Angriffsvektoren und mögliche Schutzmaßnahmen von Distributed Ledger Technologien

Sicherheit von IT-Systemen

Lars Zimmer

Sommersemester 2020

Inhaltsverzeichnis

1. Einleitung

Ihre Anfänge fand die Distributed Ledger Technologie (dt.: verteilte Kassenbücher) bereits 2002 mit der von David Maziéres and Dennis Shasha veröffentlichten wissenschaftlichen Arbeit *Building secure file systems out of Byzantine storage [Quelle 1]*.
Wenige Jahre später griff der unter dem Pseudonym Satoshi Nakamoto bekannte Author des Bitcoin-White-Papers die Idee eines Peer-to-Peer-basierten verteilten Systems zur Tätigung von Transaktionen auf. 2009 wurde sie mit der Veröffentlichung der ersten Referenzimplementierung von Bitcoin zur Realität.
Seitdem haben Distributed Ledger Technologien in vielen Sektoren Einzug gehalten und werden zu vielfältigen, innovativen Zwecken eingesetzt.
Wie jede Vorreitertechnologie, bringen die neuartigen Aspekte von Digital Ledger Technology, die sie als Lösungsansatz für bisher unlösbare Probleme interessant machen, aber auch neue Sicherheitsrisiken mit sich.
Im Folgenden soll die Digital Ledger Technologie aus sicherheitstechnischer Sicht genauer betrachtet werden.

Zunächst wird die grundlegende Funktionsweise und der Aufbau von DLT genauer erläutert.
Anschließend wird sowohl auf bereits aus anderen IT-Systemen bekannte Angriffsvektoren, als auch auf neuartige, DLT-spezifische und teils Protokoll-spezifische Sicherheitsaspekte eingegangen.
Letztlich werden mögliche Schutzmaßnahmen gegen die untersuchten Angriffsvektoren beschrieben.

2. Grundlagen

2.1 Distributed Ledger Technologien (DLT)

Distributed Ledger Technologie, kurz DLT, bezeichnet eine Technik, um Transaktionen in einem verteilten System durchzuführen und zu speichern, wobei Transaktionen nicht zwangsläufig finanzieller Natur sein müssen.
Im Prinzip handelt es sich hierbei um eine synchronisierte Datenbank, die auf mehreren Geräten gleichzeitig gespeichert ist.
Verschiedene Methoden, sogenannte Konsens-Protokolle, sorgen dafür, dass die verschiedenen Kopien zu einem Konsens bezüglich des aktuellen Stands aller Transaktionen kommen.

2.2 Anwendungen

Die wohl bekannteste Anwendung von DLTs ist die Blockchain. Sie bildet die Grundlage für Kryptowährungen wie Bitcoin oder Ethereum.
Neben diesem Anwendungsfall werden sie in verschiedenen Bereichen wie der Logistik, insbesondere bei der Supply-Chain, oder der Finanzindustrie verwendet.

Viele große Unternehmen befinden sich bezüglich verteilter Ledger aber noch in der Planungs- und Testphase weshalb sich über viele Anwendungsfälle bislang nur spekulieren lässt. So hat Volkswagen kürzlich ein Patent angemeldet um Unique Identifier für Fahrzeuge zu erzeugen und diese in einem Ledger zu speichern [Quelle 2].

Plattformen wie Ethereum ermöglichen das Nutzen von Smart Contracts, digitaler Verträge, die automatisiert vordefinierter Ereignisse unter der Erfüllung von Bedingungen ausführen.

Außerdem ermöglicht die Persistenz von DLTs das Umgehen von Zensur. So wurden in China bereits mehrere staatskritische Flugschriften als Metadaten in der Ethereum-Blockchain veröffentlicht, um der Zensur zu entgehen. [Quelle 3]

Allgemein könnte immer dann, wenn verschiedene Parteien Vertrauen in Daten aus unterschiedlichen Quellen haben müssen, auf DLT zurückgreifen. Dies kann bei korrekter Umsetzung gewährleisten, dass jede Partei über die selben Daten verfügt und diese nicht nachträglich manipuliert werden können.

Weitere Vorteile stellen die Verfügbarkeit, die Skalierbarkeit, die Fehlertoleranz und die Unveränderbarkeit dar.

2.3 Varianten

2.3.1 Federated, Permissioned/Private, Permissionless/Public, Hybrid

Es gibt verschiedene Arten von DLTs:

Federated DLTs werden von einem Konsortium oder einer Gruppe von Firmen verwendet, wenn nur diese Teilnehmer direkt mit dem Ledger interagieren können sollen. Ein möglicher Anwendungszweck einer solchen Technologie wäre beispielsweise eine komplexe Lieferkette die mehrere Unternehmen umfasst.

Private DLTs können nur von verifizierten Nutzern verwendet werden und werden meist aufgrund von unveränderbaren Einträgen und einer hohen Verfügbarkeit genutzt.

Permissioned DLTs bezeichnen Ledger, bei denen unterschiedliche Nutzergruppen verschiedene Rechte haben. So könnte es beispielsweise reguläre Nutzer geben, die nur Transaktionen tätigen können, eine weitere Personengruppe könnte berechtigt sein, Transaktionen zu verifizieren. Eine weitere, administrative Gruppe könnte Gewichtungen festlegen, um den regulären Ablauf des Konsens-Protokolls zu umgehen und somit einen weniger stark verifizierten Ledger-Status durchzusetzen. Je nach Definition kann sich die “permissioned” Eigenschaft aber auch auf die Einsehbarkeit von Daten beziehen.

Public DLTs sind öffentlich. Jedem Nutzer ist es freigestellt, daran teilzunehmen, sich Transaktionen anzusehen, zu tätigen oder zu empfangen. Häufig wird ein Anreiz für das zur Verfügung stellen von Ressourcen wie Rechenleistung geboten.

Permissionless DLTs bezeichnen Ledger bei denen alle Nutzer gleichgestellt sind und über gleiche Rechte verfügen.

In der Regel werden Permissioned und Private sowie Public und Permissionless kombiniert, da es bei einer privaten Datenverwaltung häufig Personen mit verschiedenen Berechtigungen gibt und öffentliche Ledger sich meist selbst regulieren können sollen, um Verwaltungsaufwand zu sparen.

Unter Hybrid DLTs versteht man meist eine Kombination aus einem Privaten und öffentlichen Ledger. Dies ermöglicht eine höhere Skalierbarkeit als bei Public DLTs, da teilweise bereits eine Verifikation eines Nodes des privaten Netzes ausreicht, um

eine Transaktion zu validieren. Ebenso ist es denkbar, Daten partiell im privaten Netz zu speichern, um die Privatsphäre zu erhöhen.

2.3.2 Blockchain

Die Blockchain stellt die populärste Form eines DLTs dar. Sie besteht, wie der Name bereits andeutet, aus einer Kette von Blöcken. Vereinfacht besteht jeder Block aus einem Header, dem Hash des Headers seines Vorgängers und einem Hash-Baum von Transaktionen. Dadurch, dass jeder Block den Hash des vorhergehenden Block Headers beinhaltet entsteht eine Verkettung von Blöcken mit Transaktionen.

2.3.3 Gerichtete azyklische Graphen (DAG)

DAG (Directed Acyclic Graph, zu deutsch: gerichteter azyklischer Graph) bezeichnet einen Graphen, dessen Kanten eine Richtung aufweisen und unter Berücksichtigung dieser Richtung keine Zyklen beinhaltet.

Im Kontext von DLTs könnte man eine Blockchain ebenfalls als DAG bezeichnen, da hier jeder Block einen Verweis auf den vorherigen Block beinhaltet und somit eine Aneinanderreihung von Knoten bis hin zum Ursprungsblock bilden.

Im Kontext von DLTs versteht man unter DAG meist einen Lösungsansatz, der die Vorzüge der Blockchain mit einer besseren Performanz vereinen soll.

Dies soll erreicht werden, indem anstelle einer einfachen Kette mehrere Kanten parallel hinzugefügt und verifiziert werden können.

3. Konsens-Protokolle

3.1 Proof of Work

Proof of Work, kurz PoW, ist ein Konsens-Protokoll, bei dem Rechenleistung aufgebracht werden muss, um ein kryptografisches Rätsel zu lösen.

Im Fall von Bitcoin wird der Hashcash Algorithmus mit SHA256 verwendet, um mit einem Brute-Force-Ansatz eine sogenannte “nonce” (eine Zahl) zu finden, die zu einem Hash führt, der den Anforderungen des Netzwerkes entspricht.
Die Anforderung des Netzwerks wird dynamisch über die Anzahl führender Nullen im Hash gebildet und als “Difficulty” (dt.: Schwierigkeit) bezeichnet. Die Difficulty sorgt dafür, dass die Rate gelöster Rätsel und damit neuer Blöcke gleich bleibt. [Quelle 4]

Angenommen eine Person A transferiert Person B Bitcoin, so würde die Wallet (eine Art digitale Geldbörse) von Person A eine Transaktion broadcasten. Nodes im Bitcoin Netzwerk können diese Transaktion bereits einsehen und mit ihrem Speicher der Blockchain abgleichen, um zu überprüfen, ob die Transaktion legitim ist. Sie ist allerdings noch kein Teil der Blockchain. Um Teil der Blockchain zu werden, müssen Miner (Personen die Rechenleistung investieren um neue Blöcke zu erzeugen) diese Transaktion aus dem Pool nicht bestätigter Transaktionen auswählen und in einen von ihnen gefundenen Folgeblock schreiben.
Die Auswahl der Transaktionen geschieht in der Regel über die Höhe der Gebühr, die bei dem Tätigen der Transaktion gezahlt wird.
Nun ist die Transaktion zwar Teil der Blockchain, es kann allerdings passieren, dass parallel ein anderer Miner einen neuen Block mit anderen Transaktionen gefüllt hat. Damit es nicht zu einer Zersplitterung der Blockchain kommt, übernehmen die Nodes, wenn vom Besitzer nicht anders angegeben, immer die Längste Kette.
So ist es also insbesondere bei größeren Transaktionen sinnvoll abzuwarten, bis die Transaktion bereits seit mehreren Blöcken in der Blockchain integriert ist, bevor man davon ausgeht, dass sie tatsächlich erfolgreich war.

PoW steht allerdings aufgrund der Ineffizienz bezüglich natürlicher Ressourcen in der Kritik. Das Minen mit normaler Computer Hardware ist insbesondere bei Bitcoin längst nicht mehr rentabel. Stattdessen wird spezialisierte Hardware (Anwendungsspezifische Integrierte Schaltung, kurz ASIC) gebaut, die keinem anderen Verwendungszweck, als dem Lösen solcher kryptografischen Rätsel, dient. Es kommt zu einer Art Wettrüsten, bei der Hardware nach wenigen Jahren bereits kaum noch sinnvoll nutzbar ist.

Neben den Ressourcen, die in Hardware fließen, ist auch der Stromverbrauch nicht zu verachten. So verbraucht derzeit (Stand Juli 2020) das Minen von Bitcoin etwa so viel Strom wie Griechenland [Quelle 5].

3.2 Proof of Stake

Proof of Stake, kurz PoS, ist ein Konsens-Protokoll, bei dem der Ersteller des nächsten Blocks auf Basis verschiedener Zufallsfaktoren und ihrem sogenannten “Stake” ausgewählt wird.
Der Stake bezieht in der Regel Aspekte wie Anteile der Kryptowährung und deren Lagerzeit ein.
Das System basiert darauf, dass hoch investierte Personen einen Anreiz am Erfolg des Systems haben und daher keine Aufspaltung der Kette verursachen würden.
Lösungsansätze für dieses Risiko beinhalten das Absichern alter Blöcke im Protokoll, beispielsweise durch zentralisierte Checkpoints (bei Peercoin) oder Einschränkungen bei der Reorganisation alter Blöcke im Protokoll (bei Nxt).
Beide Ansätze stellen allerdings eher eine Risikominderung als eine tatsächliche Lösung dar.
[Quelle 6]

3.3 Delegated PoS

Delegated Proof of Stake, kurz dPoS, ist eine Weiterentwicklung von PoS, bei der nicht mehr jeder Anteilhaber einen Block erstellen kann.
Stattdessen gibt es eine Gruppe von Delegierten, die üblicherweise für ihre Arbeit vergütet und für böswilliges Verhalten bestraft wird. Meist müssen Blöcke von mehr als einem Delegierten verifiziert werden.
Die Auswahl der Delegierten kann variieren. Es gibt Systeme, bei denen Nutzer aufgrund ihres Stakes zu Delegierten werden, andere haben eine Art Wahlsystem, bei dem der Stake der Wählenden proportional zum Einfluss auf die Wahl steht.
[Quelle 7]

3.4 Cellular Consensus

Cellular Consensus, bezeichnet ein Konsens-Protokoll, bei welchem ein Abstimmungsprozess als zellularer Automat modelliert wird.
Nodes im Netzwerk überwachen die Entscheidung ihrer Nachbarn.
Im Konfliktfall wird die Transaktion übernommen, die von über der Hälfte der Nachbarn als korrekt angesehen wird.

Da Konsens lokal immer mit den gleichen Nachbarn getroffen wird, kann es bei mehreren böswilligen Nachbarn zu einer Abweichung von lokalem und globalem Konsens kommen.

Ein Lösungsansatz ist das Verwenden eines Reputations Modells beim Peering-Prozess. Hinzu kommt die zufällige Auswahl der Nachbarn, die einen Fall von mehr böswilligen als gutwilligen Nachbarn unwahrscheinlich macht.
[Quelle 8]

3.5 Fast Probabilistic Consensus

Fast Probabilistic Consensus ist eine Abwandlung des Cellular Consensus Protokolls, bei dem die Abstimmung nicht mehr asynchron zwischen den einzelnen Nachbarn verläuft, sondern in Runden unterteilt ist.
In jeder Runde wird eine zufällige Teilmenge von Nodes gewählt und deren Meinung abgefragt. Die Meinung eines Nodes richtet sich nach der Mehrheit, wobei diese einen Schwellenwert, basierend auf einer dezentralen Zufallszahlenfolge, benötigt. Dies soll vor einem Angreifer, der die Konsensfindung beeinflussen will, schützen.
[Quelle 9]

3.6 Byzantine Fault Tolerance

Byzantinische Fehler (engl.: Byzantine faults) beschreiben einen Fehlerzustand in einem in der Regel verteilten Computersystem, bei dem unvollständige Informationen darüber vorhanden sind, ob einzelne Komponenten fehlerhaft sind oder nicht.
Die Bezeichnung leitet sich vom *Byzantine generals problem* ab, bei dem sich zwei byzantinische Generäle auf einen gemeinsamen Angriffszeitpunkt einigen müssen, zur Übermittlung von Nachrichten untereinander aber nur auf nicht-vertrauenswürdige Boten zurückgreifen können.
Im Kontext von DLT beschreibt Byzantine Fault Tolerance, kurz BFT, die Eigenschaft eines Systems, einen Konsens in einem verteilten Netzwerk zu finden, obwohl Nodes mit fehlerhaften Informationen oder gar nicht antworten, weil sie beispielsweise durch böswillige Dritte übernommen wurden.
BFT und der daraus in den späten 90ern von Barbara Liskov und Miguel Castro, abgeleitete Konsens-Algorithmus Practical Byzantine Fault Tolerance (pBFT) dienen als Grundlage vieler neuerer Algorithmen im Bereich der Distributed Ledger Technologien.
[Quelle 10]

4. Angriffsvektoren

4.1 Traditionelle Angriffe

4.1.1 Denial of Service Angriffe

Ein Denial of Service (kurz DoS) Angriff bezeichnet das Unterbinden von Zugriffen auf das Zielsystem. Dies kann beispielsweise durch das ferngesteuerte Abschalten des entsprechenden Gerätes verursacht werden, geschieht aber in den meisten Fällen durch eine Überlastung des Zielsystems mittels eines DDoS (distributed denial of service) Angriffs, also einem DoS Angriffes der von mehreren Geräten aus simultan durchgeführt wird.

DLTs sind zwar bereits von Natur aus weniger angreifbar durch DDoS Attacken als zentralisierte Systeme, gezielte Angriffe können aber dennoch große Schäden anrichten.
So können DDoS Angriffe nicht nur verwendet werden um Nodes temporär zu überlasten, sondern auch, um die Verwendung von Exchanges zu unterbinden.
Für Endnutzer hat dies zur Folge, dass sie vorübergehend nicht in der Lage sind, ihr auf dem Exchange eingelagertes Geld umzutauschen, zu versenden oder sich auszahlen zu lassen.
Für Exchanges bedeutet dies einen direkten Verlust von potentiellen Transaktions- und Handelsgebühren sowie einen Verlust an Vertrauen und Kunden.

Angreifer können dies direkt ausnutzen, indem sie bei noch funktionsfähigen Exchanges eine Short-Position einnehmen, bzw. auf einen fallenden Kurs spekulieren. Ein Hebel (vereinfacht: Gewinne und Verluste werden multipliziert) kann hierbei verwendet werden, um den Gewinn des Angreifers zu maximieren.
Ein solcher Angriff kann als "Malware-as-a-Service" ohne zusätzliches Wissen erworben und bereits für weniger als 1000€ pro Seite für mehrere Stunden aufrecht erhalten werden. Eine weitere Methode, aus einem DDoS Angriff Gewinn zu schlagen, stellt die Forderung von Lösegeld dar.
Aus Sicht eines Exchanges stellen solche Angriffe eine potenzielle, wenn auch illegale, Methode dar, um Konkurrenten Schaden zuzufügen.

Neben diesen recht typischen Motivationen für DDoS Angriffe, wäre es aber auch denkbar, Miningpools zu attackieren. Dies hätte zur Folge, dass die Mining-Rate stark gesenkt wird und es in der Konsequenz leichter wäre, an einer Sidechain (ein Alternativer Verlauf der Blockchain neben der momentan längsten Kette) zu arbeiten. Schafft man es, mehr Blöcke anzuhängen als der Teil der Hauptkette ab der Gabelung, so kann man alle Transaktionen ab der Gabelung revidieren und dadurch

sogenannte Double Spends, also das zweifache Ausgeben derselben Token, ermöglichen.

4.1.2 Hack von Dienstleistern

Auch bei Distributed Ledger Technologien gibt es viele verschiedene Dienstleister, mit denen interagiert wird. So gibt es Exchanges, bei denen man Währungen umtauschen kann, Wallets die zum Lagern und Versenden von Kryptowährungen benutzt werden oder aber auch Dienstleister, die eine Übersichtliche Darstellung und Suchfunktionen für alle bisher getätigten Transaktionen zur Verfügung stellen.

All diese Dienstleister stellen rentable Ziele für Angreifer dar.
Auf Exchanges lagern nicht selten Millionen oder sogar Milliarden von Euro in Form von verschiedenen Währungen und Nutzerdaten.
Wallets könnten manipuliert werden, um private Keys an den Hacker zu übermitteln und auch Webseiten zur Darstellung des Ledger-Status können manipuliert werden, um beim privaten Handel zu betrügen.

4.1.3 DNS cache poisoning

DNS cache poisoning, auch DNS spoofing, bezeichnet eine Methode zur Manipulation des Cache eines DNS-Resolvers. Dies ermöglicht es dem Angreifer, den Datenverkehr umzuleiten.
Generell gelten DLTs in diesem Kontext aufgrund ihrer vielen Anknüpfpunkte, meist unter verschiedenen Domains, als vergleichsweise sicher. Es ist jedoch in einigen Szenarien möglich, insbesondere neuen Teilnehmern im Netzwerk eine manipulierte Liste von Nachbarn zu übergeben. Es ist sowohl möglich, die Node-Liste von Endgeräten durch eine Manipulation der Client-Software auszutauschen, als auch den DNS Cache von existieren Nodes mittels Hacking zu verändern.
Ebenso wäre es denkbar, einen neuen Node aufzusetzen, um Endnutzer mit falschen Informationen zu versorgen. Hierfür müsste man aber zunächst eine gewisse Popularität aufbauen, um Nutzer dazu zu bewegen den eigenen Node zu verwenden.
[Quelle 11]

4.1.4 BGP Hijacks

BGP Hijacking beschreibt die Manipulation von Routing Tabellen des Border Gateway Protocol um Datenströme umzuleiten.
Mittels BGP Hijacking ist es nicht nur möglich, Nutzer von Diversen Dienstleistern wie online Wallets zu bestehlen, wie es im April 2018 mit "MyEtherWallet.com" der

Fall war [Quelle 12], sondern auch den Traffic von Mining-Pools umzuleiten [Quelle 13], um sich Hashpower zu verschaffen.
Einer Studie zufolge wäre es einem Angreifer durch das Hijacken von 100 Präfixen möglich, sich die Hälfte der Mining-Power von Bitcoin anzueignen und so einen Double Spend zu ermöglichen. [Quelle 14]

4.2 Konsens-Protokolle

4.2.1 51% Angriff

Der 51% Angriff beschreibt einen bei PoW-Algorithmen typischen Angriffsvektor. Ziel des Angriffes ist es, mittels 51% der Hashing-Power des gesamten Netzwerkes eine Gabelung der Blockchain zu verursachen um einen Double Spend zu erreichen. Ein solcher Angriff gilt insbesondere bei größeren Blockchains als sehr kostspielig und resultierte in der Vergangenheit meist in einem starken Vertrauensverlust, inklusive entsprechenden Kurseinbruch, und der Wahl einer neuen Gabelung, die vor dem Vorfall ansetzt.

4.2.2 Eclipse Angriff

Ein Eclipse Angriff ist ein Angriff, bei dem der Großteil der Nodes innerhalb eines Clusters böswillig agiert und andere Nodes isoliert. Hierdurch ist es möglich, den ein- und ausgehenden Datenstrom zu verändern, falsche Informationen weiter zu reichen und den lokale Status des Ledgers zu manipulieren. Hat jeder Node im Cluster den manipulierten Status des Ledgers angenommen, so wird auch jeder neue Node, der dem Cluster beitritt, der Sicherheitslücke ausgesetzt. [Quelle 11]

4.2.3 Spam

Spam im Kontext von DLTs bezieht sich in der Regel auf das übermäßige Versenden von Transaktionen, um das Netzwerk oder spezifische Nodes zu verlangsamen und ist somit vergleichbar mit einem DoS Angriff.

4.2.4 Mempool flooding

Der Mempool bezeichnet einen “Memory Pool”, der als temporärer Speicher nicht bestätigter Transaktionen dient. Beim Mempool flooding verschickt man eine große Zahl an Transaktionen, um die Transaktionskosten und damit die Belohnung der Verifizierenden (bei PoW i.d.R. Minern) zu erhöhen.

4.2.5 Block Withholding Angriff

Der Block Withholding Angriff beschreibt das Zurückhalten der Information über einen gefundenen Block, um anderen zu schaden oder einen eigenen Vorteil zu gewinnen.

Der klassische Fall wäre ein Miner in einem Miningpool, der nach Finden eines nächstgültigen Blockes diesen nicht mit dem Pool teilt, sondern ihn unabhängig veröffentlicht, um die komplette Belohnung für sich zu beanspruchen. [Quelle 11]

4.3 Sicherheitskritische Zusatz-Features

4.3.1 Arbitrary Blockchain Content

Neben der vorgesehenen Nutzung der Blockchain für Transaktionen gibt es Methoden, (ungewollte) Daten in Transaktionen zu injizieren. Hierfür gibt es auch frei verwendbare Tools wie CryptoGraffiti [Quelle 15] oder apertus [Quelle 16].
Auf diese Weise ist es möglich, Daten permanent und öffentlich oder in verschlüsselter Form auf der Blockchain zu speichern.
Dies kann zwar Vorteile mit sich bringen, könnte aber bei illegalen Daten das Speichern und Verwenden der Blockchain in einigen Ländern illegal machen. [Quelle 17]

4.3.2 Smart Contract

Smart Contracts sind digitale selbstausführende Verträge. Die Vertragsbedingungen werden in Code niedergeschrieben und die vordefinierten Aktionen werden infolge von bestimmten Ereignissen ausgelöst.

Neben der vielen Vorteile von Smart Contracts, bieten diese aber auch Sicherheitsrisiken. So hat die Eintrittsinvarianz von Smart Contracts im Juni 2016 den DAO hack möglich gemacht, bei dem einige Millionen US Dollar in Form von Ethereum gestohlen wurden. Dieses Ereignis führte zur Aufteilung der Ethereum Blockchain in Ethereum Classic (mit Dao Hack) und Ethereum (Gabelung von vor dem Hack). [Quelle 18]

4.4 Nutzer/Nutzerschnittstellen

4.4.1 Phishing

Beim Phishing wird die Gutgläubigkeit des Nutzers ausgenutzt, um Daten oder Geld zu erhalten.
Speziell bei Kryptowährungen wird hierbei häufig versucht, an den Seed oder private Keys des Nutzers zu kommen. Ebenso werden häufig via Socialmedia Profile populärer Personen imitiert um große Mengen von Geld im Austausch gegen einen kleinen Vorschuss zu “verschenken”.

4.4.2 Wallet Theft

Wallet Theft bezeichnet den Diebstahl einer digitalen Geldbörse oder deren Inhalt. Häufig werden online Seed-Generatoren verwendet, um Nutzer, die nicht in der Lage sind, ihren eigenen Seed zu erstellen (beispielsweise weil ihre Wallet keinen eingebauten Seed Generator hat), um ihr Geld zu betrügen. Hierbei schreibt einfach der Betreiber des Generators den erzeugten Seed mit und verfügt so über eine Backdoor.
Alternativ wäre es auch möglich, eine eigene Wallet mit Backdoor zu entwickeln oder einen Wallet Anbieter zu hacken, um in dessen Wallets eine Backdoor zu integrieren.
In seltenen Fällen kann es auch zu einem physikalischen Raub des Speichermediums der Wallet kommen.

4.4.3 Cryptojacking

Beim Cryptojacking wird der Rechner eines Nutzers ohne sein Wissen zum Minen von Kryptowährungen verwendet. Einige Webseiten, wie beispielsweise The Pirate Bay [Quelle 19] , oder Anbieter von kostenlosen Programmen lassen Nutzer für sich Kryptowährungen minen.
Neben Cryptojacking als Anbieter einer Webseite oder eines Programmes ist es aber auch möglich, Werbeplattformen auszunutzen, um Werbung auf fremden Seiten zu schalten, die im Hintergrund mittels Mining Kryptowährung generiert. So wurde 2018 die Verwendung eines Mining Skriptes über die Platform Google Ads entdeckt [Quelle 20].

5. Schutzmaßnahmen

5.1 Traditionelle Angriffe

5.1.1 Denial of Service Angriffe

Je dezentraler ein System ist, desto sicherer ist es vor Denial of Service Angriffen. Mehr Nodes bedeutet ein geringes Risiko bei einem DoS Angriff.
Eine eigene Wallet zu haben, schützt vor dem Zugriffsverlust auf das eigene Geld im Fall eines Angriffs auf einen Exchange. Die Existenz mehrere Exchanges reduziert das Risiko, dass man sein Geld während eines DoS Angriffs nicht umtauschen kann.
Miningpools, wie sie bei Bitcoin existieren, stellen ein zusätzliches Sicherheitsrisiko für die gesamte Blockchain dar und sollten möglichst dezentral sein.

Generelle Schutzmechanismen vor Denial of Service Angriffen stellen das Bereitstellen von ausreichend leistungsfähiger und redundanter Hardware mit automatischer Lastverteilung, sowie das Erkennen und Drosseln böswilliger Pakete durch Firewalls und den Internetprovider dar. [Quelle 21]

5.1.2 Hack von Dienstleistern

Ein erster Schutz stellt der Download von Software wie Wallets aus offiziellen Quellen inklusive Überprüfung der Checksummen dar.
Vor Angriffen auf Exchanges können sich Nutzer schützen, indem sie dort kein Geld einlagern und nur vertrauenswürdige Plattformen nutzen, die ihre Kunden vor Verlusten durch Hacks versichern und nur die nötigsten Kundendaten sicher abspeichern.
Vor der falschen Darstellung des Ledger Zustands kann man sich durch die Verwendung eines eigenen Nodes oder durch das Abgleichen der Informationen mehrerer Plattformen absichern.

5.1.3 DNS cache poisoning

Das Netzwerk wird resistenter gegen DNS cache poisoning, wenn mehr Nodes unter vielen verschiedenen Domains, welche von unterschiedlichen Domain Name Servern gehostet werden, registriert sind. Eine einfache Überprüfung der Peers würde in diesem Fall zeigen, ob man sich im korrekten Netzwerk befindet.
Eine Möglichkeit, vor Man-in-the-Middle Angriffen zu schützen, wäre die Verwendung eines verschlüsselten DNS Protokolls wie DNS over TLS oder DNS over HTTPS.

5.1.4 BGP Hijacks

Da es sich hierbei um eine Sicherheitslücke des Border Gateway Protocols handelt, ist es quasi nicht möglich, sich davor zu schützen. Es gibt die Möglichkeit einen BGP Hijack frühzeitig zu erkennen und die Verbindung zu unterbrechen. Ansonsten muss man auf eine Lösung des Problems durch die Fertigstellung von BGPsec (Border Gateway Protocol Security) spekulieren.
[Quelle 22]

5.2 Konsens-Protokolle

5.2.1 51% Angriff

Es gibt bereits einige Algorithmen die 51% Angriffe mildern können. So muss beispielsweise bei DASH eine große Zahl an Nodes aus den “Long Living Masternode Quorums” (LLMQ) einem Block zustimmen, bevor er an die Kette angehängt werden kann. Andere Mechanismen wie der Algorithmus von PirlGuard oder der Vorschlag von Horizen [Quelle 23] setzen eher darauf böswilliges Verhalten zu bestrafen.
[Quelle 24]

5.2.2 Eclipse Angriff

Eclipse Angriffe lassen sich über Zufallsbasiertes autopeering, wie es im Fast Probabilistic Consensus Protokoll der Fall ist, sehr stark erschweren.

5.2.3 Spam

Das übermäßige Versenden von Spam lässt sich durch das Festlegen minimaler Transaktionskosten lösen. Transaktionskosten müssen in diesem Fall aber nicht zwangsweise finanzieller Natur sein. Das Verrichten von Arbeit, beispielsweise durch das Lösen eines kryptografischen Rätsels oder das Verifizieren anderer Transaktionen im Netzwerk können ebenfalls als Kosten verwendet werden.

5.2.4 Mempool flooding

Es gibt mehrere Ansätze, um Mempool flooding zu verhindern. So ist es möglich, einen Mempool zu konstruieren, in dem nur Transaktionen akzeptiert werden, welche mehr als einen dynamisch generierten Grenzwert zahlen.
Ein anderer Ansatz wäre das Speichern des Alters einer Transaktion und das automatische Filtern von Transaktionen die ein gewisses Alter überschreiten.
[Quelle 25]

5.2.5 Block Withholding Angriff

Als Gegenmaßnahme können Mining-Pools einen modifizierten PoW Algorithmus zur Verfügung stellen, bei dem der Miner nicht erfährt, ob sein Block gültig war oder nicht.
Alternativ ist es auch möglich, andere Mining-Pools zu infiltrieren, um zu untersuchen, ob ein Miner ihres Pools Ergebnisse teilt, um mehrfach vergütet zu werden. Falls ein solcher Miner gefunden wird, so wird er nicht mehr vergütet.
[Quelle 26]

5.3 Sicherheitskritische Zusatz-Features

5.3.1 Arbitrary Blockchain Content

Es wird wohl immer möglich sein, Daten über Transaktionen zu kodieren, folglich wird ein permanenter Speicher von Transaktionen auch immer das Risiko mit sich führen, illegale Daten zu speichern. Da es nicht möglich ist, das Speichern illegaler Daten zu verhindern, wäre der nächste Schritt, das Anzeigen solcher zu unterbinden.
Es könnte ein System implementiert werden, dass Nodes über illegale Inhalte in Kenntnis setzen kann und diese infolgedessen ausgeblendet werden können.

5.3.2 Smart Contract

Um die Sicherheit von Smart Contracts zu gewährleisten, kann eine Reihe von Test und Analyse Umgebungen, wie in ZeppelinOS und Oyente enthalten, verwendet werden. Ebenso können Sicherheitsaudits abgehalten werden, um mögliche Fehler frühzeitig zu finden und zu beheben.
[Quelle 18]

5.4 Nutzer/Nutzerschnittstellen

5.4.1 Phishing

Den besten Schutz vor Phishing stellen Schulung von Nutzern und das kritische Hinterfragen von Angeboten dar.

5.4.2 Wallet Theft

Die einfachste Möglichkeit eines Herstellers, seine Nutzer zu schützen, wäre eine proprietäre eigens entwickelte Wallet mit Sicherheitsaudits.
Aus Nutzersicht muss man einen vertrauenswürdigen Anbieter finden und Aspekte wie Sicherheitsaudits und Checksummen beim Download beachten.

5.4.3 Cryptojacking

Gegen Cryptojacking im Web kann man sich mithilfe von Skriptblockern, wie uMatrix [Quelle 27], verteidigen. Bei Programmen mit eingebautem Miner, ist es schwieriger sich zu schützen. Generell ist es hilfreich, nur quelloffene Programme aus vertrauenswürdigen Quellen, im besten Fall mit Sicherheitsaudits, zu verwenden. Der beste, aber wahrscheinlich auch unkomfortabelste Schutz, dürfte aber die Verwendung einer Firewall mit Whitelist sein.

6. Abschließende Bemerkungen

6.1 Ausblick

Eine generelle Verbesserung der IT-Sicherheit würde auch DLTs zugutekommen. So bräuchten Internetprovider bessere Systeme um DDoS Attacken zu erkennen und automatisch zu unterbinden. Ebenso ist die Weiterentwicklung von BGPsec für das Internet im Allgemeinen wichtig.
Nutzer müssen auf Gefahren wie Phishing oder die Relevanz der Geheimhaltung von Seeds und Private Keys besser geschult werden.
Die stetige Weiterentwicklung verschiedener Schutzmaßnahmen von Konsens-Protokollen lässt den Schluss zu, dass viele der heute bekannten Sicherheitsrisiken in zukünftigen Ledger-Systemen minimiert sein werden oder sogar gänzlich entfallen.

6.2 Fazit

Während Distributed Ledger Technologien zwar über einige neuartige Angriffsvektoren verfügen, gibt es trotzdem Überschneidungen mit bereits bekannten Risikoaspekten wie Denial of Service oder DNS-Spoofing.
Gemäß ihrer Natur sind DLT-Systeme hierfür aber weniger anfällig, als herkömmliche IT-Systeme.
Andere Sicherheitsrisiken wie BGP Hijacks stellen hingegen grundlegende Probleme dar, von denen Internet-basierte Technologien im Allgemeinen betroffen sind.
Nutzer werden, auch unabhängig von DLTs, im Internet häufig auf Phishing-Versuche oder Malware stoßen. Unternehmen, die DLT-Technologie nutzen, oder darauf angewiesen sind, sollten Sorge tragen, durch gezielte Schulung von Nutzern und Mitarbeitern eine entsprechende Sensibilisierung zu schaffen.
Arbitrary Blockchain Content ist äquivalent zur Problematik rechtswidriger Daten im Internet. Auch hier können Daten beispielsweise von staatlichen Behörden nicht einfach gelöscht werden, wenn sie nicht unmittelbar der eigenen Kontrolle unterliegen. Die Möglichkeit, solche Daten auszublenden, bzw. den Zugriff darauf zu erschweren, scheint die Technologie aber zu legitimieren.
Smart Contracts bieten ein großes Potential, es ist allerdings wichtig zu erkennen, dass, ähnlich wie bei Rechtsverträgen, Lücken ausgenutzt werden können und diese daher genau analysiert werden müssen.
Die größte Unterschiede im Vergleich zu anderen IT-Systemen finden sich in Konsens-Protokoll-spezifischen Angriffsvektoren.
Teilweise wird hier bereits an der Implementierung von Lösungen seitens der Protokollentwickler gearbeitet, teilweise sind Lösungsansätze aber bisher nur theoretischer Natur.

Angriffsvektoren, wie 51% Angriffe, die eine unmittelbare Konsequenz der Funktionsweise des zugrundeliegenden Systems darstellen, sind nur schwer zu mitigieren, da ein grundlegendes Redesign des Systems notwendig wäre. Aufgrund der Vielfalt an DLT-Implementierungen sind aber bei weitem nicht alle Protokolle von derartigen Problemen befallen.

6.3 Quellen

1. David Mazieres and Dennis Shasha: Building secure file systems out of Byzantine storage*, https://cs.nyu.edu/cs/faculty/shasha/papers/mazpodc.pdf, Stand 30.07.2020
2. VW Patent, https://worldwide.espacenet.com/patent/search/family/068424718/publication/EP3654222A1?q=pn%3DEP3654222A1, Stand 30.07.2020
3. gq: Blockchain gegen China Zensur, https://www.gq-magazin.de/auto-technik/article/mit-der-blockchain-gegen-chinas-internet-zensur, Stand 30.07.2020
4. Bitcoin Wiki: Hashcash, https://en.bitcoin.it/wiki/Hashcash, Stand 30.07.2020
5. Digiconomist: Bitcoin Energy Consumption, https://digiconomist.net/bitcoin-energy-consumption/, Stand 30.07.2020
6. Wikipedia: Proof of Stake, https://en.wikipedia.org/wiki/Proof_of_stake, Stand 30.07.2020
7. O. Vashchuk, R. Shuwar: Pros and Cons of Consensus Algorithm Proof of Stake. Difference in the Network Safety in Proof of Work and Proof of Stake, https://pdfs.semanticscholar.org/cd88/4fe94f0981a6a1bd56021d84d0e3565f5136.pdf , Stand 30.07.2020
8. Iota Coordicide Modul 4.1.1, https://coordicide.iota.org/module4.1.1, Stand 30.07.2020
9. Serguei Popov, William J Buchanan: FPC-BI: Fast Probabilistic Consensus within Byzantine Infrastructures, https://arxiv.org/pdf/1905.10895.pdf, Stand 30.07.2020
10. Wikipedia: Byzantine Fault, https://en.wikipedia.org/wiki/Byzantine_fault, Stand 30.07.2020
11. Muhammad Saad, Jeffrey Spaulding, Laurent Njilla, Charles Kamhoua, Sachin Shetty, DaeHun Nyang, and Aziz Mohaisen: Exploring the Attack Surface of Blockchain: A Systematic Overview, https://arxiv.org/pdf/1904.03487.pdf, Stand 30.07.2020
12. The Verge (Russell Brandom): Hackers emptied Ethereum wallets by breaking the basic infrastructure of the internet, https://www.theverge.com/2018/4/24/17275982/myetherwallet-hack-bgp-dns-hijacking-stolen-ethereum, Stand 30.07.2020
13. Wired (Andy Greenberg): Hacker Redirects Traffic From 19 Internet Providers to Steal Bitcoin, https://www.wired.com/2014/08/isp-bitcoin-theft/, Stand 30.07.2020
14. Maria Apostolaki, Aviv Zohar, Laurent Vanbever: Hijacking Bitcoin: Routing Attacks on Cryptocurrencies,

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7958588, Stand 30.07.2020
15. Arbitrary Blockchain Content lesen und hochladen über cryptograffiti, https://cryptograffiti.info/, Stand 30.07.2020
16. Arbitrary Blockchain Content lesen und hochladen über apertus, http://apertus.io/, Stand 30.07.2020
17. Roman Matzutt, Jens Hiller, Martin Henze, Jan Henrik Ziegeldorf, Dirk Müllmann, Oliver Hohlfeld, and Klaus Wehrle: A Quantitative Analysis of the Impact of Arbitrary Blockchain Content on Bitcoin, https://fc18.ifca.ai/preproceedings/6.pdf, Stand 30.07.2020
18. Ardit Dika and Mariusz Nowostawski: Security Vulnerabilities in Ethereum Smart Contracts, https://www.researchgate.net/publication/333590995_Security_Vulnerabilities_in_Ethereum_Smart_Contracts, Stand 30.07.2020
19. wccftech (Rafia Shaikh): The Pirate Bay Is Cryptojacking Its Visitors' Computers to Mine for Monero, https://wccftech.com/the-pirate-bay-cryptojacking-mine-monero/#:~:text=The%20Pirate%20Bay%20Is%20Cryptojacking%20Its%20Visitors'%20Computers%20to%20Mine%20for%20Monero,-By%20Rafia%20Shaikh&text=Coinhive%2C%20a%20technology%20that%20has,spotted%20on%20the%20TPB%20again., Stand 30.07.2020
20. Forbes (Lee Mathews): Hackers Abuse Google Ad Network to Spread Malware That Mines Cryptocurrency, https://www.forbes.com/sites/leemathews/2018/01/26/hackers-abuse-google-ad-network-to-spread-malware-that-mines-cryptocurrency/, Stand 30.07.2020
21. Axel Hagedorn, Claudia Eckert, Thomas Buntrock: Distributed Denial of Service Angriffswerkzeuge und Abwehrmöglichkeiten, https://www.computec.ch/archiv/dokumente/denial_of_service/distributed_denial_of_service-angriffswerkzeuge_und_abwehrmoeglichkeiten.pdf, Stand 30.07.2020
22. Computerweekly (Michael Cobb): Wie BGP-Hijacking erkannt und verhindert werden kann, https://www.computerweekly.com/de/antwort/Wie-BGP-Hijacking-erkannt-und-verhindert-werden-kann, Stand 30.07.2020
23. Horizon (Alberto Garoffolo, Pier Stabilini, Robert Viglione, Uri Stav): Proposal to Modify Satoshi Consensus to Enhance Protection Against 51% Attacks, https://www.horizen.global/assets/files/A-Penalty-System-for-Delayed-Block-Submission-by-Horizen.pdf, Stand 30.07.2020
24. Sarwar Sayeed and Hector Marco-Gisbert: Assessing Blockchain Consensus and Security Mechanisms against the 51% Attack, https://www.researchgate.net/publication/332737156_Assessing_Blockchain_Consensus_and_Security_Mechanisms_against_the_51_Attack, Stand 30.07.2020
25. Muhammad Saad, Joongheon Kim, DaeHun Nyang, David Mohaisen: Contra-*: Mechanisms for Countering Spam Attacks on Blockchain's Memory Pools, https://arxiv.org/pdf/2005.04842.pdf, Stand 30.07.2020
26. Suhyeon Lee and Seungjoo Kim: Countering Block Withholding Attack Efficiently, https://eprint.iacr.org/2018/1211.pdf, Stand 30.07.2020
27. uMatrix von gorhill, https://github.com/gorhill/uMatrix, Stand 30.07.2020

6.4 Glossar

Die folgenden Begriffserklärungen beziehen sich auf den Kontext dieses Dokumentes:

51% Angriff
Angriffsszenario, typischerweise bei PoW Blockchains, bei dem 51% der Mining-Power verwendet werden, um eine Gabelung der Blockchain zu verursachen

Anwendungsspezifische Integrierte Schaltung (ASIC)
Speziell entwickelte Hardware, um einen bestimmten Algorithmus möglichst effizient durchzuführen

BGP Hijacking
Routing Tabellen des Border Gateway Protocols manipulieren, um Pakete umzuleiten.

Bitcoin
Eine 2008 erfundene Kryptowährung basierend auf einer PoW Blockchain.

Block
Ein Element einer Blockchain. Bei der Datenbank Analogie wäre die Blockchain eine Datenbank und ein Block ein Datensatz.

Block Withholding Angriff
Das Zurückhalten der Bekanntgabe eines gefundenen Blocks zum eigenen Vorteil

Blockchain
Eine Implementierung einer dezentralen Datenbank

Brute-Force
Das Ausprobieren aller theoretisch möglichen Lösungen für ein Problem

Byzantine Fault
Ein Fehlerzustand in einem verteilten Computersystem, bei dem unvollständige Informationen darüber vorhanden sind, ob einzelne Komponenten fehlerhaft

Byzantine Fault Tolerance
Die Eigenschaft eines Systems, einen Konsens in einem verteilten Netzwerk zu finden, obwohl Nodes mit fehlerhaften Informationen oder gar nicht antworten, weil sie beispielsweise durch böswillige Dritte übernommen wurden

Cellular Consensus
Ein Konsens-Algorithmus mit Abstimmungsprozess, modelliert als zellularer Automat

Checksumme
Ein Wert zur Überprüfung der Integrität von Daten via Bilden eines Hashs

Cryptojacking
Verfahren, um Rechenkraft des Zielsystems zum Minen von Kryptowährungen auszunutzen

Kryptowährung
Digitales Zahlungssystem auf Basis von Kryptografie.

DASH
Eine Kryptowährung basierend auf einer PoW Blockchain

delegated Proof of Stake
Eine Abwandlung des Proof of Stake Algorithmus mit Delegierten anstelle einer Zufallsauswahl basierend auf dem Stake

Denial of Service
Das verursachen einer Nichtverfügbarkeit eines Systems, meist durch eine Überlastung des Datennetzes

Difficulty
Der Schweregrad beim Minen. Er wird automatisch über die Blockrate festgelegt und wird über die Anzahl der führenden Nullen beim Hashwert des Folgeblocks umgesetzt.

Directed Acyclic Graph/Gerichteter azyklischer Graph
Modell aus der Graphentheorie, bei dem die Kanten gerichtet sind und keinen Zyklus bilden dürfen. Wird bei DLTs zur Steigerung der Effizient im Vergleich zur Blockchain verwendet.

Distributed Denial of Service
Denial of Service Angriff ausgehend von mehreren angreifenden Geräten gleichzeitig

Distributed Ledger Technologie:
Ein verteiltes System zum Speichern von Transaktionen

DNS cache poisoning
Manipulation des Cache eines DNS-Resolvers

Double Spend
Dasselbe digitale Token einer Währung wird mehr als einmal ausgegeben, um eine Zahlung zu tätigen

Eclipse Angriff
Angriff, bei dem dem eine Überzahl böswilliger Nodes in einem Cluster die nicht-kompromittierten Nodes vom restlichen Netz isolieren

Ethereum
Eine Plattform für dezentrale Programme und Smart Contracts, basierend auf einer PoW Blockchain mit eigener Kryptowährung

Exchange
Plattform zum Austausch digitaler Kryptowährungen und staatlich anerkannter Zahlungsmittel

Fast Probabilistic Consensus
Eine Abwandlung des Cellular Consensus Protokolls, bei dem die Abstimmung nicht mehr asynchron zwischen den einzelnen Nachbarn abläuft, sondern in Runden unterteilt ist

Federated DLT
DLT, die von einem Konsortium von juristischen Personen entwickelt und genutzt wird

Gabelung
Spaltung der Blockchain. Blöcke vor der Gabelung bleiben identisch, Blöcke dahinter variieren

Hashcash
Ein in den 1990ern entwickeltes Proof of Work System zu Verhinderung von DoS Angriffen

Hebel
Ein Hebel bezeichnet das Handeln mit geliehenem Kapital. Kursschwankungen wirken sich entsprechend des Hebelmultiplikators vielfach auf die Rücklagen aus

Hybrid DLT
Eine Mischung verschiedenen Ledger Variationen wie private und public DLT

Ledger
Eine Art Datenbank oder Kassenbuch. Das eigentliche Objekt einer Distributed Ledger Technologie

Man-in-the-Middle
Das unbemerkte Abhören oder Manipulieren einer Datenübertragung zwischen zwei Endpunkten

Mempool flooding
Das Versenden einer großen Zahl von Transaktionen, um Transaktionskosten künstlich zu erhöhen

Mining
Das Suchen von Folge-Blöcken unter Aufwendung von Rechenleistung

Mining-Pool
Der Zusammenschluss mehrerer Miner zu einer Gruppe, die Arbeit, sowie Belohnungen untereinander aufteilt

Node
Ein Gerät, das als Knotenpunkt im Netzwerk dient und Transaktionen speichert

nonce
Eine beliebige zufällige oder pseudo-zufällige Zahl, die einmalig in einem kryptografischen Vorgang genutzt werden kann

Permissioned DLT
DLT mit Rechteverwaltung und Nutzergruppen

Permissionless DLT
DLT, bei der jeder Nutzer gleiche Rechte hat

Practical Byzantine Fault Tolerance
Ein Konsens-Algorithmus auf dem viele Algorithmen aus dem Bereich der DLT basieren

Private DLT
DLT, die nicht öffentlich zugänglich ist

Proof of Stake
Ein Konsens-Algorithmus, der Personen auf Basis ihres Stakes auswählt um den nächsten Block zu erzeugen

Proof of Work
EIn Konsens-Algorithmus, der unter dem Einsatz von Arbeit beispielsweise durch Miner zur Konsensfindung kommt.

Public DLT
öffentlich nutzbare und zugängliche DLT

SHA256
Eine weit verbreitete Hashfunktion zur Ausgabe von 256 bit langen Hashes

Short-Position
Das Spekulieren auf einen fallenden Kurs

Sidechain
Der Pfad einer Blockchain Gabelung, der nicht als Hauptkette angesehen wird

Smart Contracts
automatisch ausführende, programmierbare, digitale Verträge

Spam
Das übermäßige Versenden von Transaktionen um das Netzwerk oder einzelne Nodes zu verlangsamen

Stake
Eine Gewichtung der Anteile die eine Person im DLT hat. Meist auf Grundlage der Token-Anzahl und weiterer Aspekte wie dem Alter der Tokens

Transaktion
Eine Übertragung digitaler Güter wie Tokens, Informationen oder Metainformationen

Unique Identifier
Ein eindeutig zuordenbarer Bezeichner

Wallet
Eine Art digitale Geldbörse, die meist auch zum Empfangen und Versenden digitaler Währungen genutzt werden kann

6.5 Abkürzungen und Akronyme

DLT	Distributed Ledger Technology
DAG	Directed Acyclic Graph
PoW	Proof of Work
PoS	Proof of Stake
dPoS	delegated Proof of Stake
FPC	Fast Probabilistic Consensus
BF	Byzantine Fault
BFT	Byzantine Fault Tolerance
DoS	Denial of Service
DDoS	Distributed Denial of Service
DNS	Domain Name System
BGP	Border Gateway Protocol
Mempool	Memory Pool
TLS	Transport Layer Security
HTTPS	Hypertext Transfer Protocol Secure