Einführung in das Lightning Netzwerk

Das Second-Layer-Blockchain-Protokoll für effiziente Bitcoin-Zahlungen verstehen und nutzen

DAS VORWORT

Vorwort

Das Lightning-Netzwerk (engl. *Lightning Network*, kurz auch LN) ist ein Second-Layer-Peer-to-Peer-Netzwerk, das es uns erlaubt, Bitcoin-Zahlungen »Off-Chain« abzuwickeln, d. h., ohne sie als Transaktionen auf der Bitcoin-Blockchain bestätigen zu müssen.

Das Lightning-Netzwerk bietet uns sichere, günstige, schnelle und deutlich vertraulichere Bitcoin-Zahlungen, und das auch bei sehr kleinen Beträgen.

Basierend auf der Idee von Zahlungskanälen, die erstmals von Bitcoin-Erfinder Satoshi Nakamoto vorgeschlagen wurden, ist das Lightning-Netzwerk ein geroutetes Netzwerk, bei dem Zahlungen über einen Pfad von Zahlungskanälen vom Sender zum Empfänger geleitet werden.

Die ursprüngliche Idee des Lightning-Netzwerks wurde 2015 in der wegweisenden Arbeit »The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments« von Joseph Poon und Thaddeus Dryja vorgeschlagen. Im Jahr 2017 lief im Internet ein Lightning-»Test«-Netzwerk, in dem unterschiedliche Gruppen kompatible Implementierungen entwickelten und einige Kompatibilitätsstandards festlegten. 2018 ging das Lightning-Netzwerk »live«, und die Zahlungen begannen zu fließen.

Im Jahr 2019 vereinbarten Andreas M. Antonopoulos, Olaoluwa Osuntokun und René Pickhardt, beim Schreiben dieses Buchs zusammenzuarbeiten. Wie es scheint, mit Erfolg!

Leserkreis

Dieses Buch richtet sich hauptsächlich an ein technisches Publikum, das die Grundlagen von Bitcoin und anderen Blockchains versteht.

Verwendete Konventionen

Im Buch folgen wir diesen typografischen Konventionen:

Kursivschrift
: Wird für neue Begriffe, URLs, E-Mail-Adressen, Dateinamen und Dateierweiterungen verwendet.

`Nichtproportionalschrift`
: Wird für Programmlistings verwendet. Im normalen Fließtext werden damit Programmelemente wie Variablen- oder Funktionsnamen, Datenbanken, Datentypen, Umgebungsvariablen, Anweisungen und Schlüsselwörter hervorgehoben.

`Nichtproportionalschrift fett`
: Wird für Befehle oder andere Eingaben eingesetzt, die Sie wortwörtlich eingeben müssen.

`Nichtproportionalschrift kursiv`
: Wird für Text verwendet, der durch benutzereigene oder durch den Kontext bestimmte Werte ersetzt wird.

Mit diesem Symbol wird ein Tipp oder ein Vorschlag angezeigt.

Mit diesem Symbol wird ein allgemeiner Hinweis angezeigt.

Mit diesem Symbol wird eine Warnung angezeigt.

Codebeispiele

Die Beispiele sind in Go, C++ und Python geschrieben und verwenden die Kommandozeilen unixoider Betriebssysteme. Alle Code-Snippets finden Sie im GitHub-Repository im *code*-Unterverzeichnis. Laden Sie den Buchcode herunter, probieren Sie die Codebeispiele aus und senden Sie Korrekturen an: GitHub (*https://github.com/lnbook/lnbook*).

Alle Code-Snippets können für die meisten Betriebssysteme mit einer minimalen Installation der Compiler, Interpreter und Bibliotheken für die entsprechenden Sprachen repliziert werden. Wenn nötig, stellen wir grundlegende Installationsanweisungen und schrittweise Beispiele für die Ausgaben bereit.

Einige der Code-Snippets wurden für den Druck aufbereitet. In diesen Fällen wurden die Zeilen mit einem Backslash-Zeichen (\) gefolgt von einem Newline-Zeichen getrennt. Wenn Sie mit diesen Beispielen arbeiten, müssen Sie die beiden Zeichen entfernen und die Zeilen wieder zusammenfassen. Die Ergebnisse sollten dann denen der Beispiele entsprechen.

Alle Code-Snippets verwenden wann immer möglich reale Werte und Berechnungen. Sie können sich also von Beispiel zu Beispiel vorarbeiten und kommen immer zu den gleichen Ergebnissen wie das Buch. So sind beispielsweise die privaten Schlüssel und die dazugehörigen öffentlichen Schlüssel und Adressen alle echt.

Verwendung der Codebeispiele

Bei technischen Fragen oder Problemen mit den Codebeispielen senden Sie bitte eine E-Mail an *bookquestions@oreilly.com*.

Dieses Buch ist dazu gedacht, Ihnen bei der Erledigung Ihrer Arbeit zu helfen. Im Allgemeinen dürfen Sie den Code in diesem Buch in Ihren eigenen Programmen oder Dokumentationen verwenden. Solange Sie den Code nicht in großem Umfang reproduzieren, brauchen Sie uns nicht um Erlaubnis zu bitten. Der Verkauf oder Vertrieb von Beispielen aus O'Reilly-Büchern ist dagegen genehmigungspflichtig. Signifikante Teile von Beispielcode aus diesem Buch für die eigene Produktdokumentation zu verwenden, ist genehmigungspflichtig.

Wir freuen uns über eine Quellenangabe, verlangen sie aber nicht unbedingt. Zu einer Quellenangabe gehören normalerweise Autor, Titel, Verlagsangabe, Veröffentlichungsjahr und ISBN, hier also: »*Mastering the Lightning Network* by Andreas M. Antonopoulos, Olaoluwa Osuntokun, and René Pickhardt (O'Reilly). Copyright 2022 aantonop Books LLC, René Pickhardt, and uuddlrlrbas LLC, ISBN 978-1-492-05486-3«.

Mastering the Lightning Network wird unter der Creative Commons Attribution-Noncommercial-No Derivative Works 4.0 International License (CC BY-NC-ND 4.0) angeboten.

Sollten Sie befürchten, dass Ihre Verwendung der Codebeispiele gegen das Fairnessprinzip oder die Genehmigungspflicht verstoßen könnte, nehmen Sie bitte unter *permissions@oreilly.com* Kontakt mit uns auf.

Hinweise auf Unternehmen und Produkte

Alle Hinweise auf Unternehmen oder Produkte dienen der Information, Demonstration oder Referenz. Die Autoren unterstützen keines der genannten Unternehmen oder Produkte. Der Einsatz und die Sicherheit der in diesem Buch vorgestellten Produkte, Projekte oder Codefragmente wurden nicht getestet. Deren Nutzung erfolgt auf eigene Gefahr!

Adressen und Transaktionen in diesem Buch

Die Bitcoin-Adressen, Transaktionen, Schlüssel, QR-Codes und Blockchain-Daten in diesem Buch sind größtenteils echt. Sie können also die Blockchain durchgehen, sich die in den Beispielen enthaltenen Transaktionen genau ansehen und sie mit Ihren eigenen Skripten/Programmen abrufen.

Beachten Sie aber, dass die in diesem Buch zur Generierung von Adressen verwendeten privaten Schlüssel »verbrannt« wurden. Wenn Sie also Geld an diese Adressen senden, ist es für immer verloren, oder es kann von jedem abgeschöpft werden, der die hier abgedruckten privaten Schlüssel kennt.

Bitte senden Sie keinesfalls Geld an irgendeine der in diesem Buch verwendeten Adressen! Ihr Geld landet bei einem anderen Leser oder ist für immer verloren.

Andreas kontaktieren

Sie erreichen Andreas M. Antonopoulos über seine persönliche Website:
https://aantonop.com

Folgen Sie Andreas' Kanal auf YouTube:
https://www.youtube.com/aantonop

Folgen Sie Andreas' Seite auf Facebook:
https://www.facebook.com/AndreasMAntonopoulos

Folgen Sie Andreas auf Twitter:
https://twitter.com/aantonop

Folgen Sie Andreas auf LinkedIn:
https://linkedin.com/company/aantonop

Andreas möchte sich auch bei allen Förderern bedanken, die seine Arbeit durch monatliche Spenden unterstützen. Sie können Andreas auf Patreon unterstützen unter *https://patreon.com/aantonop*.

René kontaktieren

Sie erreichen René Pickhardt über seine persönliche Website:
https://ln.rene-pickhardt.de

Folgen Sie Renés Kanal auf YouTube:
https://www.youtube.com/user/RenePickhardt

Folgen Sie René auf Twitter:
https://twitter.com/renepickhardt

Folgen Sie René auf LinkedIn:
https://www.linkedin.com/in/rene-pickhardt-80313744

René möchte sich ebenfalls bei allen Förderern bedanken, die seine Arbeit durch eine monatliche Spende unterstützen. Sie können René auf Patreon unterstützen unter *https://patreon.com/renepickhardt*.

Sie können seine Arbeit aber auch direkt unter *https://donate.ln.rene-pickhardt.de* über das Lightning-Netzwerk in Bitcoin unterstützen. Dafür ist René ebenso dankbar wie seinen Förderern bei Patreon.

Olaoluwa Osuntokun kontaktieren

Sie erreichen Olaoluwa Osuntokun über seine berufliche E-Mail-Adresse: laolu@lightning.engineering

Folgen Sie Olaoluwa auf Twitter:
https://twitter.com/roasbeef

Danksagungen von Andreas

Meine Liebe zu Wörtern und Büchern verdanke ich meiner Mutter Theresa, die mich in einem Haus aufzog, in dem Bücher jede Wand mit Beschlag belegten. Meine Mutter kaufte mir 1982 auch meinen ersten Computer, obwohl sie sich selbst als »technophob« beschrieb. Mein Vater Menelaos, ein Bauingenieur, der sein erstes Buch mit 80 veröffentlichte, lehrte mich logisches und analytisches Denken und weckte meine Leidenschaft zu Wissenschaft und Technik.

Ich danke euch allen für eure Unterstützung während meiner Reise.

Danksagungen von René

Ich möchte dem deutschen Bildungssystem danken, dem ich das Wissen verdanke, auf dem meine Arbeit aufbaut. Es ist eines der größten mir gemachten Geschenke. Ebenso möchte ich dem deutschen Gesundheitswesen danken und allen Menschen, die in diesem Bereich arbeiten. Ihr Einsatz und ihr Durchhaltevermögen machen sie zu meinen persönlichen Helden, und ich werde nie die Hilfe, Aufmerksamkeit und Unterstützung vergessen, die mir zuteilwurde, als ich sie benötigte. Mein Dank geht an all die Studenten, denen ich etwas lehren durfte und die sich in interessanten Diskussionen und Fragen engagierten. Von ihnen habe ich das meiste gelernt. Ich bin auch der Bitcoin- und Lightning-Netzwerk-Community dankbar, die mich freundlich willkommen hieß, sowie den Enthusiasten und Privatpersonen, die meine Arbeit finanziell unterstützten und das auch weiterhin tun. Ich danke ganz besonders allen Open-Source-Entwicklern (nicht nur Bitcoin und dem Lightning-Netzwerk) und den Menschen, die sie finanzieren, um diese Technik möglich zu machen. Ein besonderer Dank an meine Mitautoren, die den Weg mit mir gegangen sind. Und nicht zuletzt danke ich meinen Liebsten.

Danksagungen von Olaoluwa Osuntokun

Ich möchte dem großartigen Team von Lightning Labs danken, ohne die es kein LND gäbe. Ich möchte auch der ursprünglichen Gruppe von Autoren der BOLT-Spezifikation danken: Rusty Russell, Fabrice Drouin, Conner Fromnkchet, Pierre-Marie Padiou, Lisa Neigut und Christian Decker. Nicht zuletzt möchte ich Joseph Poon und Tadge Dryja danken, den Autoren des ursprünglichen Lightning-Netzwerk-Papers, ohne die es kein Lightning-Netzwerk gäbe, über das man ein Buch schreiben kann.

Beitragende

Viele Beitragende lieferten Kommentare, Korrekturen und Ergänzungen zu diesem Buch, als es gemeinschaftlich auf GitHub geschrieben wurde.

Nachfolgend eine alphabetisch sortierte Liste aller GitHub-Beitragenden mit deren GitHub-IDs in Klammern:

- 8go (@8go)
- Aaqil Aziz (@batmanscode)
- Alexander Gnip (@quantumcthulhu)
- Alpha Q. Smith (@alpha_github_id)
- Ben Skee (@benskee)
- Brian L. McMichael (@brianmcmichael)
- CandleHater (@CandleHater)
- Daniel Gockel (@dancodery)
- Dapeng Li (@luislee818)
- Darius E. Parvin (@DariusParvin)
- Doru Muntean (@chriton)
- Eduardo Lima III (@elima-iii)
- Emilio Norrmann (@enorrmann)
- Francisco Calderón (@grunch)
- Francisco Requena (@FrankyFFV)
- François Degros (@fdegros)
- Giovanni Zotta (@GiovanniZotta)
- Gustavo Silva (@GustavoRSSilva)
- Guy Thayakorn (@saguywalker)
- Haoyu Lin (@HAOYUatHZ)
- Hatim Boufnichel (@boufni95)
- Imran Lorgat (@ImranLorgat)

- Jeffrey McLarty (@jnmclarty)
- John Davies (@tigeryant)
- Julien Wendling (@trigger67)
- Jussi Tiira (@juhi24)
- Kory Newton (@korynewton)
- Lawrence Webber (@lwebbz)
- Luigi (@gin)
- Maximilian Karasz (@mknoszlig)
- Omega X. Last (@omega_github_id)
- Owen Gunden (@ogunden)
- Patrick Lemke (@PatrickLemke)
- Paul Wackerow (@wackerow)
- Randy McMillan (@RandyMcMillan)
- René Köhnke (@rene78)
- Ricardo Marques (@RicardoM17)
- Sebastian Falbesoner (@theStack)
- Sergei Tikhomirov (@s-tikhomirov)
- Severin Alexander Bühler (@SeverinAlexB)
- Simone Bovi (@SimoneBovi)
- Srijan Bhushan (@srijanb)
- Taylor Masterson (@tjmasterson)
- Umar Bolatov (@bolatovumar)
- Warren Wan (@wlwanpan)
- Yibin Zhang (@z4y1b2)
- Zachary Haddenham (@senf42)

Ohne die Hilfe der oben aufgeführten Personen wäre dieses Buch nicht möglich gewesen. Eure Beiträge demonstrieren die Kraft von Open Source und einer offenen Kultur, und wir sind euch unendlich dankbar.

Vielen Dank.

Quellen

Ein Teil des Materials in diesem Buch stammt aus unterschiedlichen Public-Domain- bzw. Open-License-Quellen. Für andere Teile wurden Genehmigungen erteilt. Details zu Quellen, Lizenzen und Quellenzuordnungen finden Sie in Anhang D.

Vorwort zur deutschen Ausgabe

Als Satoshi Nakamoto Bitcoin 2009 veröffentlichte, dauerte es nicht lange, bis sich eine kleine Gruppe technisch Begeisterter zusammenfand, um die Vor- und Nachteile des neuen Systems zu diskutieren und damit herumzuspielen. Ich gehöre selbst auch zu den Leuten, die eher zufällig auf Bitcoin gestoßen sind, aber seitdem ich Bitcoin entdeckt habe, lässt es mich nicht mehr los.

Das Ziel, das sich Satoshi gesetzt hatte, war, dass Bitcoin ein globales Wertetransfer-Netzwerk werden sollte, ohne Mittelsmänner. In diesem Netzwerk sollten Werte in Form von Bitcoins beliebig hin und her verschoben werden können. Das Ganze durch pseudonyme Adressen ergänzt, um die Privatsphäre der Teilnehmer zu schützen.

Anfangs waren es noch wenige, die eher aus technischem Interesse mitmachten, denn damals wurden Bitcoins noch kein Wert zugeschrieben. Bereits zu diesem Zeitpunkt war aber schon klar, dass, sollte Bitcoin erfolgreich sein, wir schnell an die Grenzen des damaligen Systems stoßen würden. Deshalb mussten neue Ideen her.

Die wahrscheinlich schwierigste Frage war, wie Bitcoin denn skalieren könnte. Wie sollte Bitcoin sich also bei steigender Nachfrage anpassen, um allen Menschen die Möglichkeit zu geben, Bitcoin zu nutzen. Das Problem dabei war nämlich, dass die Blockchain, die von Satoshi für Bitcoin erfundene Technologie, ein geteiltes Medium darstellt, allerdings mit begrenzter Kapazität für das Bearbeiten von Transaktionen. Diese Frage stellten wir uns schon sehr früh, und 2012 war mir klar, dass ich meine Forschung im Rahmen meines Doktorats diesem Thema widmen würde.

Eine potenzielle Lösung waren sogenannte Micropayment Channels. Bei Micropayment Channels, auch Off-Chain-Protokolle genannt, handelt es sich um Systeme, die auf einer Blockchain aufbauen, um deren Nutzen zu erweitern. Bei Off-Chain-Protokollen werden Änderungen nicht mehr dem gesamten Netzwerk mitgeteilt, sondern zunächst nur den Teilnehmern, die an einer Transaktion beteiligt sind. Das gesamte Netzwerk wird erst später informiert. Die Saldierung aller stattfindenden Off-Chain-Transfers hat den Vorteil, dass am Ende nur eine einzige

Transaktion bestätigt werden muss. Und da für diese Bestätigung immer On-Chain-Gebühren anfallen, ist es viel kostengünstiger, wenn eine On-Chain-Gebühr auf beliebig viele Off-Chain-Transfers verteilt werden kann.

Was anfangs noch sehr abstrakt war, wurde durch Experimente in dieser Richtung immer konkreter: angefangen mit den einfachen unidirektionalen Channels von Matt Corallo und Jeremy Spillmann bis hin zu dem Lightning Network Paper von Joseph Poon und Tadge Dryja.

Als Joseph und Tadge 2015 das Lightning Network Paper publizierten, stieß es auf großes Interesse, gerade bei denjenigen von uns, denen der Mangel einer plausiblen Lösung für die Skalierung von Bitcoin unter den Nägeln brannte. Aber Skalierbarkeit sollte nicht der einzige Vorteil des Lightning Network sein, hinzu kommt auch, dass es in Echtzeit Zahlungen ermöglicht und die Privatsphäre potenziell besser schützen kann, weil nicht mehr alles bis in alle Ewigkeit auf der Blockchain gespeichert wird.

Doch die Vorteile brachten wiederum einiges an Komplexität und neuen Konzepten mit sich, die die Nutzer der Technologie erst einmal kennenlernen und verstehen müssen. Beim Lightning-Netzwerk muss man sowohl Bitcoin als auch die Konzepte hinter Lightning verstehen, wodurch der Einstieg alles andere als einfach ist.

Hinzu kommt, dass das Paper an sich sehr komplex ist und eher ein abstraktes Bild als ein voll funktionsfähiges System beschreibt. Es fehlten alle technischen Details, bis auf das Grundgerüst in Form der Bitcoin-Transaktionen.

So passierte erst mal nichts, bis Rusty Russell von Blockstream sich des Problems annahm und anfing, die fehlenden Stellen auszuschmücken, denn eines war uns klar, dieses System musste unbedingt real werden. Kurz darauf fingen dann auch die Kollegen von Acinq und Lightning Labs an, eine Implementierung zu bauen. Im Herbst 2016 entschieden die drei Teams dann zusammenzuarbeiten, um ein einziges großes Netz zu bauen, in dem jeder jedem anderen Teilnehmer Bitcoins senden konnte: schnell und anonym. Und so fing die Lightning Network Specification an, ein Prozess, der bis heute anhält.

Dieses Buch wendet sich an alle, die die Hintergründe hinter dem Lightning Netzwerk interessieren, ob das nun ein Nutzer ist, eine Entwicklerin, die auf Lightning aufbauen will, oder vielleicht sogar ein zukünftiger Protokollentwickler. Es bildet das fehlende Bindeglied zwischen dem Paper von 2015 und dem Protokoll, so wie es heute aktiv im Netzwerk genutzt wird, und die Leserinnen und Leser werden langsam und schrittweise an das komplexe Thema herangeführt.

Bei den Autoren handelt es sich um echte Heavyweights der Bitcoin Community. Andreas Antonopoulos ist langjähriger Speaker und hat die unglaublich wertvolle Fähigkeit, komplexe Themen anschaulich zu vermitteln. Olaoluwa Osuntokun, Entwickler der ersten Stunde, bringt die praktische Erfahrung und das notwendige Detailwissen mit und René Pickhardt den theoretischen Hintergrund, um das Protokoll abstrakt zu behandeln.

Ein solches Buch zu schreiben, ist nicht einfach, denn es handelt sich beim Lightning-Netzwerk um ein bewegliches Ziel, und so beleuchtet das Buch sowohl Konzepte als auch deren konkrete Umsetzungen.

Es handelt sich bei diesem Buch also um ein Nachschlagewerk, über das ich mich damals, als wir das Protokoll entwickelt haben, sehr gefreut hätte.

Dr. Christian Decker
Researcher, Blockstream