Nick, T. Ruffing, „Schnorr Signatures for secp256k1,“ BIP-340, 2020.

[8] „zkCoins-Protokollspezifikation,“ https://github.com/zk-coins/docs, und „zkBTC-Tokenstandard,“ https://github.com/zk-BTC/zkbtc, 2026.