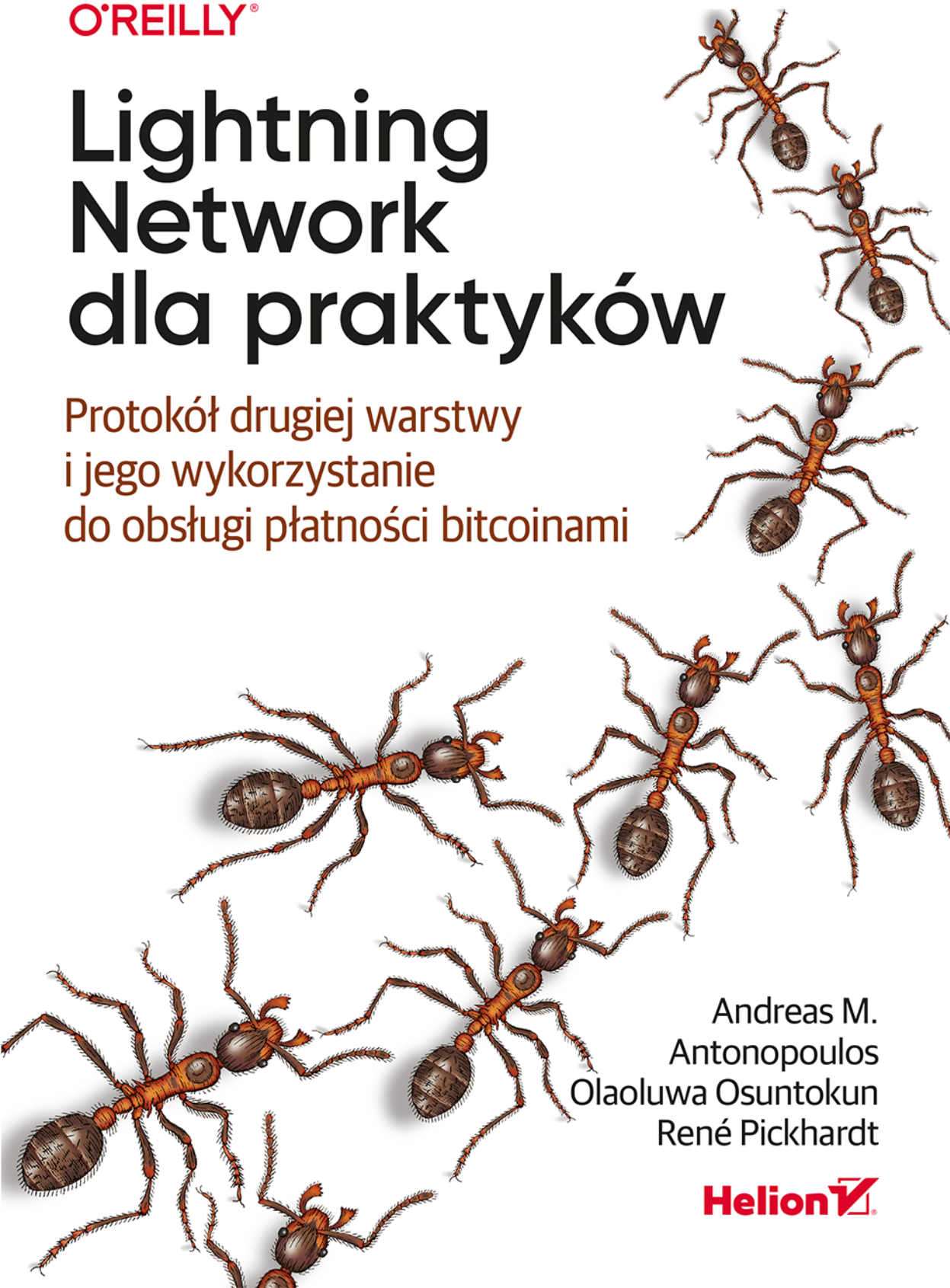


O'REILLY®

# Lightning Network dla praktyków

Protokół drugiej warstwy  
i jego wykorzystanie  
do obsługi płatności bitcoinami



Andreas M.  
Antonopoulos  
Olaoluwa Osuntokun  
René Pickhardt

Helion 

Tytuł oryginału: Mastering the Lightning Network: A Second Layer Blockchain Protocol  
for Instant Bitcoin Payments

Tłumaczenie: Radosław Meryk

ISBN: 978-83-283-9322-6

© 2023 Helion S.A.

Authorized Polish translation of the English *Mastering the Lightning Network* ISBN 9781492054863  
© 2022 aantopn Books LLC, René Pickhardt, and uuddlrbrbas LLC.

This translation is published and sold by permission of O'Reilly Media, Inc., which owns or controls all rights to publish and sell the same.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz wydawca dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz wydawca nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<https://helion.pl/user/opinie/linepr>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Helion S.A.

ul. Kościuszki 1c, 44-100 Gliwice

e-mail: [helion@helion.pl](mailto:helion@helion.pl)

WWW: <https://helion.pl> (księgarnia internetowa, katalog książek)

Printed in Poland.

- Kup książkę
- Poleć książkę
- Oceń książkę

- Księgarnia internetowa
- Lubię to! » Nasza społeczność

Przedmowa .....	15
-----------------	----

---

## Część I. Podstawy Lightning Network

<b>1. Wprowadzenie .....</b>	<b>25</b>
Lightning Network — podstawowe pojęcia	25
Zaufanie w sieciach zdecentralizowanych	27
Uczciwość bez organu centralnego	28
Protokoły zaufania bez pośredników	28
Protokół uczciwości w działaniu	29
Prymitywy zabezpieczeń jako bloki konstrukcyjne	30
Przykład protokołu uczciwości	31
Sens istnienia sieci Lightning Network	32
Skalowanie łańcuchów bloków	33
Cechy charakterystyczne Lightning Network	35
Przypadki użycia Lightning Network, użytkownicy i ich historie	35
Podsumowanie	36
<b>2. Pierwsze kroki .....</b>	<b>37</b>
Pierwszy portfel Lightning Alicji	37
Węzły Lightning	38
Eksploratory Lightning	38
Portfele Lightning	39
Bitcoin Testnet	42
Równoważenie złożoności i kontroli	42
Pobieranie i instalowanie portfela Lightning	43
Tworzenie nowego portfela	44
Odpowiedzialność związana z przechowywanymi kluczami	44
Słowa mnemoniczne	45
Bezpieczne przechowywanie frazy mnemonicznej	46

---

Ładowanie bitcoinów do portfela	46
Zdobywanie bitcoinów	47
Odbieranie bitcoinów	47
Od sieci Bitcoin do Lightning Network	50
Kanały sieci Lightning Network	50
Otwieranie kanału Lightning	53
Kupowanie filiżanki kawy za pomocą Lightning Network	54
Kawiarnia u Bogdana	55
Rachunek Lightning	55
Podsumowanie	57
<b>3. Jak działa sieć Lightning Network? .....</b>	<b>58</b>
Czym jest kanał płatności?	59
Podstawowe informacje o kanale płatności	59
Routowanie płatności między kanałami	60
Kanały płatności	61
Adres wielopodpisowy	62
Transakcja finansowania	62
Transakcja zobowiązania	63
Oszukiwanie z wykorzystaniem poprzedniego stanu	65
Ogłaszanie kanału	67
Zamykanie kanału	68
Faktury	72
Skrót płatności i preobraz	73
Dodatkowe metadane	73
Dostarczanie płatności	74
Protokół plotkarski peer-to-peer	74
Znajdowanie ścieżek i wyznaczanie tras	76
Znajdowanie ścieżek oparte na źródle	76
Routing cebulowy	77
Algorytm przekazywania płatności	79
Szyfrowanie komunikacji peer-to-peer	80
Garść uwag o zaufaniu	81
Porównanie z Bitcoinem	81
Adresy kontra faktury, transakcje kontra płatności	81
Wybieranie wyjść a znajdowanie ścieżki	82
Zmiana wyjść w sieci Bitcoin kontra brak zmian w sieci Lightning	82
Opłaty za wydobycie a opłaty za routing	83
Różne opłaty w zależności od ruchu w porównaniu z opłatami ogłaszanymi	83
Publiczne transakcje Bitcoin kontra prywatne płatności Lightning	84

Oczekiwanie na potwierdzenie kontra natychmiastowe rozliczenie	84
Wysyłanie dowolnych kwot kontra ograniczenia pojemności	85
Zachęta do płatności o dużej wartości kontra zachęta do płatności o małej wartości	85
Korzystanie z blockchajna jako księgi kontra „system sądowy”	85
Offline kontra online, operacje asynchroniczne kontra synchroniczne	86
Satoshi kontra milisatoshi	86
Cechy wspólne sieci Bitcoin i Lightning Network	87
Jednostka monetarna	87
Nieodwracalność i ostateczność płatności	87
Zaufanie i ryzyko kontrahenta	87
Operacje w trybie „bez uprawnień”	87
Open Source i Open System	88
Podsumowanie	88
<b>4. Programowanie węzłów Lightning .....</b>	<b>89</b>
Środowisko programistyczne Lightning	90
Korzystanie z wiersza polecenia	90
Pobieranie repozytorium książki	91
Kontenery Docker	91
Bitcoin Core i Regtest	93
Budowanie kontenera Bitcoin Core	93
Projekt c-lightning	96
Budowanie c-lightning jako kontenera Docker	96
Konfiguracja sieci Docker	97
Uruchamianie kontenerów bitcoind i c-lightning	97
Instalowanie c-lightning z kodu źródłowego	99
Instalowanie wymaganych bibliotek i pakietów	99
Kopiowanie kodu źródłowego c-lightning	100
Kompilowanie kodu źródłowego c-lightning	100
Projekt węzła Lightning Network Daemon	102
Kontener Docker węzła LND	102
Uruchamianie kontenerów bitcoind i LND	103
Instalowanie kontenera LND z kodu źródłowego	104
Kopiowanie kodu źródłowego LND	105
Kompilowanie kodu źródłowego LND	106
Projekt węzła Eclair Lightning	106
Kontener Docker serwera Eclair	106
Uruchamianie kontenerów bitcoind i Eclair	107
Instalowanie serwera Eclair z kodu źródłowego	109
Kopiowanie kodu źródłowego Eclair	109
Kompilowanie kodu źródłowego Eclair	109

Budowanie kompletnej sieci z różnych węzłów Lightning	110
Korzystanie z funkcji docker-compose do orkiestracji kontenerów Docker	111
Konfigurowanie polecenia docker-compose	111
Uruchamianie przykładu Lightning Network	112
Otwieranie kanałów i routowanie płatności	113
Podsumowanie	115
<b>5. Obsługa węzła sieci Lightning Network .....</b>	<b>116</b>
Wybór platformy	117
Dlaczego ważna jest niezawodność działania węzła Lightning?	117
Rodzaje sprzętu dla węzłów Lightning	117
Utrzymywanie węzła w chmurze	118
Uruchamianie węzła w warunkach domowych	118
Jaki sprzęt jest wymagany do uruchomienia węzła Lightning?	119
Przełączanie konfiguracji serwerowej w chmurze	120
Korzystanie z instalatora lub pomocnika	121
RaspiBlitz	122
Mynode	123
Umbrel	123
BTCPay Server	124
Węzeł Bitcoin czy węzeł Lightning w trybie uproszczonym?	125
Wybór systemu operacyjnego	126
Wybór implementacji węzła Lightning	126
Instalowanie węzła Bitcoin lub Lightning	127
Usługi w tle	128
Izolacja procesu	128
Uruchamianie węzła	129
Konfiguracja węzła	130
Konfiguracja sieci	131
Bezpieczeństwo węzła	136
Bezpieczeństwo systemu operacyjnego	136
Dostęp do węzłów	137
Kopie zapasowe węzłów i kanałów	138
Zagrożenia dla gorących portfeli	140
„Wymiatanie” funduszy	140
Czas pracy i dostępność węzła Lightning	142
Tolerowanie błędów i automatyzacja	143
Monitorowanie dostępności węzłów	144
Monitorowanie dostępności węzłów	144
Wieże strażnicze	144

Zarządzanie kanałami	146
Otwieranie kanałów wychodzących	146
Osiąganie płynności przychodzącej	149
Zamykanie kanałów	150
Rebalancing kanałów	150
Opłaty za routing	152
Zarządzanie węzłami	153
Ride The Lightning	153
lndmon	154
ThunderHub	154
Podsumowanie	154

---

## Część II. Lightning Network w szczegółach

<b>6. Architektura Lightning Network .....</b>	<b>157</b>
Zestaw protokołów Lightning Network Protocol Suite	157
Lightning w szczegółach	158
<b>7. Kanały płatności .....</b>	<b>160</b>
Inny sposób korzystania z systemu Bitcoin	161
Własność sieci Bitcoin i zarządzanie nią	162
Różnorodność (niezależnej) własności i wielopodpisowość	162
Współwłasność bez niezależnej kontroli	162
Niedopuszczenie do zablokowania bitcoinów i przekształcenia ich w niemożliwe do wydania	163
Konstruowanie kanału płatności	163
Prywatne i publiczne klucze węzłów	163
Sieciowy adres węzła	164
Identyfikator węzłów	164
Łączenie węzłów jako bezpośrednich partnerów	165
Budowanie kanału	165
Protokół partnerski do zarządzania kanałem	165
Przepływ komunikatów podczas ustanawiania kanału	165
Transakcja finansowania	169
Generowanie adresu wielopodpisowego	169
Konstruowanie transakcji finansowania	169
Utrzymywanie podpisanych transakcji bez ich publikowania	170
Refundacja przed finansowaniem	170
Konstruowanie wstępnie podpisanej transakcji zwrotu	171
Łączenie transakcji w łańcuch bez publikowania	171
Rozwiązywanie problemu plastyczności (Segregated Witness)	172
Publikowanie transakcji finansowania	174

Wysyłanie płatności przez kanał	175
Podział salda	175
Konkurencyjne zobowiązania	176
Oszustwa z wykorzystaniem przedawnionych transakcji zobowiązania	177
Odwoływanie nieaktualnych transakcji zobowiązania	177
Asymetryczne transakcje zobowiązania	178
Opóźnione (zablokowane w czasie) wydatki to_self	179
Klucze odwołania	180
Transakcja zobowiązania	180
Zmiany stanu kanału	182
Komunikat commitment_signed	183
Komunikat revoke_and_ack	184
Odwoływanie i ponowne zobowiązanie	184
Oszukiwanie i karanie w praktyce	185
Rezerwacja kanału: zapewnienie ryzyka w grze	188
Zamykanie kanału (zamknięcie wzajemne)	188
Komunikat shutdown	189
Komunikat closing_signed	189
Transakcja wzajemnego zamknięcia	190
Podsumowanie	191
<b>8. Routing w sieci kanałów płatności .....</b>	<b>192</b>
Routowanie płatności	192
Routing a znajdowanie ścieżek	194
Tworzenie sieci kanałów płatności	194
Fizyczny przykład „routingu”	195
Protokół uczciwości	200
Implementacja atomowych, wieloprzeskokowych płatności w trybie „bez zaufania”	201
Wracamy do przykładu napiwków	201
Rozliczanie kontraktów HTLC w trybie on-chain a rozliczanie ich w trybie off-chain	203
Kontrakty HTLC	203
Kontrakty HTLC w Bitcoin Script	204
Preobraz płatności i weryfikacja skrótu	205
Rozszerzanie kontraktu HTLC od Alicji do Darii	206
Wsteczne propagowanie sekretu	207
Wiązanie podpisów: zabezpieczenia kontraktów HTLC przed kradzieżą	209
Optymalizacja skrótów	210
Kooperytywne kontrakty HTLC i błędy limitu czasu	212
Zmniejszanie blokad czasowych	213
Podsumowanie	213



<b>9. Obsługa kanału i przekazywanie płatności .....</b>	<b>215</b>
Lokalne (jednokanałowe) a routowane (wielokanałowe)	216
Przekazywanie płatności i aktualizacja zobowiązań za pomocą kontraktów HTLC	216
Przepływ komunikatów związanych z HTLC i transakcjami zobowiązań	216
Przekazywanie płatności z wykorzystaniem kontraktów HTLC	217
Dodawanie kontraktu HTLC	217
Komunikat update_add_HTLC	218
Kontrakty HTLC w transakcjach zobowiązania	219
Nowe zobowiązanie z wyjściem HTLC	220
Zobowiązania Alicji	221
Bogdan uznaje nowe zobowiązanie i odwołuje stare	222
Zobowiązania Bogdana	224
Wielokrotne kontrakty HTLC	225
Realizacja kontraktu HTLC	226
Propagacja HTLC	226
Daria wypełnia kontrakt HTLC z Chanem	226
Bogdan rozlicza HTLC z Alicją	227
Usuwanie kontraktu HTLC z powodu błędu lub wygaśnięcia ważności	230
Wykonywanie płatności lokalnej	231
Podsumowanie	232
<b>10. Routing cebulowy .....</b>	<b>233</b>
Fizyczny przykład ilustrujący routing cebulowy	234
Wybieranie ścieżki	234
Budowanie warstw	235
Obieranie warstw	237
Wprowadzenie do routingu cebulowego opartego na kontraktach HTLC	238
Alicja wybiera ścieżkę	238
Alicja konstruuje ładunki kanału	240
Generowanie kluczy	243
„Owijanie” warstw cebuli	247
Cebule o stałej długości	247
Owijanie cebuli (zarys)	247
Owijanie ładunku przeskoku Darii	249
Opakowywanie ładunku przeskoku Chana	253
Opakowywanie ładunku przeskoku Bogdana	254
Końcowa postać pakietu cebuli	255
Wysyłanie cebuli	256
Komunikat update_add_htlc	256
Alicja wysyła cebulę do Bogdana	256
Bogdan sprawdza cebulę	257
Bogdan generuje wypełniacz	257
Bogdan usuwa zaciemnienie swojego ładunku przeskoku	257

Bogdan wydobywa zewnętrzny skrót HMAC dla następnego przeskoku	258
Bogdan usuwa swój ładunek i przesuwa zawartość cebuli w lewo	259
Bogdan konstruuje nowy pakiet cebuli	260
Bogdan weryfikuje szczegóły kontraktu HTLC	260
Bogdan wysła komunikat update_add_htlc do Chana	260
Chan przekazuje cebulę	261
Daria otrzymuje ostateczny ładunek	261
Zwracanie błędów	262
Komunikaty o błędach	263
Spontaniczne płatności Keysend	265
Niestandardowe rekordy TLV cebuli	265
Wysyłanie i odbieranie płatności Keysend	266
Płatności Keysend i niestandardowe rekordy w aplikacjach Lightning	266
Podsumowanie	266
<b>11. Plotki i graf kanałów .....</b>	<b>267</b>
Odkrywanie węzłów partnerskich	269
Bootstrapping w sieci P2P	270
Bootstrapping DNS	270
Opcje zapytań SRV	274
Graf kanałów	275
Graf skierowany	275
Komunikaty protokołu plotkarskiego	276
Komunikat node_announcement	276
Komunikat channel_announcement	278
Komunikat channel_update	282
Bieżące utrzymywanie grafu kanałów	283
Podsumowanie	283
<b>12. Znajdowanie ścieżek i dostarczanie płatności .....</b>	<b>284</b>
Znajdowanie ścieżek w pakiecie protokołów Lightning Protocol Suite	284
Gdzie jest BOLT?	285
Znajdowanie ścieżek. Jaki problem rozwiązujemy?	285
Wybieranie najlepszej ścieżki	286
Znajdowanie ścieżek w matematyce i informatyce	286
Pojemność, saldo, płynność	287
Niepewność sald	287
Złożoność znajdowania ścieżek	288
Zachowaj prostotę	289
Znajdowanie ścieżek i dostarczanie płatności	289
Konstrukcja grafu kanałów	290
Niepewność płynności i prawdopodobieństwo	293
Opłaty i inne metryki kanałów	294

Znajdowanie ścieżek kandydatów	295
Dostarczanie płatności (pętla prób i błędów)	296
Pierwsza próba (ścieżka nr 1)	296
Druga próba (ścieżka nr 4)	297
Płatności wieloczęściowe	298
Korzystanie z płatności MPP	299
Próby i błędy w wielu „rundach”	300
Podsumowanie	302
<b>13. Protokół łącza fizycznego: tworzenie ramek i rozszerzalność .....</b>	<b>303</b>
Warstwa komunikatów w pakiecie protokołów Lightning Protocol Suite	303
Tworzenie ramek łącza fizycznego	304
Wysokopoziomowe tworzenie ramki łącza fizycznego	304
Kodowanie typów	305
Rozszerzenia komunikatów „Typ-Długość-Wartość”	306
Format komunikatów Protobuf	306
Zgodność wstecz i w przód	306
Format „Typ-Długość-Wartość”	307
Kodowanie z wykorzystaniem typu BigSize	307
Ograniczenia kodowania TLV	308
Kodowanie kanoniczne TLV	308
Bity funkcji i rozszerzalność protokołu	309
Bity funkcji jako mechanizm wykrywania uaktualnień	309
TLV dla zgodności wstecz i w przód	310
Taksonomia mechanizmów aktualizacji	311
Uaktualnienia „od końca do końca”	311
Aktualizacje na poziomie budowy kanału	312
Podsumowanie	312
<b>14. Szyfrowany transport komunikatów w Lightning Network .....</b>	<b>313</b>
Szyfrowany transport w pakiecie protokołów Lightning Protocol Suite	313
Wprowadzenie	314
Graf kanałów jako zdecentralizowana infrastruktura klucza publicznego	314
Dlaczego nie TLS?	315
Framework protokołów Noise	315
Szyfrowany transport Lightning w szczegółach	316
Noise_XK: uzgadnianie w Lightning Network	316
Notacja uzgadniania i przepływ protokołu	316
Ogólny przebieg protokołu	317
Uzgadnianie w trzech aktach	318
Podsumowanie	326

<b>15. Żądania płatności w sieci Lightning .....</b>	<b>327</b>
Faktury w pakiecie protokołów Lightning Protocol Suite	327
Wprowadzenie	327
Żądania płatności Lightning a adresy Bitcoin	328
BOLT nr 11: serializacja i interpretacja żądania płatności w Lightning	329
Kodowanie żądań płatności w praktyce	329
Prefiks czytelny dla człowieka	329
bech32 i segment danych	330
Podsumowanie	332
<b>16. Bezpieczeństwo i prywatność w Lightning Network .....</b>	<b>333</b>
Dlaczego prywatność jest ważna?	333
Definicje prywatności	333
Proces oceny prywatności	334
Zbiór anonimowości	335
Różnice między Lightning Network a Bitcoinem w kontekście prywatności	336
Ataki na sieć Lightning	338
Obserwowanie kwot płatności	338
Łączenie nadawców z odbiorcami	338
Ujawnianie sald kanałów (sondowanie)	339
Ataki DoS	341
Zagłuszanie zobowiązań	343
Blokada płynności kanału	343
Dezanonimizacja obejmująca wiele warstw	343
Klasteryzacja podmiotów on-chain Bitcoina	344
Klasteryzacja węzłów off-chain w sieci Lightning Network	345
Łączenie obejmujące wiele warstw: węzły Lightning i podmioty Bitcoin	345
Graf sieci Lightning	346
Jak naprawdę wygląda graf sieci Lightning?	346
Centralizacja w sieci Lightning Network	349
Zachęty ekonomiczne a struktura grafu	349
Praktyczne porady co do ochrony prywatności użytkowników	350
Kanały nieogłoszone	350
Zagadnienia dotyczące routingu	351
Akceptowanie kanałów	351
Podsumowanie	352
Bibliografia i dalsza lektura	353

<b>17. Podsumowanie .....</b>	<b>354</b>
Innowacje asynchroniczne i zdecentralizowane	354
Usprawnienia protokołu Bitcoin i języka Bitcoin Script	355
Innowacje w protokole Lightning	355
Rozszerzalność TLV	356
Konstrukcja kanału płatności	356
Funkcje opt-in	356
Aplikacje Lightning (LApps)	357
Gotowi, do startu, start!	357
<b>A. Podstawy Bitcoina — przegląd .....</b>	<b>359</b>
<b>B. Podstawowa instalacja i użytkowanie platformy Docker .....</b>	<b>377</b>
<b>C. Komunikaty protokołu komunikacyjnego .....</b>	<b>380</b>
<b>D. Źródła i informacje o licencjach .....</b>	<b>395</b>
<b>Glosariusz .....</b>	<b>397</b>
<b>Skorowidz .....</b>	<b>414</b>



# Pierwsze kroki

W tym rozdziale zaczniemy od tego, od czego zaczyna większość osób, gdy po raz pierwszy spotyka się z Lightning Network — wyboru oprogramowania do korzystania z ekonomii LN. Przeanalizujemy wybory dwóch użytkowników reprezentujących powszechny przypadek użycia Lightning Network i będziesz uczyć się na ich przykładzie. Alicja, klientka kawiarni, będzie używać portfela Lightning na swoim urządzeniu mobilnym, aby płacić za kawę z „Kawiarni u Bogdana”. Bogdan, handlowiec, będzie używał węzła Lightning i portfela do utrzymywania systemu punktów sprzedaży w swojej kawiarni, dzięki czemu będzie mógł akceptować płatności za pośrednictwem Lightning Network.

## Pierwszy portfel Lightning Alicji

Alicja od dawna korzysta z sieci Bitcoin. Po raz pierwszy spotkaliśmy się z nią w rozdziale 1. książki *Mastering Bitcoin*<sup>1</sup>, gdy za pomocą transakcji Bitcoin kupowała filiżankę kawy w kawiarni u Bogdana. Jeśli jeszcze nie wiesz, jak działają transakcje Bitcoin, lub potrzebujesz przypomnienia, przeczytaj książkę *Mastering Bitcoin* lub podsumowanie w dodatku A.

Alicja niedawno się dowiedziała, że w „Kawiarni u Bogdana” zaczęto akceptować płatności LN! Alicja bardzo by chciała poznać technologię Lightning Network i poeksperymentować z nią. Postanowiła stać się jednym z pierwszych klientów Bogdana korzystających z LN. Aby to zrobić, Alicja musi najpierw wybrać portfel Lightning, który spełnia jej potrzeby.

Alicja nie chce powierzać opieki nad swoimi bitcoinami podmiotom zewnętrznym. Dowiedziała się o kryptowalutach wystarczająco dużo, aby wiedzieć, w jaki sposób korzystać z portfela. Potrzebuje również portfela mobilnego, umożliwiającego dokonywanie niewielkich płatności w podróży. Dlatego wybrała portfel Eclair, popularny *niewierniczy* (ang. *non custodial*) mobilny portfel Lightning. Zaraz powiemy więcej o tym, jak i dlaczego dokonała takiego wyboru.

---

<sup>1</sup> Andreas M. Antonopoulos, *Mastering Bitcoin*, wydanie drugie, O'Reilly, rozdział 1. (<https://github.com/bitcoin-book/bitcoin-book/blob/develop/ch01.asciidoc>).

# Węzły Lightning

Dostęp do sieci Lightning Network można uzyskać za pośrednictwem aplikacji, które mogą się komunikować za pomocą protokołu LN. Węzeł Lightning Network (zwany również węzłem LN lub po prostu węzłem) to aplikacja, która ma trzy ważne funkcje. Po pierwsze, węzły Lightning są portfelami, więc wysyłają i odbierają płatności za pośrednictwem sieci Lightning Network, a także w sieci Bitcoin. Po drugie, węzły muszą się komunikować z innymi węzłami Lightning tworzącymi sieć na zasadach *peer-to-peer*. Na koniec, węzły Lightning potrzebują dostępu do sieci blockchain Bitcoina (lub sieci blockchain innych kryptowalut) w celu zabezpieczenia środków używanych w płatnościach.

Użytkownicy mogą uzyskać najwyższy stopień kontroli dzięki uruchomieniu własnego węzła Bitcoin i węzła Lightning. Do interakcji z siecią blockchain Bitcoin węzły Lightning mogą jednak również korzystać z prostego klienta Bitcoin, powszechnie określanego jako SPV (ang. *simplified payment verification*).

## Eksploratory Lightning

Eksploratory LN są przydatnymi narzędziami do wyświetlania statystyk węzłów, kanałów oraz monitorowania możliwości sieci.

Oto kilka przykładów eksploratorów Lightning:

- 1ML (<https://1ml.com>),
- ACINQ (<https://explorer.acinq.co>), z wyszukanyimi wizualizacjami,
- Amboss Space (<https://amboss.space>), z metrykami społeczności i intuicyjnymi wizualizacjami,
- Fiatjaf (<https://ln.bigsun.xyz>), z wieloma wykresami,
- hashXP (<https://hashxp.org/lightning/node>).



Należy pamiętać, że podczas korzystania z eksploratorów Lightning, podobnie jak innych eksploratorów bloków, może dojść do naruszeń prywatności. Jeśli użytkownicy są nieostrożni, witryna może śledzić ich adresy IP i zbierać zapis ich zachowania (na przykład uzyskiwać informacje o węzłach, którymi użytkownicy są zainteresowani).

Należy również zapamiętać, że ze względu na brak zgody co do obecnego grafu sieci Lightning lub obecnego stanu polityki kanału użytkownicy nie powinni polegać na eksploratorach Lightning w uzyskiwaniu możliwie aktualnych informacji. Co więcej, gdy użytkownicy otwierają, zamykają i aktualizują kanały, graf się zmienia, a poszczególne eksploratory Lightning mogą być nieaktualne. Korzystaj z eksploratorów Lightning do wizualizacji sieci lub zbierania informacji, ale nie stosuj ich jako autorytatywnego źródła informacji na temat tego, co się dzieje w sieci Lightning Network. Aby uzyskać autorytatywny widok sieci Lightning Network, uruchom własny węzeł Lightning, który zbuduje graf kanału i zbierze różne statystyki dostępne do przeglądania w interfejsie webowym.



# Portfele Lightning

Termin *portfel Lightning* jest nieco mylący, ponieważ może opisywać szeroką gamę komponentów w połączeniu z wybranymi interfejsami użytkownika. Do najbardziej powszechnych składników oprogramowania portfela Lightning należą:

- magazyn kluczy zawierający poufne informacje (ang. *secrets*), takie jak klucze prywatne,
- węzeł LN (węzeł Lightning), który komunikuje się w sieci *peer-to-peer*, tak jak opisano powyżej,
- węzeł Bitcoin, który przechowuje dane sieci blockchain i komunikuje się z innymi węzłami Bitcoin,
- „mapa” bazy danych i kanałów, które są rozgłaszane w sieci Lightning Network,
- menedżer kanałów, który może otwierać i zamykać kanały LN,
- system „przybliżania” (ang. *close-up*), który pozwala znaleźć ścieżkę połączonych kanałów od źródła płatności do jej miejsca docelowego.

Portfel Lightning może zawierać wszystkie te funkcje. Dzięki temu może być wykorzystywany jako „pełnoprawny” portfel, niewymagający korzystania z jakichkolwiek usług zewnętrznych. Jeden lub większa liczba tych składników może polegać (częściowo lub całkowicie) na usługach innych firm, pośredniczących w dostarczaniu potrzebnych funkcji.

Kluczową cechą wyróżniającą jest to, czy funkcja magazynu kluczy jest wewnętrzna czy zewnętrzna. W sieciach typu blockchain zarządzanie kluczami określa opiekę nad funduszami, co upamiętnia fraza „twoje klucze, twoje monety, nie twoje klucze, nie twoje monety”. Każdy portfel, w którym zarządzanie kluczami zostało zlecone na zewnątrz, jest nazywany *powierniczym* (ang. *custodial*), ponieważ to podmiot zewnętrzny, a nie użytkownik, ma kontrolę nad funduszami użytkownika. Dla porównania, portfel *niewierniczy* (ang. *noncustodial* lub *self-custody*) to taki, w którym magazyn kluczy jest częścią portfela, a klucze są zarządzane bezpośrednio przez użytkownika. Termin portfel niewierniczy oznacza, że magazyn kluczy jest lokalny i pod kontrolą użytkownika. Jednak jeden lub więcej innych komponentów portfela może, ale nie musi, pochodzić z zewnątrz i polegać na zaufanych podmiotach zewnętrznych.

Sieci blockchain, a zwłaszcza otwarte sieci blockchain, takie jak Bitcoin, próbują zminimalizować lub wyeliminować zaufanie do podmiotów zewnętrznych i wzmocnić pozycję użytkowników. Taki model działania często jest nazywany modelem „niezaufanym” (ang. *trustless*), chociaż lepszym określeniem byłoby „model o minimalnej potrzebie zaufania” (ang. *trust-minimized*). W takich systemach użytkownik ufa regułom oprogramowania, a nie podmiotom zewnętrznym. Dlatego przy wyborze portfela Lightning głównym czynnikiem jest kwestia kontroli nad kluczami.

Do wszystkich innych elementów portfela Lightning mają zastosowanie podobne względy zaufania. Jeśli wszystkie komponenty są pod kontrolą użytkownika, wtedy stopień zaufania do podmiotów zewnętrznych jest zminimalizowany, co daje użytkownikom maksymalne możliwości. Oczywiście jest z tym związany bezpośredni kompromis, ponieważ z tymi możliwościami wiąże się odpowiedzialność za zarządzanie złożonym oprogramowaniem.

Każdy użytkownik przed podjęciem decyzji o wyborze rodzaju portfela Lightning musi wziąć pod uwagę własne umiejętności techniczne. Osoby z obszerną wiedzą techniczną powinny korzystać z portfela Lightning, w którym wszystkie komponenty znajdują się pod bezpośrednią kontrolą użytkownika. Osoby, które mają mniejsze umiejętności techniczne, ale chcą kontrolować własne fundusze, powinny wybrać niepowierniczy portfel Lightning. Często w takich przypadkach zaufanie jest powiązane z prywatnością. Jeśli użytkownicy decydują się zlecić niektóre funkcje podmiotom zewnętrznym, zwykle rezygnują z prywatności, ponieważ podmioty zewnętrzne uzyskają o nich pewne informacje.

Na koniec, osoby poszukujące prostoty i wygody, nawet kosztem kontroli i bezpieczeństwa, powinny wybrać powierniczy portfel Lightning. Jest to najmniej trudna technicznie opcja, ale *podważa model zaufania kryptowaluty* i dlatego powinna być traktowana jedynie jako krok w kierunku uzyskania większej kontroli i samodzielności.

Istnieje wiele sposobów, na jakie można charakteryzować lub kategoryzować portfele. Oto najważniejsze pytania na temat konkretnego portfela, które należy zadać:

1. Czy portfel Lightning zawiera kompletny węzeł Lightning, czy też używa węzła Lightning firmy zewnętrznej?
2. Czy portfel Lightning zawiera kompletny węzeł Bitcoin, czy też korzysta z węzła Bitcoin za pośrednictwem innej firmy?
3. Czy portfel Lightning przechowuje własne klucze, które są pod kontrolą użytkownika (samoopieka), czy też klucze są przechowywane u zewnętrznego opiekuna?



Jeśli portfel Lightning korzysta z węzła zewnętrznej firmy, to ten węzeł decyduje o sposobie komunikowania się z siecią Bitcoin. W związku z tym korzystanie z węzła Lightning należącego do firmy zewnętrznej oznacza również korzystanie z jej węzła Bitcoin. Wybór między pełnoprawnym węzłem Bitcoin a węzłem Bitcoin firmy zewnętrznej istnieje tylko wtedy, gdy portfel Lightning używa własnego węzła Lightning.

Na najwyższym poziomie abstrakcji najbardziej podstawowe są pytania 1. i 3. Na podstawie odpowiedzi na te dwa pytania możemy wyodrębnić cztery możliwe kategorie. Te cztery kategorie można umieścić w kwadrancie, co pokazano w tabeli 2.1. Należy jednak pamiętać, że jest to tylko jeden ze sposobów kategoryzacji portfeli Lightning.

Tabela 2.1. Kwadrant portfeli Lightning

	Pełnoprawny węzeł Lightning	Zewnętrzny węzeł Lightning
Niepowierniczy	Q1: Wysokie umiejętności techniczne, najmniejsze zaufanie do podmiotów zewnętrznych, w większości nie wymaga żadnych uprawnień	Q2: Poziom techniczny poniżej średniego, zaufanie do podmiotów zewnętrznych, wymaga pewnych uprawnień
Powierniczy	Q3: Poziom techniczny powyżej średniego, zaufanie do podmiotów zewnętrznych, wymaga pewnych uprawnień	Q4: Niski poziom techniczny, wysokie zaufanie do podmiotów zewnętrznych, wymaga uprawnień

Kwadrant 3 (Q3), w którym korzysta się z pełnoprawnego węzła Lightning, ale klucze przechowywane są u opiekuna, nie jest powszechnie używany. Przyszłe portfele w tym kwadrancie mogą wymagać od użytkownika dbania o operacyjne aspekty swojego węzła, ale wymagać przekazywania dostępu do kluczy podmiotowi zewnętrznemu, który przede wszystkim używa magazynowania w trybie *cold storage*.

Portfele Lightning można zainstalować na wielu urządzeniach, w tym na laptopach, serwerach i urządzeniach mobilnych. Aby uruchomić pełnoprawny węzeł Lightning, należy użyć serwera lub komputera stacjonarnego. Urządzenia mobilne i laptopy zwykle nie są wystarczająco mocne pod względem pojemności, mocy obliczeniowej, żywotności akumulatora czy też możliwości komunikacyjnych. Kategorię zewnętrznych węzłów Lightning można podzielić dalej na następujące podkategorie:

#### *Lekkie (ang. lightweight)*

Oznacza, że portfel nie obsługuje węzła Lightning, a zatem informacje o Lightning Network musi pobrać przez internet z węzła należącego do podmiotu zewnętrznego.

#### *Brak*

Oznacza to, że nie tylko węzeł Lightning jest obsługiwany przez podmiot zewnętrzny, ale także większość portfela jest obsługiwana przez podmiot zewnętrzny w chmurze. Jest to portfel powierniczy, w którym ktoś inny ma kontrolę nad funduszami.

Wspomniane podkategorie zostały wymienione w tabeli 2.2.

Tabela 2.2. Przykłady popularnych portfeli Lightning

Aplikacja	Urządzenia	Węzeł Lightning	Węzeł Bitcoin	Magazyn kluczy
Blue Wallet	mobilne	Brak	Brak	powierniczy
Breez Wallet	mobilne	Pełnoprawny węzeł	Neutrino	niewierniczy
Eclair Mobile	mobilne	lekki	Electrum	niewierniczy
Intxbot	mobilne	Brak	Brak	powierniczy
Muun	mobilne	lekki	Neutrino	niewierniczy
Phoenix Wallet	mobilne	lekki	Electrum	niewierniczy
Zeus	mobilne	Pełnoprawny węzeł	Bitcoin Core (btcd)	niewierniczy
Electrum	desktop	Pełnoprawny węzeł	Bitcoin Core (Electrum)	niewierniczy
Zap Desktop	desktop	Pełnoprawny węzeł	Neutrino	niewierniczy
c-lightning	serwery	Pełnoprawny węzeł	Bitcoin Core	niewierniczy
Eclair Server	serwery	Pełnoprawny węzeł	Bitcoin Core/Electrum	niewierniczy
Ind	serwery	Pełnoprawny węzeł	Bitcoin Core (btcd)	niewierniczy

Oto inne terminy z kolumny *Węzeł Bitcoin* z tabeli 2.2 wymagające omówienia:

#### *Neutrino*

Ten portfel nie obsługuje węzła Bitcoin. Zamiast tego węzłem Bitcoin zarządza ktoś inny (podmiot zewnętrzny). Dostęp do niego uzyskuje się za pośrednictwem protokołu Neutrino.

## *Electrum*

Ten portfel nie obsługuje węzła Bitcoin. Zamiast niego dostęp do węzła Bitcoin obsługiwanego przez kogoś innego (podmiot zewnętrzny) uzyskuje się za pośrednictwem protokołu Electrum.

## *Bitcoin Core*

Implementacja węzła Bitcoin.

## *btcd*

Inna implementacja węzła Bitcoin.

Kilka przykładów popularnych aplikacji węzłów Lightning oraz portfeli dla różnych typów urządzeń zestawiono w tabeli 2.2. Lista została posortowana najpierw według typu urządzenia, a następnie alfabetycznie.

## **Bitcoin Testnet**

System Bitcoin oferuje alternatywny blockchain do celów testowych, o nazwie *testnet*, w odróżnieniu od „normalnego” łańcucha Bitcoin, określanego jako *mainnet*. Walutą w *testnet* jest *testnet bitcoin* (tBTC) — bezwartościowa imitacja bitcoina używana wyłącznie w celach testowych. Wszystkie funkcje Bitcoina w *testnet* zostały dokładnie odtworzone. Te pieniądze nie są jednak nic warte, zatem dosłownie nie masz nic do stracenia!

W *testnet* mogą również działać niektóre portfele Lightning, co umożliwia dokonywanie płatności Lightning za pomocą testowych bitcoinów, bez ryzyka utraty prawdziwych pieniędzy. Sieć *testnet* zapewnia świetny sposób na bezpieczne eksperymentowanie z Lightning Network. Jednym z przykładów portfela Lightning obsługującego sieć *testnet* jest Eclair Mobile, którego Alicja używa w tym rozdziale.

Pewną kwotę tBTC do testów możesz pobrać z portfeli *testnet bitcoin faucet*, które dostarczają na żądanie darmowej waluty tBTC. Oto kilka z nich:

- <https://coinafaucet.eu/en/btc-testnet>,
- <https://testnet-faucet.mempool.co>,
- <https://bitcoinafaucet.uo1.net>,
- <https://testnet.help/en/btcfaucet/testnet>.

Wszystkie przykłady w tej książce można dokładnie odtworzyć w sieci *testnet* z wykorzystaniem waluty tBTC. Dzięki temu możesz śledzić przykłady bez ryzyka utraty prawdziwych pieniędzy.

## **Równoważenie złożoności i kontroli**

W portfelach Lightning należy zachowywać właściwą równowagę pomiędzy złożonością a poziomem kontroli sprawowanej przez użytkownika. Portfele dające użytkownikowi największą kontrolę nad funduszami, zapewniające najwyższy stopień prywatności i największą niezależność od usług podmiotów zewnętrznych są z konieczności bardziej złożone i trudne w obsłudze. Wraz

z rozwojem technologii niektóre z tych kompromisów stają się mniej restrykcyjne, a użytkownicy mogą uzyskać większą kontrolę bez większej złożoności. Na razie jednak liczne firmy, w różnych projektach, badają rozmaite rozwiązania spektrum kontrola-złożoność, licząc na znalezienie „idealnego punktu” dla docelowych użytkowników.

Gdy wybierasz portfel, pamiętaj, że zastosowano w nim wspomniane kompromisy, nawet jeśli ich nie dostrzeżesz. Na przykład w wielu portfelach podjęto próbę zdjęcia z użytkowników obowiązków związanych z zarządzaniem kanałami. W tym celu w tych portfelach zastosowano węzły centralne, do których automatycznie łączą się wszystkie portfele. Zaletą tego kompromisu jest uproszczony interfejs i wrażenie łatwiejszej obsługi, wprowadza on jednak pojedynczy punkt awarii (ang. *single point of failure* — SPOF), ponieważ do działania portfela stają się niezbędne węzły centralne. Co więcej, poleganie na takim „centralnym węźle” może pogorszyć prywatność użytkownika, „centrala” zna bowiem nadawcę i potencjalnie (jeśli wykonuje płatności w imieniu użytkownika) również odbiorcę każdej płatności dokonanej z wykorzystaniem portfela użytkownika.

W następnym podrozdziale powrócimy do Alicji i spróbujemy wykonać wraz z nią pierwszą konfigurację portfela Lightning. Wybrała portfel, który jest bardziej zaawansowany od prostszych portfeli powierniczych. Pozwoliło to nam zaprezentować część związanej z tym złożoności i wprowadzić w tematykę niektórych z wewnętrznych działań zaawansowanego portfela. Może się okazać, że Twój pierwszy idealnie dopasowany portfel został zaprojektowany pod kątem łatwości obsługi i zostały w nim przyjęte pewne kompromisy związane z poziomem kontroli i prywatnością. A być może jesteś bardziej zaawansowanym użytkownikiem i chcesz uruchomić własne węzły Lightning i Bitcoin w ramach rozwiązania takiego portfela?

## Pobieranie i instalowanie portfela Lightning

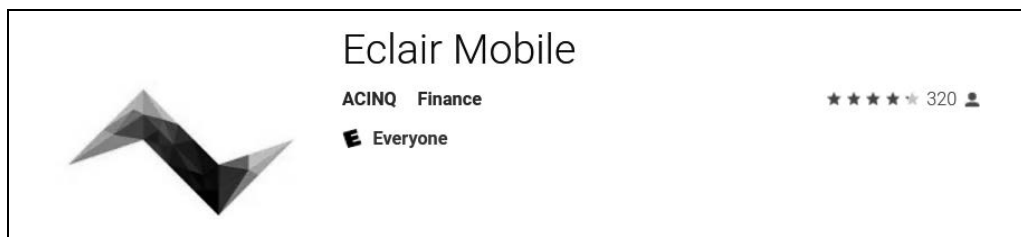
Gdy szukasz nowego portfela kryptowalut, musisz bardzo uważnie wybrać bezpieczne źródło oprogramowania.

Niestety, istnieje wiele fałszywych aplikacji portfelowych, które mogą skraść Twoje pieniądze. Niektóre z nich znajdują się nawet na renomowanych i rzekomo sprawdzonych stronach z oprogramowaniem, takich jak sklepy Apple i Google. Niezależnie od tego, czy instalujesz swój pierwszy czy dziesiąty portfel, zawsze zachowuj szczególną ostrożność. Fałszywa aplikacja może nie tylko ukraść wszystkie powierzone jej pieniądze, ale również, dzięki przejęciu kontroli nad systemem operacyjnym urządzenia mobilnego, ukraść klucze i hasła z innych aplikacji.

Alicja korzysta z urządzenia z systemem Android. Zamierza pobrać i zainstalować aplikację portfela Eclair ze sklepu Google Play. Szukając aplikacji w Google Play, znalazła pozycję „Eclair Mobile”, co pokazano na rysunku 2.1.



Dzięki wykorzystaniu testowych bitcoinów sieci testnet możliwe jest eksperymentowanie i testowanie oprogramowania Bitcoin bez żadnego ryzyka (z wyjątkiem utraty własnego czasu). Aby wypróbować Lightning Network (w sieci testnet), możesz także pobrać ze sklepu Google Play testowy portfel Eclair.



Rysunek 2.1. Eclair Mobile w sklepie Google Play

Alicja zauważyła na tej stronie kilka różnych elementów, które pomogły jej zyskać pewność, że ten portfel „Eclair Mobile” jest tym, którego szuka. Po pierwsze, jako twórcę tego mobilnego portfela wymieniono organizację ACINQ<sup>2</sup>. Alicja na podstawie przeprowadzonego własnego rozpoznania wie, że to właściwa firma. Po drugie, portfel został zainstalowany ponad 10 tysięcy razy i ma ponad 320 pozytywnych komentarzy. Jest mało prawdopodobne, aby była to nieuczciwa aplikacja, która przypadkiem wkradła się do Sklepu Google Play. W trzecim kroku Alicja przeszła na stronę internetową ACINQ (<https://acinq.co>). Sprawdziła, czy strona internetowa jest bezpieczna. W tym celu zweryfikowała, czy adres strony zaczyna się od *https* oraz czy jest poprzedzony w niektórych przeglądarkach kłódką. Na stronie internetowej przeszła do sekcji *Download*, a potem poszukiwała odnośnika do serwisu *Google App Store*. Znalazła odpowiedni odnośnik i go kliknęła. Porównała, że ten odnośnik przenosi ją do tej samej aplikacji w *Google App Store*. Zadowolona z tych odkryć, Alicja zainstalowała na swoim urządzeniu mobilnym aplikację Eclair.



Podczas instalowania oprogramowania na dowolnym urządzeniu zawsze zachowuj szczególną ostrożność. Istnieje wiele fałszywych portfeli kryptowalut, które mogą nie tylko przejąć Twoje pieniądze, ale również zagrozić wszystkim innym aplikacjom na Twoim urządzeniu.

## Tworzenie nowego portfela

Kiedy Alicja po raz pierwszy otworzyła aplikację Eclair Mobile, wyświetliła się w niej prośba o wybranie opcji „Utwórz nowy portfel” (*Create a New Wallet*) lub „Zaimportuj istniejący portfel” (*Import an Existing Wallet*). Alicja zdecydowała się utworzyć nowy portfel. Najpierw jednak opowiemy, dlaczego te opcje zostały jej przedstawione i co to znaczy „zaimportować istniejący portfel”.

## Odpowiedzialność związana z przechowywanymi kluczami

Jak wspomnieliśmy na początku tego rozdziału, Eclair jest portfelem niepowierniczym. Oznacza to, że Alicja ma wyłączną opiekę nad kluczami używanymi do kontrolowania jej bitcoinów. Oznacza to również, że Alicja jest odpowiedzialna za ochronę i tworzenie kopii zapasowych tych kluczy. Jeśli Alicja utraci klucze, nikt nie będzie mógł jej pomóc w odzyskaniu bitcoinów i zostaną one bezpowrotnie utracone.

<sup>2</sup> ACINQ: programiści portfela Eclair Mobile Lightning.



W przypadku portfela Eclair Mobile Alicja sprawuje opiekę i ma kontrolę nad kluczami, a zatem ponosi pełną odpowiedzialność za bezpieczeństwo kluczy i tworzenie k kopii zapasowych. Jeśli zgubi klucze, straci bitcoiny i nikt nie będzie w stanie pomóc jej w ich odzyskaniu!

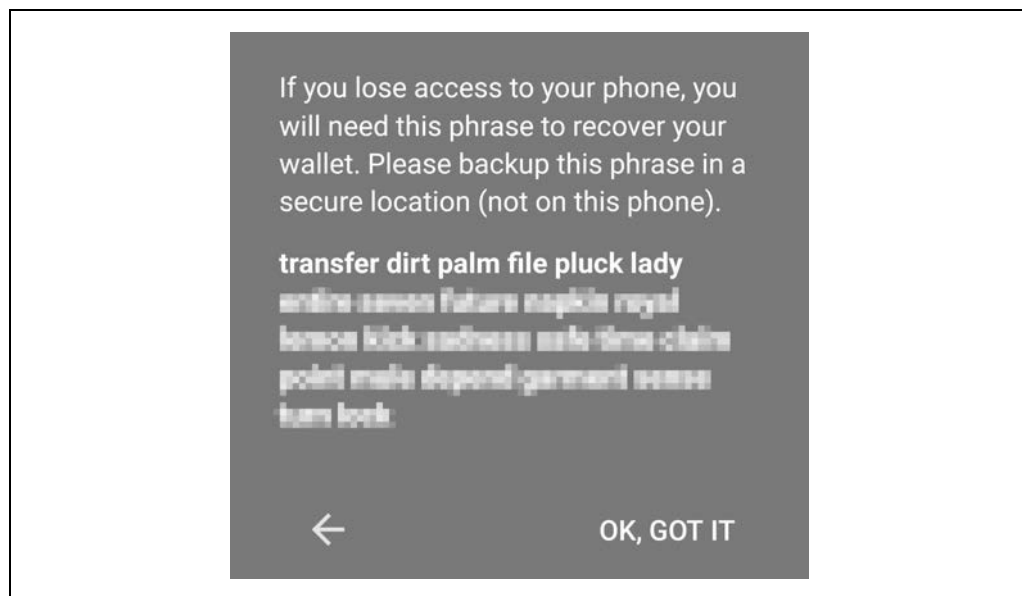
## Słowa mnemoniczne

Eclair Mobile, podobnie jak większość portfeli Bitcoina, zapewnia tzw. *frazę mnemoniczną* (czasami nazywaną również „ziarnem” — ang. *seed*, lub „frazą zalążkiem” — ang. *seed phrase*), której Alicja może użyć do odtworzenia klucza w warunkach awarii. Fraza mnemoniczna składa się z 24 angielskich słów, wybranych losowo przez oprogramowanie i stosowanych jako podstawa kluczy generowanych przez portfel. W razie zagubienia urządzenia mobilnego, wystąpienia błędu w oprogramowaniu lub uszkodzenia pamięci Alicja może skorzystać z frazy mnemonicznej, aby przywrócić wszystkie transakcje i środki w portfelu Eclair Mobile.



Poprawną nazwą słów zapewniających kopię zapasową jest „frazę mnemoniczną” (ang. *mnemonic phrase*). Unikamy używania terminu „ziarno” na określenie frazy mnemonicznej, ponieważ jego użycie, choć powszechne, jest błędne.

Kiedy Alicja podjęła decyzję o utworzeniu nowego portfela, zobaczyła ekran z frazą mnemoniczną, która ma postać pokazaną na rysunku 2.2.



Rysunek 2.2. Nowa fraza mnemoniczna portfela

Na rysunku 2.2 celowo zamazaliśmy część frazy mnemonicznej, aby uniemożliwić czytelnikom tej książki jej wykorzystanie.

## Bezpieczne przechowywanie frazy mnemonicznej

Alicja musi być ostrożna przy przechowywaniu frazy mnemonicznej. Powinna przechowywać ją tak, aby zapobiec jej kradzieży, a jednocześnie uniknąć przypadkowej utraty. Zalecanym sposobem zrównoważenia wymienionych zagrożeń jest zapisanie dwóch kopii frazy mnemonicznej na papierze i ponumerowanie poszczególnych słów — kolejność ma znaczenie.

Gdy Alicja zapisała frazę mnemoniczną, dotknęła na ekranie linku *OK GOT IT*. Po chwili wyświetlił się quiz, którego zadaniem jest sprawdzenie, czy fraza została poprawnie zapisana. W quizie wyświetliły się prośby o podanie trzech lub czterech losowych słów. Alicja nie spodziewała się quizu, ale ponieważ poprawnie zapisała frazę mnemoniczną, bez żadnych trudności przeszła test.

Po zapisaniu frazy mnemonicznej i zdaniu testu zaprezentowanego za pomocą quizu Alicja powinna schować każdą z kopii w osobnym, bezpiecznym miejscu, takim jak zamknięta szuflada biurka lub ognioodporny sejf.



Nigdy nie próbuj stosować schematu bezpieczeństwa typu „zrób to sam”, który w jakikolwiek sposób odbiega od metody zalecanej w punkcie „Bezpieczne przechowywanie frazy mnemonicznej”. Nie przecinaj frazy na pół, nie twórz zrzutów ekranu, nie przechowuj jej na dyskach USB lub dyskach w chmurze, nie szyfruj jej ani nie stosuj żadnej innej niestandardowej metody. W ten sposób przechyliłsz szalę i podejmiesz niepotrzebne ryzyko trwałej utraty środków. Wiele osób straciło swoje fundusze nie z powodu kradzieży, ale dlatego, że próbowały skorzystać z niestandardowych rozwiązań bez specjalistycznej wiedzy potrzebnej do zredukowania związanych z tym zagrożeń. Zalecona najlepsza metoda została dobrze przemyślana przez ekspertów i jest odpowiednia dla zdecydowanej większości użytkowników.

Po zainicjowaniu portfela Eclair Mobile Alicja wyświetliła krótki samouczek, w którym wyjaśniono różne elementy interfejsu użytkownika. Nie będziemy powtarzać tu opisu samouczka, ale przeanalizujemy wszystkie te elementy, gdy będziemy śledzić, jak Alicja próbuje kupić filiżankę kawy!

## Ładowanie bitcoinów do portfela

Alicja ma teraz portfel Lightning. Ale jest on pusty! Teraz stoi przed jednym z trudniejszych aspektów tego eksperymentu: musi znaleźć sposób na zdobycie bitcoinów i załadowanie ich do portfela Eclair.



Jeżeli Alicja ma już bitcoiny w innym portfelu, to zamiast kupować nowe, może je wysłać do swojego portfela Eclair i w ten sposób załadować portfel.



## Zdobywanie bitcoinów

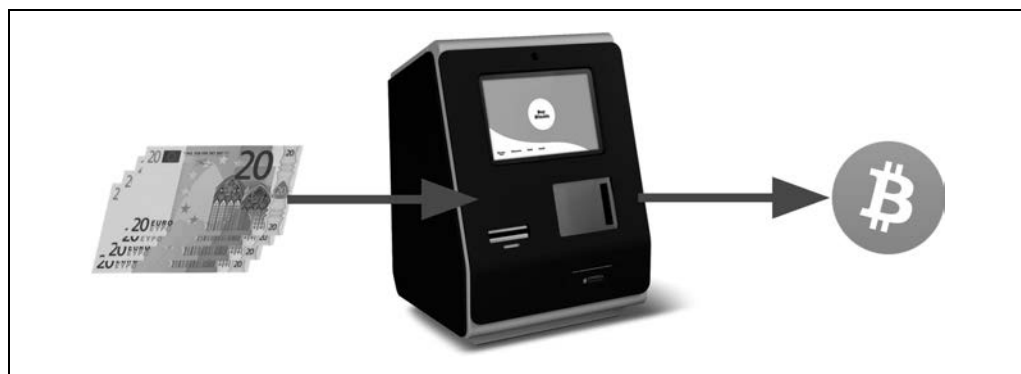
Istnieje kilka sposobów, w jakie Alicja może nabyć bitcoiny:

- Może wymienić na giełdzie kryptowalut na bitcoiny część swojej waluty krajowej (na przykład PLN).
- Może w zamian za gotówkę kupić trochę bitcoinów od przyjaciela lub znajomego.
- Może znaleźć w swojej okolicy *bankomat Bitcoin*, który działa jak automat sprzedający bitcoiny za gotówkę.
- Może akceptować płatności w bitcoinach, oferując swoje usługi lub sprzedaż produktów.
- Może poprosić swojego pracodawcę lub klientów, aby dokonywali płatności w bitcoinach.

Wszystkie te metody charakteryzują się różnym stopniem trudności, a wiele z nich wymaga uiszczenia opłaty. Niektóre także, aby zachowana była zgodność z lokalnymi przepisami bankowymi, będą wymagały od Alicji dostarczenia dokumentów tożsamości. Jednak wszystkie te metody pozwolą Alicji odbierać bitcoiny.

## Odbieranie bitcoinów

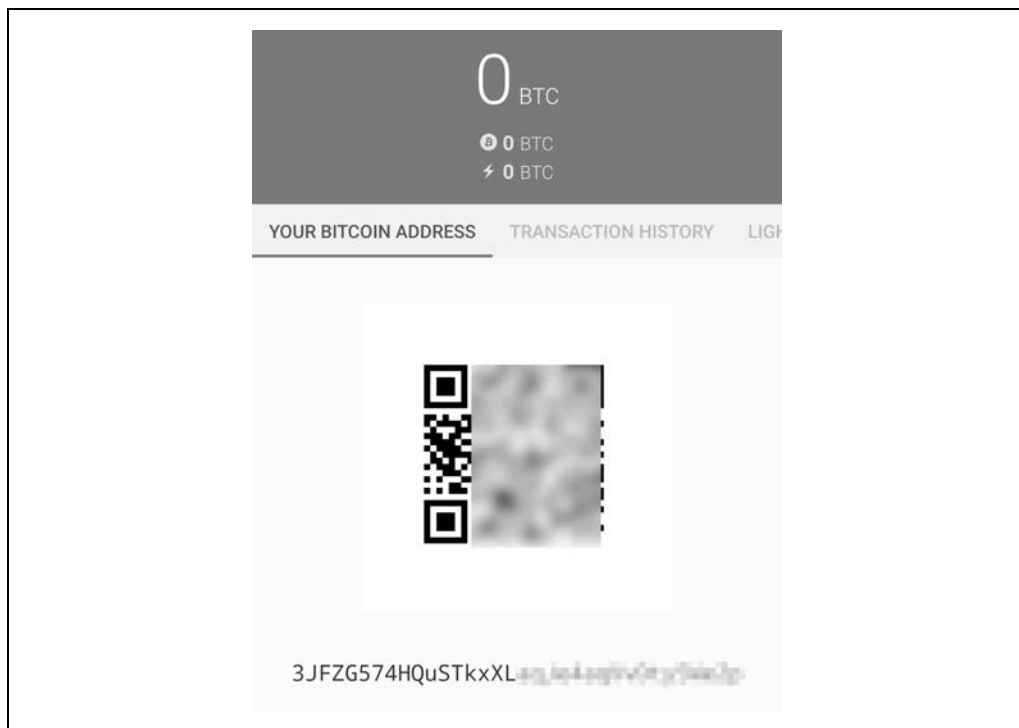
Żałóżmy, że Alicja znalazła lokalny bankomat Bitcoin i postanowiła kupić trochę bitcoinów za gotówkę. Przykład bankomatu Bitcoin, zbudowanego przez firmę Lamassu, jest pokazany na rysunku 2.3. Takie bankomaty Bitcoin przyjmują walutę krajową (gotówkę) za pośrednictwem automatu gotówkowego i wysyłają bitcoiny na adres zeskanowany z portfela użytkownika za pomocą wbudowanej kamery.



Rysunek 2.3. Bankomat Bitcoin firmy Lamassu

Aby otrzymać bitcoiny do swojego portfela Eclair Lightning, Alicja musi okazać w bankomacie adres Bitcoin z portfela Eclair Lightning. Następnie bankomat wyśle Alicji nabyte bitcoiny na podany adres.

Aby zobaczyć adres Bitcoin portfela Eclair, Alicja musi przesunąć palcem do lewej kolumny o nazwie *YOUR BITCOIN ADDRESS* (rysunek 2.4). Zobaczysz kwadrat z kodem QR, a poniżej niego ciąg liter i cyfr.



Rysunek 2.4. Adres bitcoin Alicji w aplikacji Eclair

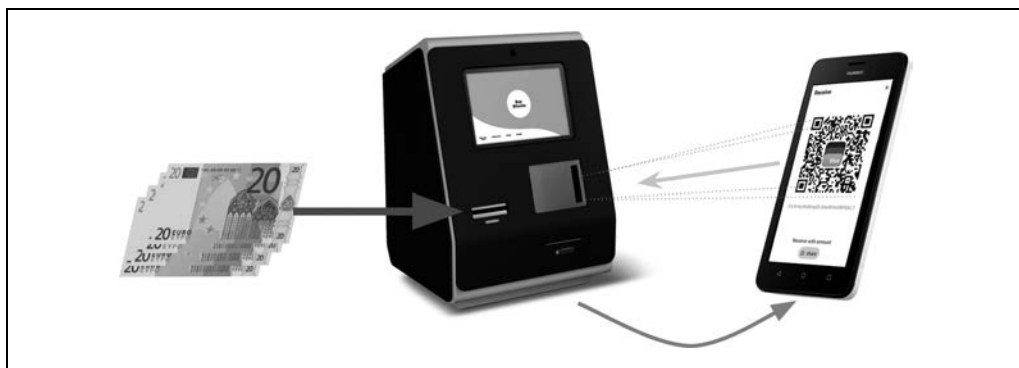
Kod QR zawiera ten sam ciąg liter i cyfr, który znajduje się pod kodem, ale w formacie łatwym do zeskanowania. Dzięki temu Alicja nie musi wpisywać adresu Bitcoin. Na powyższym rzucie ekranu (rysunek 2.4) celowo zamazaliśmy oba elementy, aby zapobiec przypadkowemu przesyłaniu przez czytelników bitcoinów na podany adres.



Zarówno adresy Bitcoin, jak i kody QR zawierają informacje o wykrywaniu błędów. Zapobiega to błędom podczas wpisywania lub skanowania, w których wyniku mogłyby być wygenerowane „niepoprawne” adresy Bitcoin. Jeśli w adresie wystąpi błąd, każdy portfel Bitcoin go zauważy i odmówi zaakceptowania ważności adresu.

Alicja może wziąć swoje urządzenie mobilne do bankomatu i pokazać je do wbudowanej kamery, tak jak pokazano na rysunku 2.5. Po umieszczeniu gotówki w bankomacie otrzyma bitcoiny do portfela Eclair!

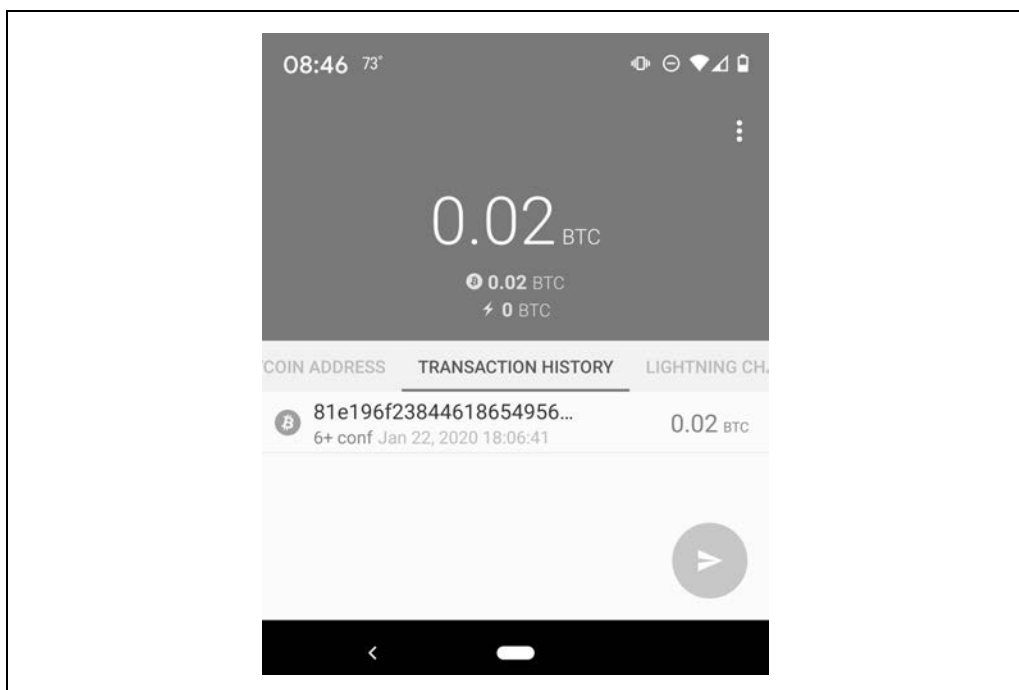
Alicja zobaczy transakcję z bankomatu w zakładce *TRANSACTION HISTORY* portfela Eclair. Chociaż Eclair wykryje transakcję Bitcoin w ciągu zaledwie kilku sekund, to „potwierdzenie” transakcji Bitcoin w blockchainie może zająć około godziny. Jak widać na rysunku 2.6, portfel Eclair Alicji wyświetla pod transakcją „6+ conf”. Oznacza to, że transakcja otrzymała wymagane minimum sześć potwierdzeń, a środki są teraz gotowe do użycia.



Rysunek 2.5. Bankomat Bitcoin skanujący kod QR



Liczba potwierżeń transakcji to liczba bloków wydobytych od czasu wydobycia bloku, który zawierał tę transakcję (włącznie z nim). Uzyskanie sześciu potwierżeń jest najlepszą praktyką, ale inne portfele Lightning mogą uznawać kanał za otwarty po dowolnej liczbie potwierżeń. Niektóre portfele nawet skalują w górę liczbę wymaganych potwierżeń o wartość pieniężną kanału.



Rysunek 2.6. Alicja otrzymuje bitcoiny

Chociaż w tym przykładzie Alicja skorzystała z bankomatu do nabycia swojego pierwszego bitcoina, te same pojęcia miałyby zastosowanie także w przypadkach, gdyby użyła jednej z innych metod wymienionych w punkcie „Zdobywanie bitcoinów”. Na przykład, gdyby Alicja chciała

sprzedać produkt lub zapewnić profesjonalną usługę w zamian za bitcoiny, jej klienci mogliby skanować adres Bitcoin za pomocą swoich portfeli i płacić jej w bitcoinach.

Na podobnej zasadzie, gdyby Alicja wystawiła klientowi rachunek za usługę oferowaną przez internet, mogłaby wysłać do niego wiadomość e-mail lub SMS-a z adresem Bitcoin lub kodem QR. Klient mógłby wówczas zapłacić jej przez wklejenie informacji do portfela Bitcoin lub ich zeskanowanie.

Alicja mogłaby nawet wydrukować kod QR, umieścić go w widocznym miejscu i prezentować publicznie, aby otrzymywać napiwki. Na przykład może przymocować kod QR do gitary i otrzymywać napiwki podczas ulicznych występów!<sup>3</sup>

Na koniec, jeśli Alicja kupiłaby bitcoiny na giełdzie kryptowalut, mogłaby (i powinna) „wypłacić” je przez wklejenie na stronie internetowej giełdy swojego adresu Bitcoin. W odpowiedzi giełda wysłałaby bitcoiny bezpośrednio na jej adres.

## Od sieci Bitcoin do Lightning Network

Bitcoin Alicji jest teraz obsługiwany przez portfel Eclair i został zarejestrowany w blockchainie Bitcoin. W tym momencie bitcoiny Alicji są w trybie *on-chain*, co oznacza, że transakcja została rozesłana do całej sieci Bitcoin. Została zweryfikowana przez wszystkie węzły i wydobyta (zarejestrowana) w blockchainie Bitcoin.

Do tej pory portfel Eclair Mobile działał tylko jako portfel Bitcoin, a Alicja jeszcze nie korzystała z funkcji Lightning Network Eclair. Podobnie jak w przypadku wielu innych portfeli Lightning Eclair łączy w sobie węzeł Bitcoin i Lightning Network. Działa zarówno jako portfel Bitcoin, jak i portfel Lightning.

Alicja jest teraz gotowa, by używać Lightning Network. W tym celu powinna przenieść swoje bitcoiny do trybu *off-chain*. Dzięki temu będzie mogła skorzystać z szybkich, tanich i prywatnych płatności oferowanych przez Lightning Network.

### Kanały sieci Lightning Network

Alicja przesunęła palcem w prawo i uzyskała dostęp do sekcji *LIGHTNING CHANNELS* aplikacji Eclair. Może tutaj zarządzać kanałami, które połączą jej portfel z siecią Lightning Network.

Aby nieco rozjaśnić pojęcie kanału LN, spróbujmy przyjrzeć się jego definicji. Po pierwsze, słowo „kanał” jest metaforą *relacji finansowej* pomiędzy portfelem Lightning Alicji a innym portfelem Lightning. Nazywamy tę relację kanałem, ponieważ jest to mechanizm umożliwiający portfelowi Alicji i temu drugiemu portfelowi wymianę wielu płatności w sieci Lightning Network (*off-chain*) bez dokonywania transakcji w blockchainie Bitcoin (*on-chain*).

---

<sup>3</sup> Ogólnie rzecz biorąc, nie zaleca się używania tego samego adresu Bitcoin do wielu płatności, ponieważ wszystkie transakcje Bitcoin są publiczne. Wścibska osoba przechodząca obok może zeskanować kod QR Alicji i zobaczyć, ile napiwków Alicja już otrzymała na ten adres na blockchainie Bitcoin. Na szczęście Lightning Network oferuje bardziej prywatne rozwiązania tego problemu, omówione w dalszej części książki!

Portfel lub węzeł, dla którego Alicja otwiera kanał, nazywa się *kanalem partnerskim* (ang. *channel peer*). Po „otwarciu” kanał może być używany do wysyłania wielu płatności w obu kierunkach między portfelem Alicji a partnerem kanału.

Co więcej, kanał partnerski może *przekazywać* płatności dalej za pośrednictwem innych kanałów do sieci Lightning Network. Dzięki temu Alicja może *przekazać* płatność do dowolnego portfela (na przykład portfela Lightning Bogdana), pod warunkiem że portfel Alicji znajdzie realną *ścieżkę* utworzoną za pomocą przeskoków pomiędzy kanałami, aż do portfela Bogdana.



Nie wszystkie kanały partnerskie są odpowiednie do przekazywania płatności. Tzw. kanały *well-connected* (dosłownie: dobrze połączone) mogą kierować płatności krótszymi ścieżkami do miejsca docelowego, co zwiększa szansę powodzenia operacji. Kanały partnerskie ze sporymi funduszami mogą obsługiwać większe płatności.

Innymi słowy, Alicja potrzebuje jednego lub większej liczby kanałów, które połączą ją z jednym lub kilkoma innymi węzłami w Lightning Network. Nie potrzebuje kanału, aby połączyć swój portfel bezpośrednio z „Kawiarnią u Bogdana” i wysłać Bogdanowi płatność, choć może również otworzyć kanał bezpośredni. Rolę pierwszego kanału Alicji może odgrywać dowolny węzeł w sieci Lightning Network. Im węzeł ma lepsze połączenia, tym większa jest liczba osób, do których Alicja może dotrzeć. W tym przykładzie chcemy również zademonstrować przekierowywanie płatności, dlatego Alicja nie otworzy kanału bezpośrednio do portfela Bogdana. Zamiast tego otworzy kanał do węzła *well-connected*. Użyje go potem do przekazania swojej płatności, a jeśli to będzie konieczne, przekieruje ją przez inne węzły, tak by płatność dotarła do Bogdana.

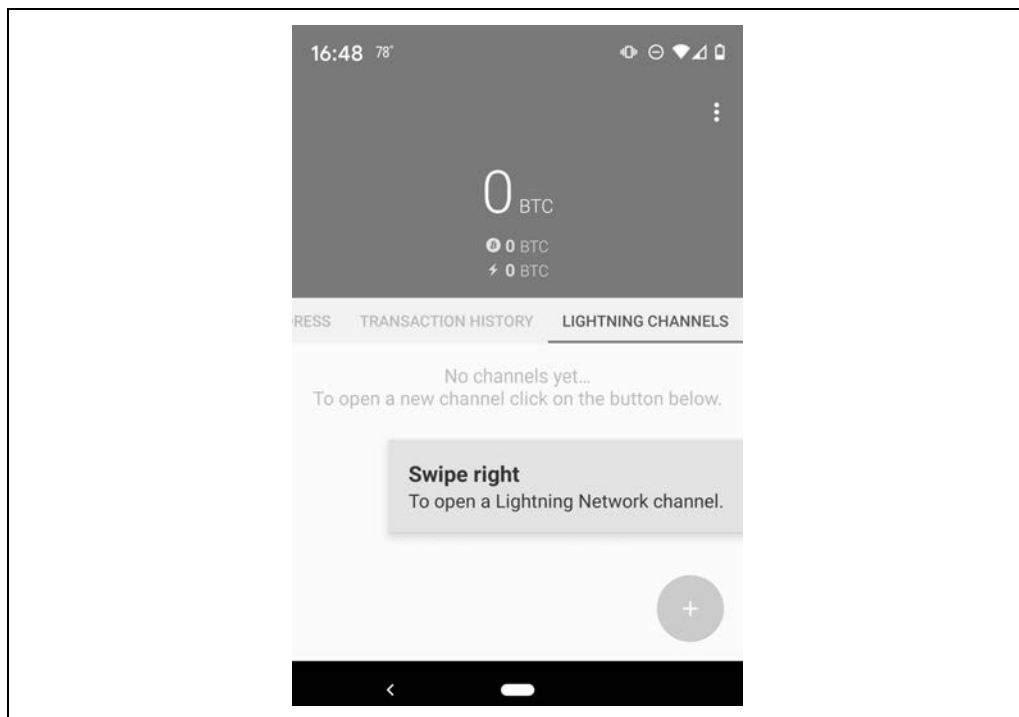
Na początku Alicja nie ma otwartych kanałów, więc jak widać na rysunku 2.7, zakładka *LIGHTNING CHANNELS* wyświetla pustą listę. W prawym dolnym rogu można zauważyć symbol plus (+). Jest to przycisk pozwalający otworzyć nowy kanał.

Alicja nacisnęła symbol plusa. W odpowiedzi wyświetliły się cztery możliwe sposoby otwarcia kanału:

- wklejenie identyfikatora URI węzła (*Paste a node URI*),
- skanowanie identyfikatora URI węzła (*Scan a node URI*),
- wybranie losowego węzła (*Random node*),
- wybranie węzła ACINQ (*ACINQ node*).

„Identyfikator URI węzła” to uniwersalny identyfikator zasobu (ang. *universal resource identifier*), który identyfikuje określony węzeł Lightning. Alicja może wkleić taki identyfikator URI ze schowka lub zeskanować kod QR zawierający te same informacje. Przykład identyfikatora URI węzła w postaci kodu QR jest pokazany na rysunku 2.8. Poniżej rysunku widnieje ten sam identyfikator w formacie tekstowym.

Alicja mogłaby wybrać określony węzeł Lightning lub użyć opcji *Random node*, aby portfel Eclair wybrał go losowo. Alicja wybrała jednak opcję *ACINQ Node*, co pozwoli jej połączyć się z jednym z węzłów *well-connected* ACINQ w sieci Lightning.



Rysunek 2.7. Zakładka LIGHTNING CHANNELS



Rysunek 2.8. Kod QR identyfikatora URI węzła

```
0237fefbe8626bf888de0cad8c73630e32746a22a2c4faa91c1d9877a3826e1174@1.ln.aantonop.com:9735
```

Wybór węzła ACINQ nieznacznie ograniczy prywatność Alicji, ponieważ sieć ACINQ daje możliwość podglądu wszystkich transakcji użytkownika. Taki wybór spowoduje również powstanie pojedynczego punktu awarii. Alicja będzie dysponować tylko jednym kanałem, a jeśli węzeł ACINQ nie będzie dostępny, nie będzie mogła dokonywać płatności. Aby wszystko było na początku proste, zaakceptujemy te kompromisy. W dalszych rozdziałach będziesz się stopniowo uczyć, w jaki sposób uzyskać większą niezależność oraz jak przyjmować mniej kompromisów!

Alicja wybrała pozycję *ACINQ Node* i jest gotowa do otwarcia swojego pierwszego kanału w sieci Lightning Network.

## Otwieranie kanału Lightning

Gdy Alicja wybrała węzeł, aby otworzyć nowy kanał, wyświetliło się pytanie o kwotę w bitcoinach, jaką chce zaalokować w tym kanale. W kolejnych rozdziałach omówimy implikacje tych wyborów. Na razie jednak Alicja zaalokowała w tym kanale prawie wszystkie swoje fundusze. Ponieważ Alicja jest zobowiązana do uiszczenia opłat transakcyjnych za otwarcie kanału, wybrała kwotę nieco mniejszą, niż wynosiło jej całkowite saldo<sup>4</sup>.

Alicja zaalokowała 0,018 BTC ze swoich 0,020 BTC do swojego kanału i zaakceptowała domyślną stawkę opłaty, co widać na rysunku 2.9.

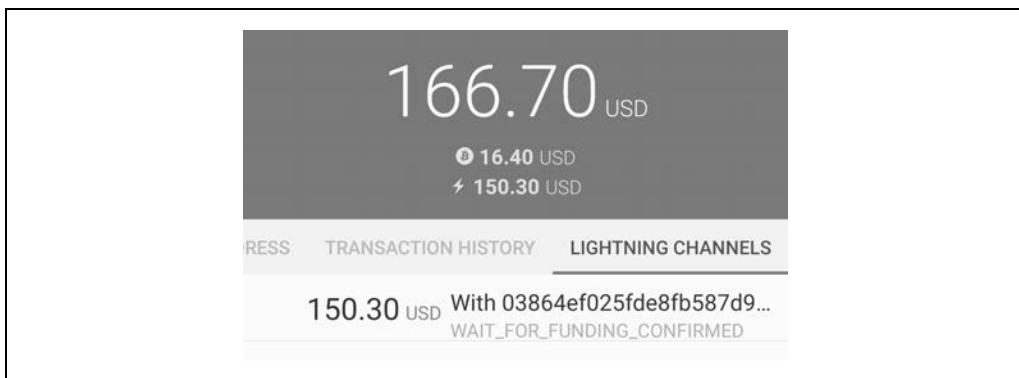


Rysunek 2.9. Otwieranie kanału Lightning

Po kliknięciu *OPEN* jej portfel utworzył specjalną transakcję Bitcoin, zwaną *transakcją finansowania* (ang. *funding transaction*), która otwiera kanał Lightning. Transakcja finansowania jest typu *on-chain* i jest wysyłana do sieci Bitcoin w celu jej potwierdzenia.

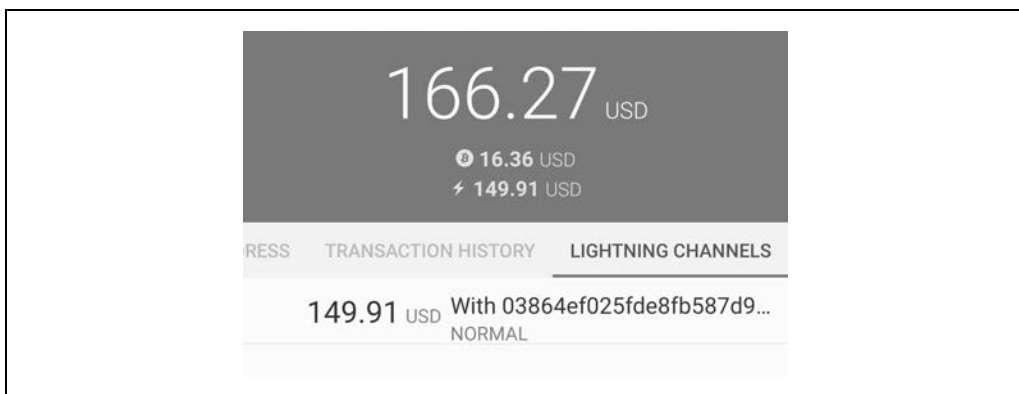
Alicja musi teraz ponownie poczekać (patrz rysunek 2.10), aż transakcja zostanie zarejestrowana w blockchainie Bitcoin. Podobnie jak w przypadku początkowej transakcji Bitcoin, użytej do nabycia bitcoina, Alicja musi poczekać na sześć lub więcej potwierdzeń (około godziny).

<sup>4</sup> Portfel Eclair nie oferuje opcji automatycznego obliczania niezbędnych opłat i przydzielania maksymalnej kwoty środków do kanału, więc Alicja musi sama je obliczyć.



Rysunek 2.10. Oczekiwanie na otwarcie kanału przez transakcję finansowania

Po potwierdzeniu transakcji finansowania kanał Alicji do węzła ACINQ zostaje otwarty, jest sfinansowany i gotowy, co widać na rysunku 2.11.



Rysunek 2.11. Kanał jest otwarty



Być może Ci się wydaje, że kwota kanału się zmieniła. Otóż nie. Kanał zawiera 0,018 BTC, ale w czasie między zrzutami ekranu zmienił się kurs BTC, zatem wartość w USD jest inna. W portfelu możesz wybrać pomiędzy wyświetlaniem salda w BTC lub w USD. Należy jednak pamiętać, że wartości w USD są obliczane w czasie rzeczywistym i się zmieniają!.

## Kupowanie filiżanki kawy za pomocą Lightning Network

Alicja ma teraz przygotowane to, co jest jej potrzebne do rozpoczęcia korzystania z Lightning Network. Jak widać, wymagało to trochę pracy i czasu oczekiwania na potwierdzenia. Teraz jednak wszystko może przebiegać szybko i łatwo. Sieć Lightning Network umożliwia dokonywanie płatności bez konieczności czekania na potwierdzenia, ponieważ środki są rozliczane w ciągu kilku sekund. Alicja wzięła swój telefon i poszła do „Kawiarni u Bogdana” w swojej okolicy. Jest podskrytowana perspektywą wypróbowania nowego portfela Lightning. Chce coś kupić!



## Kawiarnia u Bogdana

Bogdan posiada prostą aplikację punktu sprzedaży (ang. *point-of-sale* — PoS) dostępną dla każdego klienta, który chciałby płacić bitcoinami w sieci Lightning Network. Jak się dowiesz w następnym rozdziale, Bogdan korzysta z popularnej platformy open source *BTCPay Server*. Zawiera ona niezbędne komponenty rozwiązania e-commerce lub punktu sprzedaży detalicznej, takie jak:

- węzeł Bitcoin korzystający z oprogramowania Bitcoin Core,
- węzeł Lightning korzystający z oprogramowania c-lightning,
- prosta aplikacja PoS na tablet.

BTCPay Server upraszcza instalację potrzebnego oprogramowania, przesyłanie zdjęć i cen produktów oraz pozwala szybko uruchomić sklep.

Na ladzie w „Kawiarni u Bogdana” znajduje się tablet. Jego ekran jest pokazany na rysunku 2.12.



Rysunek 2.12. Aplikacja punktu sprzedaży w Kawiarni u Bogdana

## Rachunek Lightning

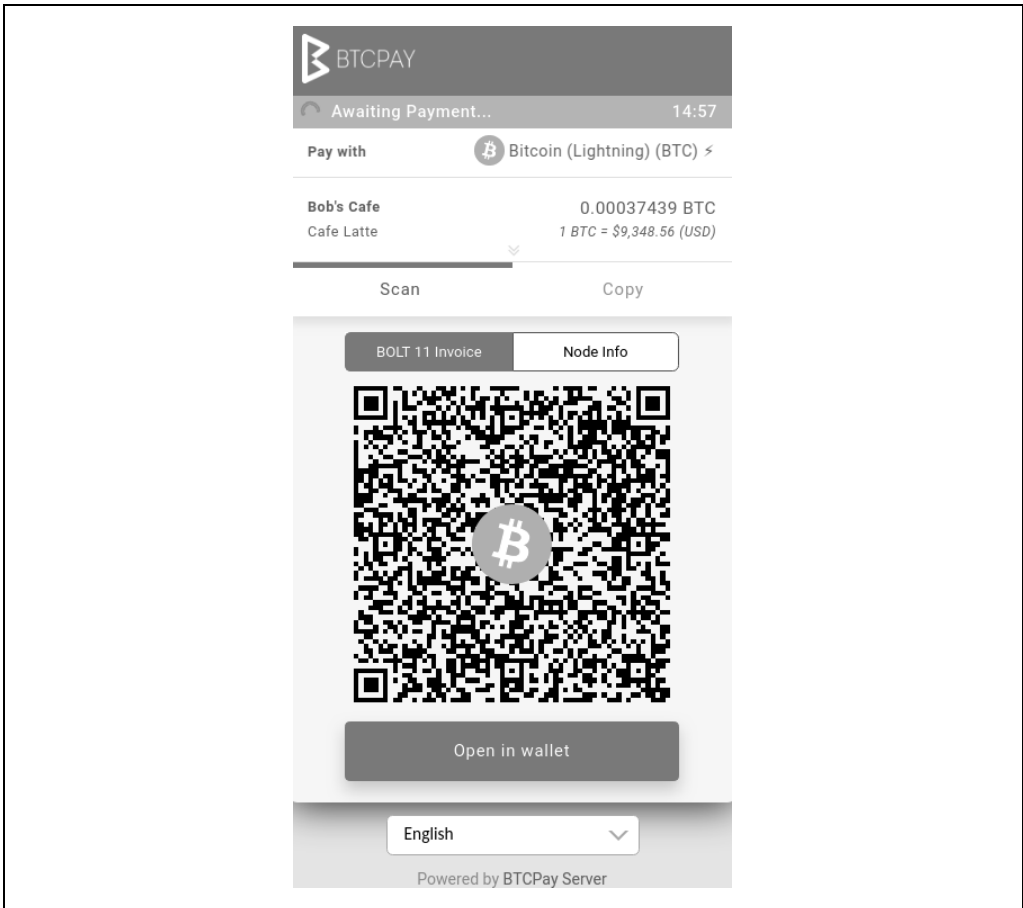
Alicja wybrała na ekranie opcję *Cafe Latte* i otrzymała rachunek Lightning (nazywany również „żądaniem płatności”), co pokazano na rysunku 2.13.

Aby zapłacić rachunek, Alicja otworzyła portfel Eclair i wybrała na karcie *TRANSACTION HISTORY* przycisk przesyłania płatności (oznaczony strzałką w górę), co pokazano na rysunku 2.14.

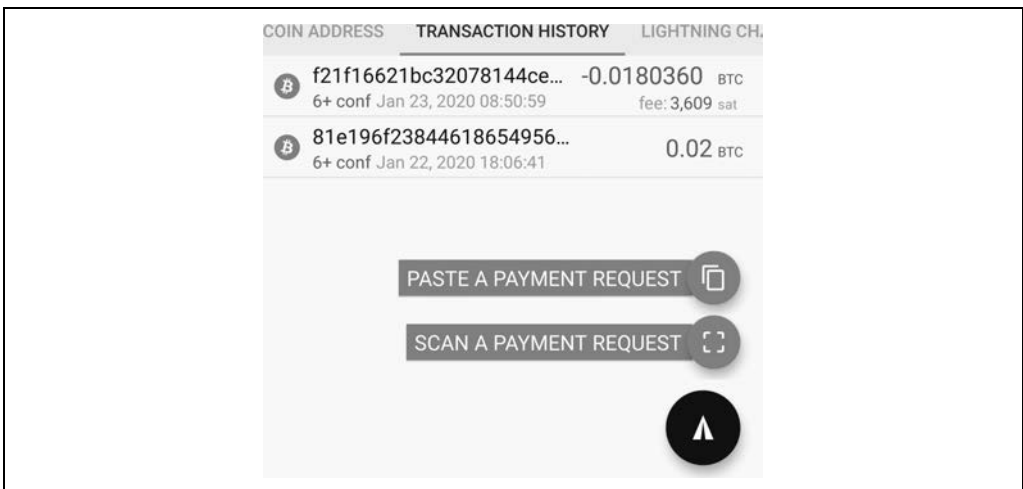


Termin „żądanie płatności” może się odnosić do żądania płatności Bitcoin lub rachunku Lightning. Terminy „faktura” i „żądanie płatności” są często używane zamiennie. Poprawny termin techniczny to „rachunek Lightning”, niezależnie od tego, jak nazywa się w portfelu.

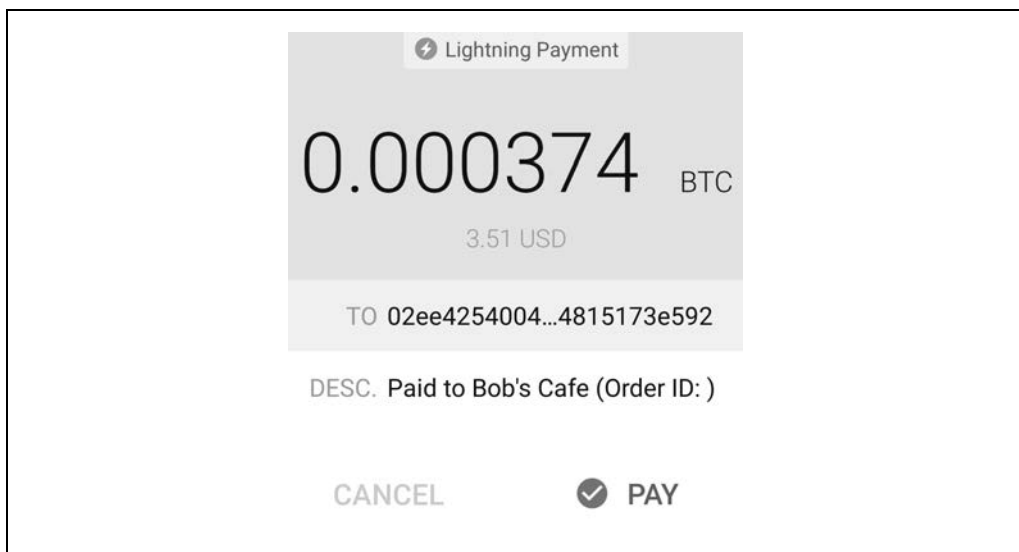
Alicja wybrała opcję *SCAN A PAYMENT REQUEST* (dosłownie: zeskanuj żądanie płatności) i zeskanowała kod QR wyświetlany na ekranie tabletu (patrz rysunek 2.13). Wyświetliła się prośba o potwierdzenie płatności, co pokazano na rysunku 2.15.



Rysunek 2.13. Rachunek Lightning za latte Alicji



Rysunek 2.14. Alicja powinna kliknąć przycisk Send



Rysunek 2.15. Potwierdzenie transakcji Alicji

Alicja nacisnęła *PAY*, a sekundę później na tablecie Bogdana wyświetliła się potwierdzona płatność. Alicja wykonała swoją pierwszą płatność LN! Płatność była szybka, niedroga i łatwa. Teraz może cieszyć się kawą latte kupioną za pomocą bitcoinów za pośrednictwem systemu płatności, który jest szybki, tani i zdecentralizowany. Od teraz Alicja może wybrać towar na ekranie tabletu Bogdana, zeskanować kod QR za pomocą telefonu komórkowego, kliknąć *PAY* i otrzymać kawę. Wszystko w ciągu kilku sekund i bez transakcji w trybie *on-chain*.

Płatności Lightning dla Bogdana również są lepsze. Jest przekonany, że otrzyma zapłatę za latte Alicji bez oczekiwania na potwierdzenie transakcji *on-chain*. Następnym razem, gdy Alicja będzie miała ochotę wypić kawę w „Kawiarni u Bogdana”, będzie mogła zapłacić bitcoinami w sieci Bitcoin lub Lightning Network. Którą metodę wybierze Twoim zdaniem?

## Podsumowanie

W tym rozdziale obserwowaliśmy Alicję podczas pobierania i instalowania pierwszego portfela Lightning. Alicja nabyła trochę bitcoinów, otworzyła swój pierwszy kanał Lightning i kupiła filiżankę kawy, za którą zapłaciła z wykorzystaniem Lightning Network. W dalszych rozdziałach zajrzemy „pod maskę” wszystkich komponentów w Lightning Network i się dowiesz, w jaki sposób płatność Alicji dotarła do „Kawiarni u Bogdana”.

---

# Skorowidz

## A

adres, 81  
    Bitcoin, 328  
    węzła, 164  
    wielopodpisowy, 62, 169  
akceptowanie kanałów, 351  
aktualizacje, 309  
    „od końca do końca”, 311  
    na poziomie budowy kanału, 312  
    sieci wewnętrznej, 311  
    zobowiązań, 216  
algorytm przekazywania płatności, 79  
anonimowość, 335  
aplikacje Lightning, 41, 357  
architektura Lightning Network, 157  
asymetryczne transakcje zobowiązania, 178  
ataki na sieć Lightning, 338  
    blokada płynności kanału, 343  
    DoS, 341, 342  
    łączenie nadawców z odbiorcami, 338  
    obserwowanie kwot płatności, 338  
    sondujące, 339  
    typu man-in-the-middle, 314  
    ujawnianie sald kanałów, 339  
    zagłuszanie zobowiązań, 343  
atomowy routing płatności, 192  
automatyczne  
    ponowne połączenie z siecią, 143  
    przekierowywanie portów, 133  
automatyczny restart  
    serwera, 143  
    węzła, 143  
autopilot, 147  
AWS, Amazon Web Services, 118

## B

bech32, 330  
bezpieczeństwo, 333  
    systemu operacyjnego, 136  
    węzła Lightning, 136  
BigSize, 307  
Bitcoin Core, 42, 93, 95  
Bitcoin Script, 204, 355, 369  
    skrypty  
        blokady czasowej, 374  
        blokujące, 371  
        odblokowujące, 371  
        wielopodpisowe, 373  
        z wieloma warunkami, 374  
    sterowanie przepływem w skryptach, 375  
    uruchamianie programu, 369  
Bitcoin Testnet, 42  
bitcoind  
    budowanie kontenera, 93  
    interakcje, 95  
    uruchamianie kontenera, 97, 103, 107  
bitcoiny  
    ładowanie do portfela, 46  
    odbieranie, 47  
    zdobywanie, 47  
bity funkcji, 309  
Blockchain, 26  
blokada czasowa, 213, 374  
bloki konstrukcyjne, 30  
blokowanie  
    użycie klucza publicznego, 371  
    użycie skrótu, 372  
błędy, 143, 230, 296  
    cebuli, 263, 264  
    limitu czasu, 212  
    zwracanie, 262

BOLT, 285  
bootstrapping, 270  
btcd, 42  
BTCPay Server, 124  
budowanie  
  kanału płatności, 163, 165, 356  
  kontenera, 378  
    bitcoind, 93  
    c-lightning, 96  
  nowego portfela, 44  
  ramki, framing, 304  
  sieci z węzłów Lightning, 110  
  sieci kanałów płatności, 194  
  transakcji finansowania, 169  
  transakcji wstępnie podpisanej, 171

## C

cebule  
  niestandardowe rekordy TLV, 265  
  o stałej długości, 247  
  opakowanie, 253  
  owijanie, 248  
  sprawdzanie, 257  
  typy błędów, 263, 264  
  wysyłanie, 256  
centralizacja, 349  
chmura, 118  
c-lightning, 96  
  budowanie kontenera, 96  
  instalowanie, 99  
  kod źródłowy, 99, 100  
  kompilowanie kodu, 100  
  uruchamianie kontenera, 97  
CPU, 119  
czas pracy węzła Lightning, 142

## D

dezanonimizacja, 335, 343  
DNS, 270  
Docker  
  instalowanie platformy, 377  
  konfiguracja sieci, 97  
  kontenery, 91  
    orkiestracja, 111  
    serwera Eclair, 106  
    węzła LND, 102  
  podstawowe polecenia, 378  
docker-compose, 111

dostarczanie płatności, 74, 289, 296  
  niepewność płynności, 293  
  prawdopodobieństwo, 293  
  próby i błędy, 296  
dostęp do węzłów, 137, 144  
dostępność węzła Lightning, 142  
dysk  
  HDD, 120  
  SSD, 120  
działanie sieci, 58

## E

Eclair, 106  
  instalowanie serwera, 109  
  kod źródłowy, 109  
  kompilowanie kodu, 109  
  uruchamianie kontenera, 107  
Eclair Lightning  
  projekt węzła, 106  
eksploratory Lightning, 38  
Electrum, 42

## F

faktury, 72, 81, 331  
format „Typ-Długość-Wartość”, 307  
fraza mnemoniczna, 46  
funkcja  
  docker-compose, 111  
  skrótów, hash function, 26  
funkcje opt-in, 356

## G

generowanie  
  adresu wielopodpisowego, 169  
  kluczy, 243, 245  
  współdzielonych sekretów, 245  
grafy  
  kanałów, 275, 290, 314  
  niepewność, 292  
  utrzymywanie, 283  
  sieci Lightning, 346  
  skierowane, 275

## H

historia transakcji, 85  
HTLC, hash time-locked contracts, 203, 215

## I

- identyfikatory
  - transakcji, 367
  - węzłów, 164
  - wyjść, 368
- implementacja referencyjna Bitcoin Core, 125
- informacje
  - o kanale płatności, 59
  - o licencjach, 395
- infrastruktura klucza publicznego, 314
- instalowanie
  - c-lightning, 99
  - kontenera LND, 104
  - platformy Docker, 377
  - portfela Lightning, 43
  - serwera Eclair, 109
  - węzła, 127
    - Bitcoin, 121
    - Lightning, 121
- interfejs API, 137
- izolacja procesu, 128

## J

- jednostka monetarna, 87
- język Bitcoin Script, 355

## K

- kanały
  - akceptowanie, 351
  - graf, 290
  - informacje, 59
  - kontrakty HTLC, 215
  - Lightning, 50, 53, 161
  - naruszenie protokołu, 71
  - nieogłoszone, 350
  - ogłaszanie, 67
  - opłaty, 294
  - płatności, 26, 59, 61, 160
    - budowanie, 163, 165, 356
  - podwójnego finansowania, 170
  - rebalancing, 150
  - wychodzące, 146
  - wysyłanie płatności, 175
  - zamykanie, 68, 150, 188
    - wymuszone, 69
    - wzajemne, 69

- zarządzanie, 146
- zmiany stanu, 182
- kary, 187
- klasteryzacja
  - podmiotów on-chain Bitcoina, 344
  - węzłów off-chain w sieci Lightning, 345
- klucze
  - odwołania, 180
  - prywatne, 360
  - publiczne, 314, 360, 371
  - sesji, 244
  - węzłów, 163
- kod źródłowy
  - c-lightning, 100
  - Eclair, 109
  - LND, 104, 105
- kodowanie
  - bech32, 330
  - kanoniczne TLV, 308
  - TLV, 308
  - typów, 305
  - żądań płatności, 329
- kompilowanie kodu
  - c-lightning, 100
  - Eclair, 109
  - LND, 106
- komunikacja peer-to-peer, P2P, 74, 80, 158
- komunikat
  - accept\_channel, 168, 386
  - announce\_signatures, 392
  - channel\_announcement, 278, 391
  - channel\_update, 282, 392
  - closing\_signed, 189, 388
  - commitment\_signed, 183, 389
  - error, 383
  - funding\_created, 173, 386
  - funding\_locked, 387
  - funding\_signed, 174, 387
  - gossip\_timestamp\_range, 394
  - init, 382
  - node\_announcement, 276, 391
  - open\_channel, 167, 385
  - ping, 384
  - pong, 384
  - query\_channel\_range, 393
  - query\_short\_chan\_ids, 393
  - reply\_channel\_range, 394
  - reply\_short\_chan\_ids\_end, 393

- revoke\_and\_ack, 184, 390
- shutdown, 189, 387
- update\_add\_htlc, 256, 260
- update\_add\_HTLC, 218, 388
- update\_fail\_htlc, 230, 231, 296, 389
- update\_fail\_malformed\_htlc, 390
- update\_fee, 390
- update\_fulfill\_htlc, 230, 296, 389
- komunikaty
  - „Typ-Długość-Wartość”, 306
  - o błędach, 263
  - odbieranie, 325
  - odszyfrowywanie, 325
  - Protobuf, 306
  - protokołu
    - komunikacyjnego, 380
    - plotkarskiego, 276
  - przepływ, 216, 228
  - szyfrowanie, 325
  - typy wysokopoziomowe, 305
  - wysyłanie, 325
  - z kategorii
    - ustanawianie połączenia, 382
    - komunikowanie błędów, 383
    - żywność połączenia, 383
    - finansowanie kanału, 384
    - zamykanie kanału, 387
    - obsługa kanału, 388
    - ogłaszanie kanału, 391
    - synchronizacja grafu kanałów, 393
- konfiguracja
  - sieci, 131
  - sieci Docker, 97
  - węzła, 130
- kontener
  - bitcoind, 93, 97
  - c-lightning, 97
  - Docker, 91
    - serwera Eclair, 106
    - węzła LND, 102
  - Eclair, 106, 107
- kontenery
  - budowanie, 93, 96
  - uruchamianie, 97
- kontrakty HTLC, 203
  - aktualizacja zobowiązań, 216
  - dodawanie, 217
  - kooperatywne, 212
  - przekazywanie płatności, 216, 217
  - przepływ komunikatów, 228
  - realizacja, 226
  - rozliczanie, 230
  - rozszerzanie, 206
  - usuwanie, 228, 230
  - w Bitcoin Script, 204
  - w transakcjach zobowiązania, 219
  - weryfikowanie, 260
  - wielokrotne, 225
  - zabezpieczanie, 209
- kontrola, 42
- kooperatywne kontrakty HTLC, 212
- kopie zapasowe węzłów i kanałów, 138
- krzywa eliptyczna Diffiego-Hellmana, 245
- kupowanie, 54
- kwadrant portfeli Lightning, 40

## L

- Lamassu Industries AG, 396
- Lightning Network Daemon, LND, 102
- Lightning Network, LN, 15
- Lightning Protocol Suite, 157
  - atomowy routing płatności, 192
  - kanały płatności, 160
  - obsługa kanału, 215
  - protokół plotkarski, 267
  - przekazywanie płatności, 215
  - routing cebulowy, 233
  - szyfrowany transport komunikatów, 313
  - warstwa komunikatów, 303
  - zestaw protokołów, 157
  - znajdowanie ścieżek, 284
  - żądania płatności, 327
- limit czasu, 212
- LND, Lightning Network Daemon, 102
  - instalowanie kontenera, 104
  - kod źródłowy, 104, 105
  - kompilowanie kodu, 106
  - projekt węzła, 102
  - uruchamianie kontenerów, 103
- lndmon, 153

## **Ł**

- ładowanie bitcoinów, 46
- ładunek
  - kanału, 240
  - przeskoku, 241, 242
    - opakowywanie, 253, 254
    - owijanie, 249
    - usuwanie, 259
    - zaciemnianie, 257
  - węzła, 240
- łańcuch
  - bloków, 33
  - transakcji, 366, 372
- łączenie transakcji, 171

## **M**

- macierz zgodności bitów funkcji, 309
- magazyn
  - danych, 121
  - kluczy, 41
- metadane, 73
- milisatoshi, 86
- mnemoniczna fraza, 45
- monitorowanie dostępności węzłów, 144
- MPP, multipart payments, 298
- Mynode, 123

## **N**

- napiwek, 201
- naruszenie protokołu, 71
- narzędzie
  - BTCPay Server, 124
  - lndmon, 153
  - Loop, 142
  - Mynode, 123
  - RaspiBlitz, 122
  - Ride The Lightning, 153
  - ThunderHub, 154
  - Umbrel, 123
- Neutrino, 41
- nieodwracalność, 87
- niepewność
  - płynności, 293
  - sald, 287
  - w grafie kanałów, 292
- niezawodność działania węzła, 117

- Noise, noise protocol framework, 315
  - noise\_XK, 316
  - uzgadnianie, 316
  - notacja, 316
  - w trzech aktach, 318

## **O**

- obciążenie zwrotne, refund transaction, 32
- ocena prywatności, 334
- ochrona prywatności użytkowników, 350
- odbieranie komunikatów, 325
- odszyfrowywanie komunikatów, 325
- odwołane zobowiązania, 186
- odwoływanie, 184
  - transakcji, 177
- ogłaszanie kanału, 67
- ograniczenia pojemności, 85
- on-chain kontra off-chain, 26
- opakowywanie
  - cebuli, 253
  - ładunku przeskoku, 253, 254
- opcje zapytań SRV, 274
- Open Source, 88
- Open System, 88
- operacje
  - asynchroniczne, 86
  - submarine swap, 142
  - synchroniczne, 86
- opłaty, 294
  - za routing, 83, 151
  - za wydobycie, 83
- optymalizacja skrótów, 210
- orkiestracja kontenerów Docker, 111
- ostateczność płatności, 87
- oszukiwanie, 65, 177, 185, 187
- otwieranie kanałów, 113
- owijanie
  - cebuli, 248
  - ładunku przeskoku, 249

## **P**

- pakiety, 324
- pakiety cebuli, 255
- peer-to-peer, P2P, 74, 158
  - bootstrapping, 270
  - szyfrowanie komunikacji, 80
- pierwszy portfel, 37



- plastyczność, 172
  - płatności, 26, 59, 81
    - algorytm przekazywania, 79
    - atomowe, 201
    - dostarczanie, 74, 289
    - jednokanałowe, 216
    - kanały, 26, 59, 61, 160
    - Keysend, 265, 266
    - lokalne, 216, 231
    - MPP, 299
    - o dużej wartości, 85
    - o małej wartości, 85
    - on-chain, 26
    - prywatne Lightning, 84
    - przekazywanie, 215, 216
    - routing atomowy, 192
    - routowanie, 60, 113, 192, 216
    - wieloczęściowe, multipart payments, 298
    - wielokanałowe, 216
      - próby i błędy, 300
    - wieloprzeskokowe, 201
    - z wykorzystaniem HTLC, 217
    - zablokowane, 265, 296
    - żądania, 327
  - płynność
    - przychodząca, 149
    - wychodząca, 149
  - podpisy cyfrowe, 26, 363
    - typy, 364
  - podział salda, 175
  - polecenia platformy Docker, 378
    - docker-compose, 111
  - port 9735, 133
  - portfel Lightning, 37, 39
    - instalowanie, 43
    - pobieranie, 43
    - tworzenie, 44
  - portfele popularne, 41
  - porty
    - automatyczne przekierowywanie, 133
    - ręczne przekierowywanie, 135
  - potwierdzenie, 84
    - transakcji, 57
  - prawdopodobieństwo dostarczenia płatności, 293
  - praworządność, 28
  - prefiks czytelny dla człowieka, 329
  - preobraz, 73
  - problem
    - plastyczności, 172
    - transportu satoshi, 287
  - procedura otwierania kanałów, 63
  - proces oceny prywatności, 334
  - projekt
    - c-lightning, 96
    - węzła Eclair Lightning, 106
    - węzła LND, 102
  - propagacja HTLC, 226
  - Protobuf, 306
  - protokoły, *Patrz także* Lightning Protocol Suite
    - Lightning Network, 157
    - łącza fizycznego, 303
    - Noise, 315, 316
    - partnerskie, 165
    - plotkarskie, 267
      - komunikaty, 276
      - peer-to-peer, 74
    - TLS, 315
    - transportu komunikatów, 316
    - uczciwości, 28–31, 200
    - uczciwości oparte na teorii gier, 28
    - zaufania bez pośredników, 28
    - znajdowania ścieżek, 76
  - prymitywy zabezpieczeń, security primitives, 30
  - prywatność, 333
    - ocena prywatności, 334
    - w sieci Bitcoin, 336
    - w sieci Lightning Network, 336
  - przechowywanie
    - frazy mnemonicznej, 46
    - kluczy, 44
  - przekazywanie płatności, 79, 215, 216
  - przekierowywanie portów
    - automatyczne, 133
    - ręczne, 135
  - przepływ
    - komunikatów, 165, 216, 228
    - protokołu, 316
  - przycisk Send, 56
  - przypadki użycia, 35
- ## R
- rachunek Lightning, 55
  - RAM, 119
  - ramki łącza fizycznego, 304
  - RaspiBlitz, 122

- rebalancing kanałów, 150
- refundacja, 170
- Regtest, 93
- rekordy TLV cebuli, 265
- relacja mapa-obszar, 291
- rezerwacja kanału, 188
- Ride The Lightning, 153
- rotacja kluczy komunikatów, 326
- routing, 26, 194, 195, 351
  - atomowy płatności, 192
  - cebulowy, 77, 233
    - budowanie warstw, 235
    - cebule o stałej długości, 247
    - generowanie kluczy, 243
    - obieranie warstw, 237
    - oparty na kontraktach HTLC, 238
    - owijanie warstw cebuli, 247
    - wybieranie ścieżki, 234
    - wysyłanie cebuli, 256
    - zabezpieczanie, 252
    - zwracanie błędów, 262
  - oparty na źródle, 235
  - opłaty, 151
  - płatności, 60, 113, 192, 216
- rozliczenie kontraktów HTLC, 203, 207, 208
  - natychmiastowe, 84
- rozszerzalność
  - kontraktu HTLC, 206
  - protokołu, 309
  - TLV, 356
- równoważenie złożoności i kontroli, 42
- ryzyko kontrahenta, 87

## S

- satoshi, 86
- Segregated Witness, 172
- sekrety współdzielone, 245
- serializacja, 329
- serwer
  - BTCPay, 395
  - Eclair, 106
  - VPS, 118
- sieć
  - Docker
    - konfiguracja, 97
  - kanałów płatności, 194
  - Lightning Network
    - wizualizacja, 285
    - tworzenie, 110

- P2P
  - bootstrapping, 270
  - Tor, 134
- skalowanie łańcuchów bloków, 33
- skrót, 361
  - HMAC, 258
  - optymalizacja, 210
  - płatności, payment hash, 73, 203
- skrypty
  - blokady czasowej, 374
  - blokujące, 371
  - odblokowujące, 371
  - wielopodpisowe, 373
  - z wieloma warunkami, 374
- słowa mnemoniczne, 45
- sprzęt, 119
- SRV
  - opcje zapytań, 274
- stany kanału płatności, 182
- sterowanie przepływem w skryptach, 375
- struktura
  - grafu, 349
  - komunikatów, 382
  - zaszyfowanego komunikatu, 325
- system
  - operacyjny, 126
  - sądowy, 85
- szyfrowanie
  - i wysyłanie komunikatów, 313, 325
  - komunikacji, 80
  - transportu Lightning, 316

## Ś

- ścieżka, 76, 234
  - płatności, 238
- szczegółowa, 239
- znajdowanie, 76, 82, 193, 284–289, 295
- środowisko programistyczne Lightning, 90

## T

- taksonomia mechanizmów aktualizacji, 311
- TCP/IP, 164
- TCP/Tor, 164
- ThunderHub, 154
- TLS, 315
- TLV, 308, 310
- Tor, The Onion Router, 134

- transakcje, 27, 81
  - asymetryczne, 178
  - Bitcoin, 364
  - finansowania, funding transactions, 32, 62, 169
  - nieaktualne, 177
  - on-chain i off-chain, 161
  - przedawnione, 177
  - publiczne Bitcoin, 84
  - publikowanie, 174
  - wstępnie podpisane, 171
  - wzajemnego zamknięcia, 190
  - zobowiązania, commitment transactions, 63, 180
    - asymetryczne, 178
    - kontrakty HTLC, 219
  - zwrotu, 171
- tryb
  - „bez zaufania”, 201
  - off-chain, 141, 203, 207, 208
  - on-chain, 140, 203
  - uproszczony, 125
- TxID, 367
- tymczasowość sieci Lightning Network, 348
- typ BigSize, 307
- typy
  - komunikatów, 380
  - podpisów, 364

## U

- uczciwość, 28, 200
- ujawnianie sald kanałów, 339
- Umbrel, 123
- UPnP, 133
- uruchamianie
  - kontenera, 378
    - bitcoind, 97, 103, 107
    - c-lightning, 97
    - Eclair, 107
    - LND, 103
  - Lightning Network, 112
  - węzła, 129
  - węzła w warunkach domowych, 118
- usługi w tle, 128
- usprawnienia protokołu Bitcoin, 355
- ustanawianie kanału, 165
- usuwanie
  - kontenera według nazwy, 378

- kontraktu HTLC, 230
- ładunku przeskoku, 259
- utrzymywanie podpisanych transakcji, 170
- uzgadnianie, 316
  - w trzech aktach, 318

## V

- VPS, 118

## W

- warstwa
  - komunikatów, messaging layer, 158, 303
  - peer-to-peer, P2P, 158
  - płatności, 158
  - połączenia sieciowego, 158
  - routingu, 158
- wejścia i wyjścia, 364
- weryfikacja
  - kontraktu HTLC, 260
  - skrót, 205
- węzeł, 26
  - Bitcoin, 41
  - Eclair Lightning, 106
  - Lightning, 38, 41, 89, 110
    - bezpieczeństwo węzła, 136
    - czas pracy, 142
    - dostępność, 142
    - implementacja, 126
    - instalowanie, 121, 127
    - interfejs API, 137
    - konfiguracja, 130
    - kopie zapasowe, 138
    - niezawodność działania, 117
    - przełączanie konfiguracji serwerowej, 120
    - rodzaje sprzętu, 117
    - tryb uproszczony, 125
  - uruchamianie w warunkach domowych, 118
  - utrzymywanie w chmurze, 118
  - wybór implementacji, 126
  - wybór platformy, 117
  - wybór systemu operacyjnego, 126
  - wymagany sprzęt, 119
  - zarządzanie, 153
  - zarządzanie kanałami, 146
- LND, 102
- RaspiBlitz, 122

- węzły
  - identyfikatory, 164
  - łączenie, 165
  - partnerskie, 269
  - projektowanie, 102, 106
  - zarządzanie, 153
- wielokrotne kontrakty HTLC, 225
- wielopodpisowość, 162
- wiersz polecenia, 90
- wieże strażnicze, watchtowers, 144
- wirtualne serwery prywatne, VPS, 118
- własność sieci Bitcoin, 162
- współwłasność, 162
- wybieranie ścieżki, 234
- wybór
  - implementacji węzła, 126
  - platformy, 117
  - systemu operacyjnego, 126
- wydatki to\_self, 179
- wygaśnięcie ważności, 230
- wyjścia UTXO, 82
- wykonywanie
  - płatności lokalnej, 231
  - polecenia w kontenerze, 378
- wykrywanie uaktualnień, 309
- wymiatanie
  - środków, 140
  - typu „submarine swap”, 141
- wysokopoziomowe typy komunikatów, 305
- wysyłanie, 26
  - dowolnych kwot, 85
  - komunikatów, 325
  - płatności, 175
- wyświetlanie
  - listy uruchomionych kontenerów, 379
  - obrazów platformy Docker, 379
- wyznaczanie tras, 76

## Z

- zabezpieczanie
  - kontraktów HTLC, 209
  - routingu cebulowego, 252

- zablokowane
  - płatności, 265, 296
  - wydatki, 179
- zablokowanie bitcoinów, 163
- zachęty ekonomiczne, 349
- zaciemnienie ładunku przeskoku, 257
- zagrożenia, 140
- zamykanie kanałów, 68, 150
- zapytania SRV, 274
- zarządzanie
  - kanałami, 146, 165
  - siecią Bitcoin, 162
  - węzłami, 153
- zatrzymywanie i uruchamianie kontenera, 378
- zaufane podmioty zewnętrzne, 28
- zaufanie, 27, 81, 87
- zdobywanie bitcoinów, 47
- zgodność wstecz i w przód, 306, 310
- zmiana wyjść, 82
- zmiany stanu kanału, 182
- znajdowanie ścieżek, pathfinding, 76, 82, 193, 284, 289, 295
  - teoria grafów, 286
  - złożoność, 288
- zobowiązania, 176, 181, 184, 219, 221, 224
  - aktualizacja, 216
  - bieżące, 186
  - odwołane, 186
  - odwoływanie, 222, 229
  - uznawanie, 222
  - z wyjściem HTLC, 220
- zwracanie błędów, 262

## Ż

- żądania płatności, 327, 328
  - bech32, 330
  - interpretacja, 329
  - kodowanie, 329
  - segment danych, 330
  - serializacja, 329

# PROGRAM PARTNERSKI

— GRUPY HELION —



1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW w działający bankomat!

**Dowiedz się więcej i dołącz już dzisiaj!**

<http://program-partnerski.helion.pl>

GRUPA  
**Helion** 

# Poznaj potencjał LN i zostań mistrzem technologii przyszłości!

Potencjał blockchaina jest ogromny i zapewne wciąż nie znamy jego wszystkich możliwych zastosowań. Jest to dosyć nowa, dynamicznie rozwijająca się technologia, już teraz jednak dostrzegalne są pewne jej ograniczenia. Problemem jest na przykład skalowalność bitcoina. Rozwiązaniem wydaje się Lightning Network (LN), protokół drugiej warstwy, który zapewnia niemal natychmiastowe transakcje w systemie bitcoina. Umożliwia również zwiększenie szybkości i prywatności zawieranych transakcji, a przy tym pozwala na zmniejszenie opłat.

To książka przeznaczona dla osób, które chcą zrozumieć działanie Lightning Network i wykorzystać możliwości tej technologii we własnych aplikacjach. Z lektury skorzystają programiści, architekci systemowi i inżynierowie. Omówiono tu podstawy funkcjonowania sieci LN i sposoby jej użycia w praktyce. Przedstawiono również zasady oprogramowywania węzłów Lightning, ich implementacji i konfiguracji, a także zagadnienia budowania kanałów płatności w systemie bitcoina, obsługi kanałów i przekazywania płatności. Zaprezentowano też informacje dotyczące bezpieczeństwa i prywatności w sieci Lightning Network. Poszczególne zagadnienia zostały zilustrowane praktycznymi przykładami kodu w językach Go, C++, Python i przy użyciu wiersza poleceń uniksowych systemów operacyjnych.

## W książce między innymi:

- technologia Lightning Network a skalowanie blockchaina
- standardy stosowane w Lightning Network
- warstwy zestawu protokołów Lightning Network
- portfele i węzły oraz ich obsługa
- kanały płatności Lightning, routing cebulowy i protokół plotkarski
- kanały płatności od nadawcy do odbiorcy w trybie off-chain

**Andreas M. Antonopoulos** jest autorem, nauczycielem i jednym z czołowych ekspertów w dziedzinie bitcoina i otwartego blockchaina.

**Olaoluwa Osuntokun** jest programistą bitcoina, naukowcem i dyrektorem technicznym firmy Lightning Labs.

**René Pickhardt** jest konsultantem w dziedzinie inżynierii danych i naukowcem. Dzięki swojej pracy nad routingiem płatności odkrył tanie i niezawodne przepływy płatności, określane jako „płatności Pickhardta”.

**Helion**  
helion.pl  
HELION SA  
ul. Kosciuszki 1c  
44-100 Gliwice  
tel. 32 230 938 63  
helion@helion.pl

KOD KORZYŚCI  
Sięgnij po więcej! ▶



ISBN 978-83-283-9322-6



Cena: 119,00 zł