Skalierbare und vertraulichkeitswahrende Off-Chain Berechnungen[1]

Jacob Eberhardt[2]

Abstract: Blockchains erlauben sich gegenseitig mistrauenden Parteien gemeinsame Transaktionen auszuführen und deren Historie unveränderlich zu speichern. Aufgrund ihres technischen Aufbaus leiden Blockchains jedoch unter niedrigem Durchsatz, fehlender Skalierbarkeit und schwachen Datenschutzgarantien. Diese Arbeit adressiert diese Probleme durch das neuartige Konzept des Off-Chainings: Daten und Berechnungen werden von einer Blockchain auf externe Resourcen ausgelagert - jedoch ohne dabei Schlüsseleigenschaften der Blockchain zu kompromittieren. Insbesondere verifizierbare Off-Chain Berechnungen stellen ein mächtiges Werkzeug zur Erhöhung des Durchsatzes und der Gewährleistung von Vertraulichkeit dar. Allerdings fehlen geignete Realisierungsansätze. Unsere Analyse des Designraums identifiziert zk-SNARKs, eine Klasse nicht-interaktiver Protokolle für kryptographische Zero-Knowledge Beweise, als vielversprechenden Ansatz. Allerdings ist die Instanziierung dieser Protokolle komplex und somit wenigen Experten vorbehalten. Geeignete Programmierabstraktionen und softwaretechnische Werkzeuge fehlen. Um dieses Problem zu adressieren, präsentieren wir ZoKrates, die erste höhere Programmiersprache und Sammlung von Softwarewerkzeugen zur Übersetzung und Ausführung zk-SNARK-basierter verifizierbarer Off-Chain Berechnungen. Wir demonstrieren Relevanz und Anwendbarkeit an drei dezentralen Applikationen: Peer-to-Peer Energiehandel, Blockchain-Relays und anonyme Token-Transfers. Die im Kontext dieser Arbeit entstandenen Softwarelösungen finden darüber hinaus unabhängige Anwendung in Wissenschaft und Industrie.

1 Motivation und Problemstellung

Blockchaintechnologien erlauben sich gegenseitig misstrauenden Akteuren zensurresistent Transaktionen in einem verteilten System zu verarbeiten und dabei eine unveränderliche Transaktionshistorie zu etablieren, ohne hierfür eine vertrauenswürdige dritte Partei hinzuzuziehen. Allerdings stehen diese Eigenschaften mit anderen wünschenswerten Qualitätseigenschaften verteilter Systeme in Konflikt.

In aktuellen Blockchain-Netzwerken steigt der Durchsatz nicht mit der Anzahl der aktiven Knoten. Blockchains skalieren nicht. Der Durchsatz ist gering, die Transaktionskosten und Verarbeitungslatenzen sind hoch: Bitcoin verarbeitet derzeit 7 Transaktionen pro Sekunde, und Blöcke, die eine Reihe von Transaktionen bestätigen, werden im Durchschnitt alle 10 Minuten erstellt; Ethereum verarbeitet bis zu 25 Transaktionen pro Sekunde und hat ein durchschnittliches Blockintervall von 15 Sekunden. Im Vergleich dazu verarbeitet

[1] Englischer Titel der Dissertation [Eb21]: “Scalable and Privacy-preserving Off-Chain Computations”

[2] Die Dissertation ist in der Forschungsgruppe Information Systems Engineering (ISE) an der Technischen Universität Berlin entstanden. Kontakt: mail@jacobeberhardt.de

der Zahlungsabwickler Visa im Durchschnitt circa 1700 Transaktionen pro Sekunde, die innerhalb von Sekunden bestätigt werden. Dies ist ein grundlegender Nachteil für dezentrale Anwendungen, die mit traditionellen Diensten konkurrieren, die nicht unter solchen Einschränkungen leiden. Das Problem der Skalierbarkeit ist in der Forschung wohl bekannt und wird als intrinsisch schwierig erachtet [Cr16].

Die zweite wünschenswerte Eigenschaft ist der Schutz der Privatsphäre und die Möglichkeit zur Verarbeitung vertraulicher Daten. Diese Anforderung steht jedoch in einem grundlegenden Widerspruch zur derzeitigen Funktionsweise moderner Blockchains: In aktuellen Blockchain-Netzwerken führen alle Knoten redundant jede einzelne Transaktion aus. Diese Arbeitsweise ist grundsätzlich erforderlich, um die Korrektheit der Verarbeitungsergebnisse zu gewährleisten. Gleichzeitig bedeutet dies, dass alle Informationen, die verarbeitet werden, allen Knoten im Netzwerk bekannt sein und von ihnen gespeichert werden müssen. Andernfalls wäre eine redundante Ausführung nicht möglich. Folglich dürfen private oder vertrauliche Daten nicht auf der Blockchain verarbeitet werden - sie würden sofort netzwerköffentlich.

In der diesem Artikel zugrunde liegenden Arbeit widmen wir uns der Frage, wie diese grundlegenden Herausforderungen in Bezug auf Skalierbarkeit und Datenschutz in Blockchain-basierten Anwendungen adressiert werden können. Wir stellen unsere Beiträge und Resultate diesbezüglich in den nachfolgenden Abschnitten in verkürzter Form dar; für eine wesentlich tiefere Darstellung verweisen wir auf die Dissertationsschrift [Eb21]. In Abschnitt 2 führen wir zunächst Off-Chaining als grundlegenden Mechanismus, um Skalierungs- und Datenschutzprobleme dezentraler Anwendungen zu lösen, ein. Darauf aufbauend entwickeln wir in Abschnitt 3 ZoKrates, eine Programmiersprache und Sammlung von Softwarewerkzeugen, die es Entwicklern dezentraler Anwendungen ermöglicht, Off-Chain Berechnungen auf nutzerfreundliche Art zu spezifizieren und auszuführen. Im Rahmen einer ausführlichen Evaluation wird die praktische Signifikanz von ZoKrates und Off-Chaining in Abschnitt 4 durch die exemplarische Anwendung auf drei relevante Blockchain-basierte Applikationen demonstriert, welche sich mit Skalierbarkeits- oder Datenschutzproblemen konfrontiert sehen.

2 Off-Chaining und Off-Chain Berechnungen

In unserer Arbeit schlagen wir Off-Chaining als einen grundlegenden Ansatz vor, um Herausforderungen im Bezug auf Skalierbarkeit und Datenschutz im Kontext Blockchain-basierter Anwendungen zu adressieren.

Wir definieren Off-Chaining als die Auslagerung von Berechnungen und/oder Daten aus der Blockchain, wobei die wichtigsten Eigenschaften der Blockchain so wenig wie möglich beeinträchtigt werden. Die Kernidee besteht darin, die Datenspeicherung sowie den Rechenaufwand auf der Blockchain zu minimieren, indem Blockchain-externe Ressourcen, wie genutzt werden. Durch die Verringerung des Verarbeitungsaufwands auf der Blockchain

werden Kapazitäten für andere dezentrale Anwendungen frei. Außerdem ist die Speicherung sensibler Daten außerhalb der Blockchain die einzige Möglichkeit, die Privatsphäre zu gewährleisten — alle auf der Blockchain gespeicherten Informationen sind per Definition öffentlich einsehbar, da sie von allen Knoten zur Transaktionsvalidierung genutzt werden müssen.

2.1 Off-Chaining Entwurfsmuster

Um die Lücke zwischen dieser abstrakten Definition und praktikablen Off-Chaining-Ansätzen zu schließen, analysieren wir wiederkehrende Herausforderungen und Lösungsideen im dezentralen Anwendungsdesign [ET17]. In der Arbeit strukturieren und kategorisieren wir diese in fünf verschiedene Off-Chaining Entwurfsmuster bzw. Patterns:

1. Inhaltsadressierbarer Off-Chain Speicher Entwurfsmuster
2. Verifizierbare Off-Chain Berechnungen Entwurfsmuster
3. Off-Chain Signaturen Entwurfsmuster
4. Optimistische Finalisierungs Entwurfsmuster
5. Niedriger Contract Fußabdruck Entwurfsmuster

Die Instanziierung dieser Entwurfsmuster ermöglicht es Entwicklern, die Herausforderungen der Skalierbarkeit und des Datenschutzes zu bewältigen, mit denen sie bei der Entwicklung von Blockchain-basierten Anwendungen häufig konfrontiert werden. Während Skalierbarkeit und Datenschutz nicht im Widerspruch zueinander stehen und durch Off-Chaining gleichzeitig angegangen werden können, stellen wir fest, dass ein Tradeoff zur Verfügbarkeit besteht, der sorgfältige Abwägung verlangt.

2.2 Off-Chaining von Berechnungen

Aus unserer Analyse schließen wir, dass Off-Chain Berechnungen, wie sie im verifizierbare Off-Chain Berechnungen Entwurfsmuster beschrieben werden, besonders gut geeignet sind, um Datenschutzanforderungen in dezentralen Anwendungen zu begegnen, da sie eine vertraulichkeitswahrende Verarbeitung von Off-Chain Daten ermöglichen.

Wenn die Verifizierung von Off-Chain berechneten Ergebnissen auf der Blockhain außerdem kostengünstiger ist als die On-Chain Ausführung eben dieser Berechnung, kann dieser Ansatz den Durchsatz zudem direkt verbessern. Während in der Literatur einige Realisierungen von Off-Chain Berechnungen für bestimmte Kontexte vorgeschlagen wurden, gibt es keine systematische Analyse möglicher Ansätze, ihrer Eigenschaften und ihres Vergleichs.

In unserem zweiten Hauptbeitrag befassen wir uns mit diesem Problem, indem wir systematisch den Designraum für Off-Chain Berechnungen untersuchen, grundlegende Kategorien von Off-Chain Berechnungsansätzen identifizieren und bestehende Vorschläge aus der weißen und grauen Literatur kategorisieren. Anschließend führen wir eine vergleichende Analyse durch, bei der die Ansätze in Bezug auf Skalierbarkeit, Datenschutz, Sicherheit und Programmierbarkeit gegenübergestellt werden.

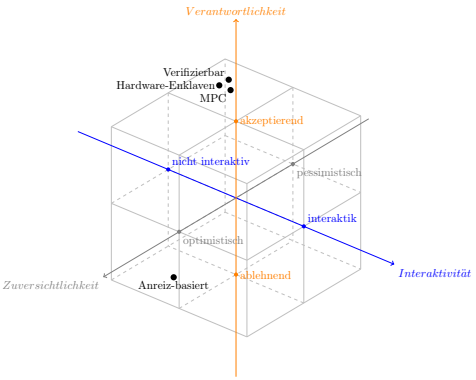

Abb. 1: Designraum der Protokolle für Off-Chain Berechnungen: Dimensionen und Ausprägungen.

Im Rahmen dieser Analyse wurden vier grundlegende Ansätze identifiziert, die sich in der Art, wie die Korrektheit der Blockchain-externen Berechnungen sichergestellt wird, unterscheiden [EH18]: Kryptographisch verifizierbare Berechnungen generieren direkt überprüfbare Korrekheitszertifikate, während Enklaven-basierte Ansätze sich auf isolierte Hardwaremodule verlassen. Anreizbasierte Ansätze nutzen interaktive Protokolle in Kombination mit werthaltigen Blockchain-Tokens, z. B. Bitcoin, um die Korrektheit von Berechnungsergebnissen spieltheoretisch durchzusetzen. Der letzte Ansatz basiert auf der Ausführung von Secure Multiparty Computation (MPC) Protokollen in einem Netzwerk von Off-Chain Knoten.

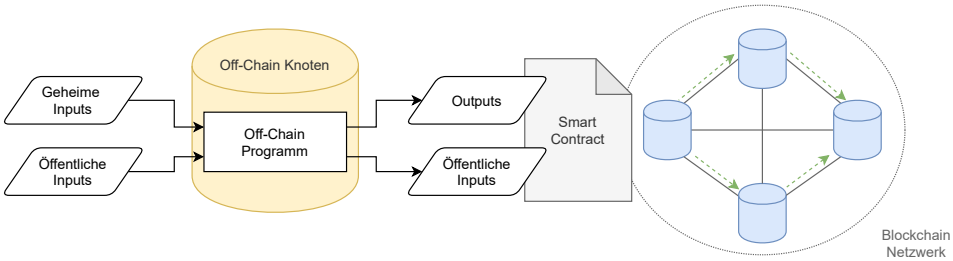

Abb. 2: Komponenten in Protokollen für Off-Chain Berechnungen: Ein Blockchain-externer Knoten erhält öffentliche und geheime Inputs, verarbeitet diese in einem Off-Chain Programm und sendet die Berechnungsergebnisse, sowie öffentliche Inputs an einen Smart Contract.

Als Ergebnis der vergleichenden Analyse stellen wir fest, dass zk-SNARKs aus der Gruppe der kryptographisch verifizierbaren Berechnungen einen besonders leistungsfähigen Ansatz im Bezug auf Skalierbarkeit und Datenschutz darstellen. Eine vereinfachte Ergebnisübersicht ist in Tab. 1 dargestellt.

Tab. 1: Vergleich der Ansätze für Off-Chain Berechnungen

		Skalierbarkeit		*Vertraulichkeit*	*Sicherheit*		*Programmierbarkeit*
Ansatz	**Realisierung**	**On-Chain Verifikation**	**Off-Chain Berechnung**	**Private Inputs**	**Sicherheitsannahme**	**Post-Quantum Sicherheit**	**Progr. Abstraktion**
Verifizierbare Berechnungen	zk-SNARK	*Einmaliges Setup:* $O(n)$, n Anzahl Multiplikationsgatter im Schaltkreis *Wiederholte Verifikation:* $O(1)$ *Beweisgröße:* $O(1)$, z.B. 3 Gruppenelemente [Gr16], i.e.127 bytes für BN254 Kurve	$O(n \log n)$, n Anzahl Multiplikationsgatter im Schaltkreis	ja	Knowledge of Exponent Annahme & Setup korrekt ausgeführt	nein	Arithmetische Schaltkreise
	Bulletproofs	*Verify:* $O(n)$, n Anzahl Multiplikationsgatter im Schaltkreis *Beweisgröße:* wenige Kilobytes, $O(\log n)$, n Anzahl Multiplikationsgatter im Schaltkreis	$O(n)$, n Anzahl Multiplikationsgatter im Schaltkreis	ja	Diskrete-Logarithmus-Annahme	nein	Arithmetische Schaltkreise
	zk-STARK	*Verify:* $O(\log^2 n)$, n Anzahl Multiplikationsgatter AIR zu Schaltkreis ausgerollt *Beweisgröße:* wenige hundert Kilobytes, $O(\log^2 n)$, n Anzahl Multiplikationsgatter AIR zu Schaltkreis ausgerollt	$O(n \log^2 n)$, n Anzahl Multiplikationsgatter AIR zu Schaltkreis ausgerollt	ja	Kollisionsresistente Hashfunktionen	ja	AIR (in Schaltkreise ausrollbar)
Hardware-Enklaven		Validierung der Attestation der Enklave: $O(1)$, Signaturprüfung	Native Ausführung & Attestations-Overhead	ja	TEEs sind isoliert & Vertrauen in Remote-Attestation Zertifikate	nein	Sprachen, die in TEE-kompat. Maschinencode kompilieren
Anreiz-basiert		Binärsuche & ein Berechnungsschritt: $O(\log n)$, n Anzahl Berechnungsschritte	Overhead Virtuelle Maschine (Ausführungshistorie)	nein	Ökonomisch rationale Teilnehmer	ja	Sprachen, die in VM-Instruktionsset kompilieren
MPC-basiert		*On-Chain Auditor:* $O(n)$, n Anzahl Gatter im Schaltkreis *Größe Audit-Trail:* $O(n)$, n Anzahl Gatter im Schaltkreis	$O(n)$, n Anzahl Gatter im Schaltkreis	ja	Mindestens ein ehrlicher Knoten & Mindestehrlichkeitsschranke für Schutz privater Inputs und Liveness	ja	Boolsche oder arithmetische Schaltkreise

3 ZoKrates - Programmierung von Off-Chain Berechnungen

In der vorangegangenen Analyse wurden zk-SNARKs als geeigneter Ansatz für die Realisierung von allgemeinen Off-Chain Berechnungen identifiziert. Allerdings ist die konkrete Instanziierung schwierig: Berechnungen müssen in schwer zu verwendenden Low-Level Abstraktionen spezifiziert werden, und die On-Chain Verifikation ist komplex, da sie tiefes Wissen über die verwendeten kryptographischen Protokolle erfordert.

Wir schließen diese Lücke mit dieser Arbeit, indem wir ZoKrates, das erste Framework für effiziente Zero-Knowledge Off-Chain Berechnungen entwerfen, implementieren und evaluieren [ET18]. ZoKrates ermöglicht es dezentralen Anwendungen, ihre Anforderungen an Datenschutz und Skalierbarkeit zu erfüllen, indem es eine entwicklerfreundliche Abstraktion für die Spezifikation, die Off-Chain Ausführung und die On-Chain Überprüfung von verifizierbaren Off-Chain Berechnungen auf Basis von zk-SNARKs bereitstellt. Benutzerfreundlichkeit, Effizienz und Allgemeingültigkeit stellten die Hauptziele für ZoKrates als Framework für verifizierbare Zero-Knowledge Off-Chain Berechnungen dar.

ZoKrates besteht aus einer domänenspezifischen Programmiersprache, die die Besonderheiten der zugrunde liegenden Abstraktionen abbildet und es Entwicklern ermöglicht, Off-Chain Berechnungen bequem als Programme auf Abstraktionsebene einer Hochsprache zu spezifizieren. Diese Programme werden dann in die proprietäre ZoKrates Intermediate Representation übersetzt und durch den ZoKrates Interpreter ausgeführt. Anschließend kann ein Korrektheitsbeweis für diese Programmausführung generiert werden. Um eine On-Chain Verifikation zu ermöglichen, unterstützt ZoKrates die Generierung und den Export von Verifikations-Smart Contracts, die die Korrektheit von Off-Chain Berechnungen überprüfen. In Abb. 3 geben wir einen Überblick über alle Schritte, die von der Spezifikation einer Berechnung als ZoKrates Programm bis zur Verifizierung dessen Ausführung auf der Blockchain erforderlich sind.

```
1  import "hashes/sha256/512bit" as sha256
2
3  def main(private u32[16] input) -> u32[8]:
4    u32[8] h = sha256(input[0..8], input[8..16])
5    return h
```

List. 1: Beispiel für ein einfaches ZoKrates Programm, das einen SHA-256-Hash auf geheimen Inputparametern berechnet. Damit kann das Wissen über das Urbild des errechneten SHA-256-Hashes bewiesen werden, ohne das Urbild je offenzulegen.

Die Implementierung des ZoKrates Frameworks ist seit der Veröffentlichung der ursprünglichen ZoKrates-Publikation [ET18] beträchtlich gereift und hat sich zu einem aktiven Open-Source-Projekt mit mehreren Beitragenden entwickelt. Dennoch haben sich die Kernkomponenten und Ideen nicht verändert. Code und Nutzerdokumentation sind auf GitHub verfügbar[3].

Während sich die Implementierung auf die Verwendung mit der Ethereum Blockchain fokussiert, unterstützt ihre Architektur jedoch beliebige Blockchains, die über eine ausreichend leistungsfähige Ausführungsumgebung für die Beweisverifizierung verfügen. Darüber hinaus wird jede Implementierung eines verifizierbaren Berechnungsschemas durch den

[3] https://github.com/ZoKrates/ZoKrates
[4] https://remix.ethereum.org

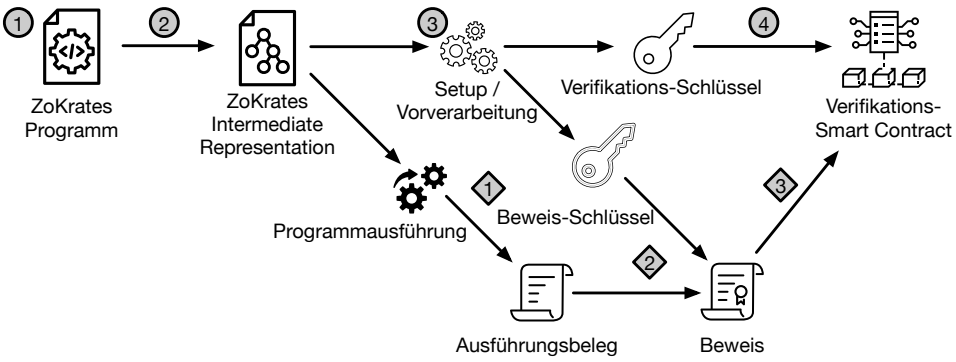

Abb. 3: Überblick über den ZoKrates Übersetzungs-, Ausführungs- und Beweisprozess. Zunächst wird eine Reihe von einmaligen Vorbereitungsschritten für ein Off-Chain Programm durchgeführt: Programmspezifikation, Kompilierung, Setup und Erzeugung eines Verifikations-Smart-Contracts. Diese Schritte sind mit eingekreisten Zahlen markiert. Anschließend wird das Off-Chain Programm ausgeführt, ein Beweis über die Korrektheit der Ausführung generiert und zur Überprüfung an den Verifikations-Smart-Contract übergeben. Diese Schritte werden für jede Programmausführung wiederholt und sind mit Rauten markiert.

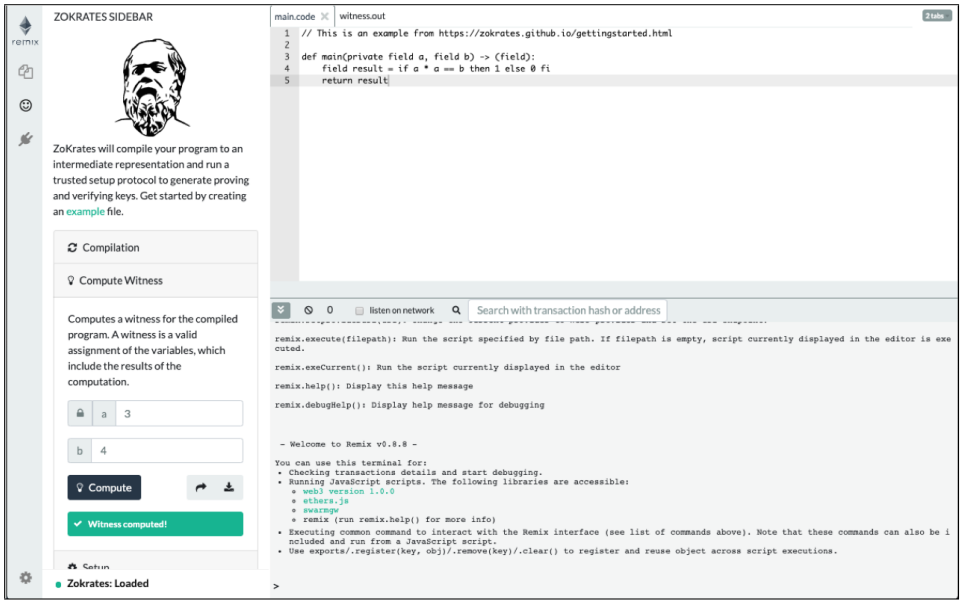

Abb. 4: ZoKrates Development in der Ethereum Remix IDE. Der Nutzer kann den kompletten Prozess von Programmspezifikation bis Beweisgenerierung und -verifizierung im Browser durchlaufen[4]

modularen Aufbau der Architektur unterstützt, solange das implementierte Schema mit der ZoKrates Intermediate Representation kompatibel ist.

Effizienz und Praktikabilität des grundlegenden Konzepts sowie der konkreten ZoKrates Implementierung werden in der zugrunde liegenden Arbeit in einer ausführlichen Performance-Evaluation belegt. Hierzu werden Programme aus den im nachfolgenden Abschnitt beschriebenen Anwendungen herangezogen.

4 Anwendungen

In einer ausführlichen Evaluierung zeigen wir, wie Datenschutz- und Skalierbarkeitsprobleme, mit denen reale Blockchain-basierte Anwendungen konfrontiert sind, durch verifizierbare Off-Chain Berechnungen gelöst werden können. Konkret beschreiben wir ZoKrates-basierte Varianten dezentraler Anwendungen für datenschutzfreundliche Token-Transfers und Peer-to-Peer-Energiehandel sowie ein skalierbares Blockchain-Relay und demonstrieren damit die Praxistauglichkeit des Frameworks.

Die dezentrale Peer-to-Peer-Energiehandelsanwendung und das skalierbare Blockchain-Relay wurden von uns mit unseren Co-Autoren vorgeschlagen, implementiert und eva-

luiert [Eb20]. Im Gegensatz dazu wurde die ZoKrates-basierte anonyme Token-Transfer-Anwendung unabhängig von der Blockchain Forschungs- und Entwicklungsabteilung von Ernst & Young entwickelt und implementiert, was Nutzbarkeit, Nützlichkeit und Reife von ZoKrates unterstreicht.

4.1 Privatsphäre-wahrender Energiehandel in zukünftigen Stromnetzen

In unserer ersten Anwendung verwenden wir ZoKrates, um Smart-Meter-Daten in zukünftigen Energienetzwerken zu verbergen und gleichzeitig eine vertrauenswürdige Verarbeitung zum Zweck der gemeinsamen Nutzung von Energie in einer Gemeinschaft von Haushalten zu ermöglichen. Wir haben unsere Lösung im Rahmen von BloGPV[5], einem nationalen Forschungsprojekt, implementiert und evaluiert [Eb20, WE20]. Das entstandene System schützt die Privatsphäre der teilnehmenden Personen und erhöht gleichzeitig die Rentabilität der erneuerbaren Energieerzeugung.

Ganz allgemein zeigen wir, wie ZoKrates-basierte Off-Chain Berechnungen mit On-Chain Commitments kombiniert werden können, um Algorithmen in einer sich mistrauenden Gruppe mit Blockchain-Eigenschaften auszuführen und dabei die Privatsphäre zu wahren.

4.2 zkRelay

Zweitens stellen wir zkRelay vor, ein skalierbares Blockchain-Relay, das Off-Chain Berechnungen für die Validierung von Block-Headern nutzt [WE20]. Wir demonstrieren, wie ZoKrates-basierte Off-Chain Berechnungen verwendet werden können, um einer Blockchain zu ermöglichen, Daten und Ereignisse einer anderen Blockchain auf effiziente und skalierbare Weise zu validieren. Wir stellen ein Relay-Design vor, das die Header-Validierung durch überprüfbare Off-Chain Berechnungen außerhalb der Blockchain realisiert und dadurch die Kosten für die Validierung von Blockheadern einer Quell-Blockchain auf einer Ziel-Blockchain reduziert. Als Proof-of-Concept stellen wir eine ZoKrates-basierte Implementierung für ein Bitcoin-Relay auf der Ethereum-Blockchain zur Verfügung, die die Validierung von 504 Bitcoin-Headern in einer einzigen Ethereum-Transaktion ermöglicht.

Unsere zkRelay-Implementierung reduziert die Kosten für die Validierung von Bitcoin-Headern auf der Ethereum-Blockchain um das bis zu 187-fache im Vergleich zu BTC-Relay, der state-of-the-art Lösung.

4.3 Anonyme Tokentransfers

Neben protokollnativen Tokens, z. B. Bitcoin oder Ether, ermöglichen programmierbare Blockchain-Plattformen den Entwicklern dezentraler Anwendungen, ihre eigenen Token

[5] `https://blogpv.net/`

durch Smart Contracts zu erstellen. Solche Token implementieren häufig eine standardisierte Schnittstelle, um die Kompatibilität mit Börsen und anderen Anwendungen zu gewährleisten, z. B. den ERC-20-Standard. Wie native Tokens leiden auch diese benutzerdefinierten Tokens unter schwachen Datenschutzgarantien: Eigentumsinformationen werden einsehbar in Smart Contracts gespeichert, bei jeder Übertragung sind Absender und Empfänger sowie die Anzahl der übertragenen Token für alle Netzwerkteilnehmer sichtbar. Pseudonyme Adressen bieten keinen ausreichenden Schutz der Privatsphäre [An13, RH13].

Um dieses Problem zu adressieren, hat Ernst & Young Nightfall entwickelt, ein Protokoll, das datenschutzkonforme Token-Transfers für das öffentliche Ethereum-Netzwerk realisiert. Hierzu erweitert Nightfall die Ideen von Zerocash [Sa14], um datenschutzfreundliche Übertragungen von Ethereum-basierten benutzerdefinierten Tokens nach den Standards ERC-20 und ERC-721 zu unterstützen. ZoKrates dient hierbei als zentrales Werkzeug zur Realisierung der Protokollprimitive: Es erlaubt die Einhaltung von Transferregeln in Off-Chain Berechnungen zu beweisen ohne die dabei verwendeten Daten auf der Blockchain zu veröffentlichen. Token-Übertragungen in Nightfall sind immer anonym, d.h., Sender und Empfänger bleiben verborgen.

5 Zusammenfassung

In der diesem Artikel zu Grunde liegenden Dissertation wurde Off-chaining als grundlegender Ansatz eingeführt, um Skalierbarkeits- und Datenschutzprobleme im Kontext Blockchain-basierter Anwendungen zu adressieren. Off-Chaining-basierte Lösungsideen für wiederkehrende Herausforderungen im Design dezentraler Anwendungen wurden identifiziert und in Form von Entwurfsmuster strukturiert. Eine vergleichende Analyse von Instanziierungsoptionen für Off-chain Berechnungen zeigte die besondere Eignung von zk-SNARKs, einer Klasse nicht-interaktiver kryptographischer Protokolle für Zero-Knowledge Beweise. Es fehlten jedoch geeignete Programmierabstraktionen, der Einsatz bleibt Experten vorbehalten.

Diese Lücke schließen wir mit ZoKrates, einer höheren Programmiersprache und Sammlung von Softwarewerkzeugen zur Übersetzung und Ausführung zk-SNARK-basierter verifizierbarer Off-Chain Berechnungen. Im Rahmen einer ausführlichen Evaluation demonstrierten wir die Praktikabilität und Anwendbarkeit von ZoKrates an drei dezentralen Applikationen: dezentraler Energiehandel, Blockchain-Relays und anonyme Token-Transfer. Über die Arbeit hinausgehend finden ZoKrates und die zugehörigen Softwarewerkzeuge unabhängige Anwendung in Wissenschaft und Industrie.

Literaturverzeichnis

[An13] Androulaki, Elli; Karame, Ghassan O; Roeschlin, Marc; Scherer, Tobias; Capkun, Srdjan: Evaluating user privacy in bitcoin. In: International Conference on Financial Cryptography and Data Security. Springer, S. 34–51, 2013.

[Cr16] Croman, Kyle; Decker, Christian; Eyal, Ittay; Gencer, Adem Efe; Juels, Ari; Kosba, Ahmed; Miller, Andrew; Saxena, Prateek; Shi, Elaine; Sirer, Emin Gün et al.: On scaling decentralized blockchains. In: International Conference on Financial Cryptography and Data Security. Springer, S. 106–125, 2016.

[Eb20] Eberhardt, Jacob; Peise, Marco; Kim, Dong-Ha; Tai, Stefan: Privacy-Preserving Netting in Local Energy Grids. In: Proceedings of the IEEE International Conference on Blockchain and Cryptocurrency 2020. IEEE, 2020.

[Eb21] Eberhardt, Jacob: Scalable and privacy-preserving off-chain computations. Doctoral thesis, Technische Universität Berlin, 2021.

[EH18] Eberhardt, Jacob; Heiss, Jonathan: Off-chaining Models and Approaches to Off-chain Computations. In: Proceedings of the 2nd Workshop on Scalable and Resilient Infrastructures for Distributed Ledgers. ACM, S. 7–12, 2018.

[ET17] Eberhardt, Jacob; Tai, Stefan: On or Off the Blockchain? Insights on Off-Chaining Computation and Data. In: Proceedings of the European Conference on Service-Oriented and Cloud Computing. Springer, S. 3–15, 2017.

[ET18] Eberhardt, Jacob; Tai, Stefan: ZoKrates - Scalable Privacy-Preserving Off-Chain Computations. In: IEEE International Conference on Blockchain. IEEE, 2018.

[Gr16] Groth, Jens: On the size of pairing-based non-interactive arguments. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, S. 305–326, 2016.

[RH13] Reid, Fergal; Harrigan, Martin: An analysis of anonymity in the bitcoin system. In: Security and Privacy in Social Networks, S. 197–223. Springer, 2013.

[Sa14] Sasson, Eli Ben; Chiesa, Alessandro; Garman, Christina; Green, Matthew; Miers, Ian; Tromer, Eran; Virza, Madars: Zerocash: Decentralized anonymous payments from bitcoin. In: Security and Privacy (SP), 2014 IEEE Symposium on. IEEE, S. 459–474, 2014.

[WE20] Westerkamp, Martin; Eberhardt, Jacob: zkRelay: Facilitating Sidechains using zkSNARK-based Chain-Relays. In: Proceedings of the IEEE European Symposium on Security and Privacy Workshops. IEEE, S. 378–386, 2020.

Jacob Eberhardt promovierte 2021 in der Gruppe Information Systems Engineering von Prof. Tai an der Technischen Universtität Berlin. Seine Forschungsarbeiten wurden mit einem IEEE Best Paper Award ausgezeichnet, gewannen einen Samsung Next Research Grant und resultierten in einem Open Source Projekt mit Förderung durch die Ethereum Foundation. Zudem diente er in verschiedenen Programmkomitees internationaler Konferenzen im Bereich Blockchains. Zuvor legte er einen Bachelor und Masterabschluss in Wirtschaftsingenieurwesen mit Schwerpunkt Informatik am Karlsruher Institut für Technologie (KIT) ab.