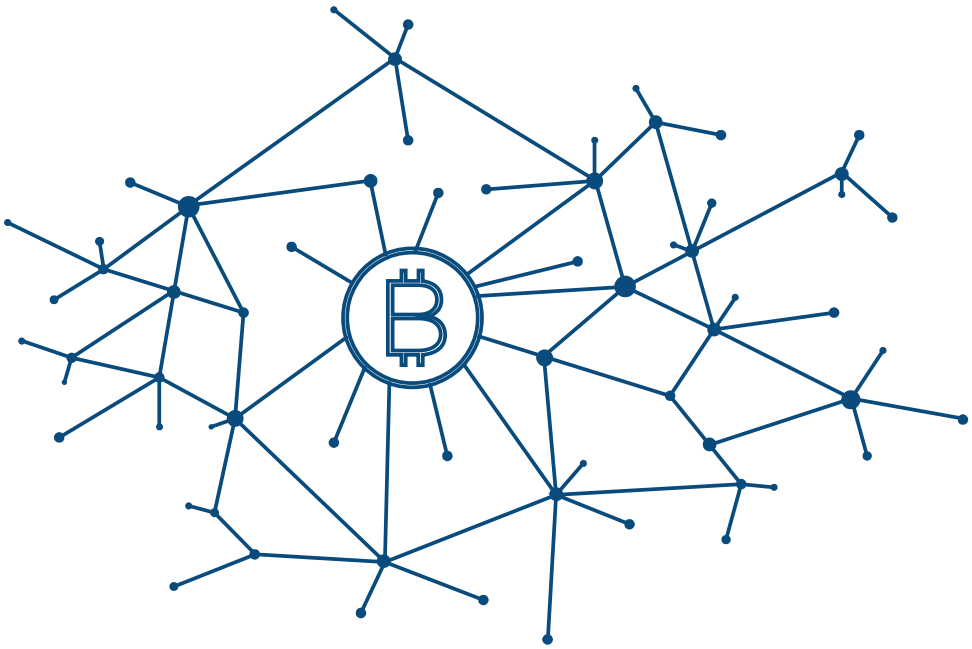
Die Anwendung der Kryptographie in der Blockchain am Beispiel von Bitcoin

Albert-Einstein-Schule
TGJ ½
Mathematik, Frau Heckmann
Daniel Vera Gilliard
16.04.18



Inhaltsverzeichnis

1 Einleitung

Die Kryptographie spielt eine sehr große Rolle in unserer Gesellschaft. Sie übernimmt sehr viele Aufgaben, und ohne sie wäre unser heutiges Leben nicht denkbar. Einsatzgebiete sind unter anderem das Internet, die Bankgeschäfte oder der Gebrauch von sozialen Medien.
In dieser GFS werde ich jedoch auf die Blockchain als Einsatzgebiet der Kryptographie näher eingehen. Die Blockchain ist eine Art, Systeme dezentral aufzubauen. Bekannt wurde die Blockchain-Technologie in den letzten Jahren durch den Einsatz im Bereich der Kryptowährungen. Diese benutzen die Blockchain-Technologie um eine dezentrale Infrastruktur aufzubauen, die nicht von einer zentralen Entität wie den Banken abhängt.

2 Grundlagen der Kryptographie

[1]Das Ziel von Kryptographie ist es, Informationen zuverlässig und vertraulich über sichere Kanäle zu übertragen. Dabei sollte eine Nachricht durch Dritte nicht abgefangen, mitgelesen oder unbemerkt verändert werden können. Um das zu ermöglichen, versucht man in der Kryptographie Methoden zu entwickeln, um Informationen so effizient wie möglich zu verschlüsseln, sowie zu entschlüsseln. Beim Codieren spricht man hierbei vom Verschlüsseln, beim Decodieren vom Entschlüsseln.
Um Informationen zu codieren, sowie zu decodieren, benutzt man Schlüssel. Dabei unterscheidet man zwei kryptographische Verfahren. Bei der ersten Methode sind die Schlüssel zum Codieren und Decodieren identisch. Man spricht von einem symmetrischen Verfahren.
Wenn die Schlüssel nicht identisch sind, spricht man von einem unsymmetrischen Verfahren.

2.1 Forderungen auf kryptographische Verfahren

[2]Bei der Betrachtung von kryptographischen Verfahren sollte man verschiedene Forderungen beachten. Diese sorgen dafür, dass ein gewisser Grad an Sicherheit gewährleistet ist:

1 Vgl. Ernst Hartmut, Schmidt Jochen, Beneken Gerd: Grundkurs Informatik: Grundlagen und Konzepte für die erfolgreiche IT-Praxis - eine umfassende, praxisorientierte Einführung, - 5., vollst. überarb. Aufl., 20. März 2015, S. 137f.

2 Vgl. ebd., S138

1. Geheimhaltung und Vertraulichkeit
    - Es sollte für einen Dritten Unbefugten nicht möglich sein, die Nachrichten zu entschlüsseln.
2. Integration
    - Falls die Nachricht doch in unbefugten Besitz kommen sollte, darf der Unbefugte die Nachricht nicht entschlüsseln oder verändern können.
3. Authentizität
    - Es sollte eine Gewissheit zwischen den Kommunikationspartnern herrschen.
4. Schlüssel-Management
    - Die Art, wie Schlüssel erzeugt, verwahrt und weitergegeben werden, sollte geheim sein.

2.2 Symmetrische Kryptographie

[3]Die Symmetrische Verschlüsselung basiert darauf, dass es nur einen geheimen Schlüssel zum Verschlüsseln und zum Entschlüsseln gibt.

Diesen geheimen Schlüssel werde ich im folgenden als k bezeichnen.

Dabei kann man mithilfe des Schlüssels k Nachrichten folgendermaßen verschlüsseln: $y=e(x,k)$. e steht für Encription (Englisch für Verschlüsseln) und x für die zu verschlüsselnde Nachricht.

Um die geheime Nachricht y zu entschlüsseln, benutzt man ebenfalls den Schlüssel k : $x=d(y,k)$. Die entschlüsselte Nachricht x ergibt sich aus der Funktion d (Decription steht im Englischen für Entschlüsseln), der verschlüsselten Nachricht y und dem Schlüssel k .

Die größte Schwäche der symmetrischen Kryptographie ist das Benutzen des gleichen Schlüssels für beide Kommunikationspartner. Der Schlüssel muss irgendwie weitergegeben werden. Dabei besteht die Gefahr, dass ein Dritter diesen Schlüssel abfangen könnte und somit Nachrichten mitlesen könnte.

Eine Möglichkeit diese Schwachstelle zu verbessern sind hybride Verfahren. Diese benutzen ein symmetrisches Verfahren um Nachrichten zu verschlüsseln, aber ein asymmetrisches Verfahren um Schlüssel sicher zu übertragen. Da symmetrische Verfahren keine wichtige

3 Vgl. ebd., S139

Rolle im Blockchain-Bereich spielen, wird im Folgenden nicht mehr auf symmetrische Verfahren eingegangen.

2.3 Asymmetrische Kryptographie

[4]Asymmetrische Verfahren benutzen öffentliche und private Schlüssel zum Verschlüsseln und Entschlüsseln der Nachrichten.

Die Nachricht y wird mithilfe der Funktion e folgendermaßen verschlüsselt: $y=e(x,ke)$. Hierbei ist x die zu verschlüsselnde Nachricht und ke der Schlüssel. Der Unterschied zum symmetrischen System ist, dass der Schlüssel ke aus einem öffentlichen Schlüsselverzeichnis genommen wird.

Zum Entschlüsseln wird hierbei ein privater Schlüssel kd benutzt: $x=d(y,kd)$. Die Nachricht x wird aus der Funktion d , der verschlüsselten Nachricht y und dem nicht öffentlich verfügbaren privaten Schlüssel kd entschlüsselt.

Das Hauptproblem von asymmetrischen Verfahren ist die Übertragung des privaten Schlüssels. Dieses kann gelöst werden, indem man hybride Verfahren benutzt.

Ein Beispiel ist der Einsatz des Diffie-Hellman Algorithmus zum Schlüsselaustausch und ein weiteres asymmetrisches Verfahren zum Austausch der Nachrichten.

Mehr zum Diffie-Hellman Algorithmus im Abschnitt 2.3.1.

2.3.1 Das Diffie-Hellman Protokoll

Das Diffie-Hellman Protokoll ist ein Algorithmus, welcher als einer der ersten das Erstellen eines gemeinsamen Schlüssels durch eine unsichere Verbindung ermöglichte. Dabei benutzen beide Verbindungspartner den gleichen Schlüssel.

2.3.1.1 Vorgehen[5]

Zwei Personen (In diesem Beispiel Alice und Bob), vereinbaren eine Primzahl p und eine natürliche Zahl g . Dabei muss $g<p$ gelten.

- Verschlüsselung:
  - Alice:

4 Vgl. ebd., S139

5 Vgl. http://ddi.uni-wuppertal.de/material/spioncamp/dl/austausch-diffie-hellman-station.pdf“, (09.03.18)

1. Alice wählt eine Zahl a , wobei $a<p$ ist.
2. Alice rechnet: $g^a \bmod p = A$.
3. Alice sendet A an Bob.

- Bob:

1. Bob wählt Zahl b , wobei $b<p$ ist.
2. Bob rechnet: $g^b \bmod p = B$.
3. Bob sendet: B an Alice.

- Erstellung des geheimen Schlüssels:
  - Alice

1. Alice rechnet $B^a \bmod p = K$, wobei K der gemeinsame geheime Schlüssel ist.

- Bob

1. Bob rechnet $A^b \bmod p = K$, wobei K der gemeinsame geheime Schlüssel ist.

2.3.1.2 Angriff auf das Diffie-Hellman Protokoll

Die Schwäche des Diffie-Hellman Protokoll liegt im privaten Schlüssel, der kleiner als die Primzahl sein muss. Damit eine dritte Person den geheimen Schlüssel berechnen könnte, bräuchte er die geheime Zahl von Alice oder Bob. Da diese Zahl kleiner als die Primzahl ist, und die Person die Primzahl kennt, muss er nur noch so lange ausprobieren bis er zufällig auf die richtige Zahl stößt.

2.3.1.3 Anwendung des Diffie-Hellman Protokolls

[6]Das Diffie-Hellman Protokoll ist die Grundlage für weitere Verschlüsselungstechniken. Die Anwendung des Diffie-Hellman Protokolls in Kombination mit anderen Austauschprotokollen sieht folgendermaßen aus:

1. Man erzeugt mithilfe des Diffe-Hellman Protokolls, wie oben gezeigt, einen geheimen Schlüssel.
2. Man wählt ein symmetrisches oder asymmetrisches Verschlüsselungsverfahren aus, um die Nachricht zu verschlüsseln.

6 Vgl. http://ddi.cs.uni-potsdam.de/Lehre/e-commerce/elBez2-5/page07.html (01.12.17)

Wie oben schon beschrieben nennt man Verfahren, die symmetrische und asymmetrische Kryptographie kombinieren, auch hybride Verschlüsselung. Der Vorteil an der hybriden Verschlüsselung gegenüber vollkommen asymmetrischen Verfahren liegt im Rechenaufwand. Dieser ist bei hybriden Verschlüsselungen geringer.

2.3.2 Das RSA Protokoll

Das Diffie-Hellman Protokoll wird in der Kryptographie hauptsächlich zum Schlüsselaustausch genutzt. Das RSA Protokoll kann man im Gegensatz dazu neben dem Schlüsselaustausch auch zum Codieren und Austauschen von Information, sowie zum Signieren von Nachrichten verwenden.

2.3.2.1 Vorgehen

Das RSA Protokoll basiert auf zwei Schritten. Zum einen auf das Erstellen der privaten und öffentlichen Schlüssel, sowie der Verschlüsselung, Versendung und Entschlüsselung von Nachrichten. Dabei wird eine Nachricht m mit dem öffentlichen Schlüssel des Empfängers verschlüsselt. Der Empfänger bekommt den Geheimtext c und entschlüsselt diesen mit dem privaten Schlüssel d .

Die Grundlage bilden hier Einwegfunktionen, bei denen man einfach einen Wert berechnen kann, aber nur schwer Zurückrechnen kann. Die Einwegfunktion ist dabei die Potenzfunktion zum Verschlüsseln der Nachricht.

Bei der Potenzfunktion $c = m^e$ kann man leicht c berechnen, wenn man e und m kennt. Jedoch ist es bei großen Zahlen sehr schwer bis unmöglich, aus c und m e zu berechnen. Möglich wird dies nur wenn man den privaten Schlüssel d hat, mit dem man die Nachricht c entschlüsseln kann.

Neben der Potenzfunktion ist bei der Erstellung des privaten Schlüssels eine Primzahl-Multiplikation als Einwegfunktion zur Sicherheit des Verfahrens vorhanden. Dabei besteht in der Einwegfunktion eine Falltürfunktion. Diese ermöglicht es dem Empfänger der Nachricht, diese zu entschlüsseln.

2.3.2.2 Schlüsselerzeugungsphase[7]

1. Als erstes wählt man zwei Primzahlen p und q .
2. Die zuvor gewählten Primzahlen werden zu einer Zahl $n=p*q$ multipliziert.
3. Mit dem Satz von Euler berechnet man dann $phi(n)$. Dies ist wichtig, um bei der Entschlüsselung die Wurzel aus einem Modulo zu berechnen. $phi(n)$ lässt sich mit $phi(n)=(p-1)*(q-1)$ berechnen.
4. Man bestimmt eine Primzahl e , wobei e mit $ggT(e,phi)=1$.
5. Der öffentliche Schlüssel bestimmt sich dabei aus den Zahlen n und e .
6. Den geheimen Schlüssel kann man berechnen, indem man die Inversen zu e berechnet: Dies kann man mit dem erweiterten euklidischen Algorithmus erreichen.

2.3.2.3 Verschlüsselungs- und Entschlüsselungsphase[8]

1. Eine zweite Person kann mit Kenntnis des öffentlichen Schlüssels e und n eine Nachricht m folgendermaßen verschlüsseln: $c=m^e \bmod n$. Dabei ist c dann der Geheimtext, der verschickt wird. Interessant ist dabei die Modulo-Exponentiation, die der Verschlüsselung die Sicherheit gibt.
2. Der Empfänger kann mit dem privaten Schlüssel d und $c^d \bmod n$ die verschlüsselte Nachricht entschlüsseln. Da eine dritte Person die Faktorisierung von n nicht kennt, ist es nahezu unmöglich, d zu berechnen.

2.3.3 Digitale Signaturen

[9]Eine digitale Unterschrift ist eine Methode, um die Authentizität von einem Kommunikationspartner zu versichern. Man kann sie in der vereinfachten Form mit einer physischen Unterschrift vergleichen. Eine digitale Unterschrift, wie sie für die Kryptographie in der Blockhain benutzt wird, muss drei Eigenschaften haben:

1. Zum einen sollte es ein Nachweis für die tatsächliche Signatur sein.
2. Sie sollte auch fälschungssicher sein.

---

7 Vgl. Schmeh Klaus: Kryptografie. Verfahren, Protokolle, Infrastrukturen, 6. aktualisierte Auflage, April 2016, S. 205 f.

8 Vgl. Ebd, S. 205 f.

9 Vgl. https://blockgeeks.com/guides/cryptocurrencies-cryptography/ (09.03.18)

3. Sie sollte die Fähigkeit haben, dass man die Signatur nicht wieder zurücknehmen kann, oder ein Dritter behaupten könnte, er hätte diese Signatur gemacht.

Eine physische Unterschrift als Signatur stößt spätestens bei dem zweiten Punkt auf Probleme. In der Blockchain werden digitale Signaturen mithilfe von sogenannten “Schlüsseln” erstellt. In Abschnitt 3.2.2 wird genauer auf die Signatur in der Bitcoin Blockchain eingegangen.
Zusammengefasst hat eine Signatur die Aufgabe, die Datenintegrität zu sichern, und stellt somit sicher, dass eine Nachricht nicht im Nachhinein verändert wurde.

3 Anwendung der Kryptographie in der Blockchain anhand der Bitcoin Blockchain

Bei einer Blockchain geht es im Grunde um ein dezentralisiertes System, welches zum Beispiel als dezentralisierte Datenbank fungieren kann. Dabei besitzt die Blockchain zwei grundlegende kryptographische Protokolle zum Schlüsselaustausch und zur Authentifikation.

Diese sind wichtig damit Person A Person B überzeugen kann, ohne die eigentliche Information preiszugeben. Eine zweite Person über die Richtigkeit einer Information zu überzeugen, ohne die eigentliche Information preiszugeben nennt man auch „Zero-Knowledge-Beweise“.
Da es sehr viele Anwendungen der Blockchain gibt, wird im folgenden die Bitcoin Blockchain als Anwendungsbeispiel genutzt. Wenn also im Folgenden der allgemeine Begriff Blockchain verwendet wird, ist hiermit die Bitcoin Blockchain im Speziellen gemeint.

3.1 Die Blockchain

Die Blockchain ist einfach gesagt ein Bestandsbuch, indem jede Aktivität aufgezeichnet wird, oder anders ausgedrückt: „eine verteilte Datenbank, die aus einer Kette von Datenblöcken besteht.“.[10]

10 Vgl. Hajo Schulz: Das macht Blockchain. Die Technik hinter Bitcoin & Co., in: c‘t Magazin für Computertechnik, 28 August 2017, Nr 23, S 103

‚Die Blockchain kann dabei die Aufgabe vieler zentralisierter Institutionen wie zum Beispiel der Banken übernehmen und sie mithilfe dezentralisierte Eigenschaften gerechter machen. Einer der wichtigsten Vorteile der Blockchain-Technologie ist die Sicherheit gegen nachträgliche Änderungen. Da es nahezu unmöglich ist, Einträge in der Blockchain zu ändern oder löschen, ist sie sehr gut für den Einsatz in der Bankenwelt geeignet.

Eine Blockchain hat hauptsächlich 4 grundlegende Eigenschaften:

1. Sie besitzt ein wiederherstellbares Bestandsbuch aller Einträge.
- Man kann die gesamte Historie nachverfolgen.
- Die Historie ist unveränderbar.
2. Sie basiert auf Kryptographische Verfahren.
- Die Authentizität der Transaktionen ist gewährleistet.
- Die Sicherheit der Transaktionen ist gewährleistet.
- Die Identität der Personen ist sicher.
3. Die Zukunft der Blockchain wird durch die Mehrheit der Mitglieder bestimmt.
- Sie basiert auf ein dezentralisiertes Protokoll.
- Die Aktivitäten werden von den Mitgliedern validiert und erst dann auf die Blockchain mit aufgenommen.
4. Es gibt eine Anwendung, mit der man auf die Blockchain zugreifen kann.
- Sie besitzt, wie bei Bitcoin der Fall, eine virtuelle Währung mit der man die Blockchain aktiv nutzen kann.

Durch diese Eigenschaften kann man das Vertrauen zentraler Entitäten verringern und durch Technologie sowie kryptographische Verfahren ersetzen.

3.1.1 Die Bitcoin Blockchain

Die Bitcoin Blockchain ist ein Ende zu Ende Protokoll (auch „Peer-to-Peer“ genannt) mit dem finanzielle Transaktionen getätigt werden können. Dabei ist das System dezentralisiert aufgebaut und wird durch kryptographische Regeln organisiert. Im Folgenden werden die Grundlagen der Kryptographie hinter der Bitcoin Blockchain thematisiert.

3.2 Kryptographische Methoden in der Bitcoin Blockchain[11]

Die Bitcoin Blockchain besteht aus verschiedenen Kryptographischen Methoden. Als asymmetrisches Verschlüsselungsverfahren, nutzt es die Elliptic-Curve Kryptographie. Dabei ist jeder Bitcoin mit seinem öffentlichen Elliptic-Curve-Schlüssel verbunden. Wenn jemand Bitcoins an eine zweite Person schickt, kommt es zu einer neuen Überweisung. Hierbei werden die Bitcoins mit dem öffentlichen Schlüssel des Empfängers verbunden und mit dem privaten Schlüssel des Versenders signiert. Wenn die Überweisung getätigt ist, wird diese Information an das Bitcoin Netzwerk gesendet. Dies lässt jeden wissen, dass der neue Besitzer der Bitcoins jener mit dem neuen Schlüssel ist und verifiziert die Transaktion. Erst wenn die Mehrheit des Bitcoin Netzwerkes der Überweisung zustimmt, ist die Überweisung beendet.

Besonderheit bei der Blockchain ist, dass alle Transaktionen öffentlich für Jeden verfügbar in der Blockchain gespeichert werden. Damit kann jede Person jede Transaktion verifizieren. Die Transaktionen sind dabei in Blöcken gespeichert, wobei jeder Block durch den Hash des Vorgängerblocks verbunden ist. Dies schützt vor Manipulationen und ist der Grund für den Namen Blockchain.

Um die Unveränderbarkeit der Blockchain zu gewährleisten, wird die „Proof-of-Work“ Methode eingesetzt.

Der „Proof-of-Work“ kann folgendermaßen zusammengefasst werden: „Der Proof-of-Work Mechanismus ist eine Form der sogenannten Konsens-Mechanismen, um im Netzwerk einen Konsens zu erzielen und sich gemeinsam auf eine identische Version der Blockchain zu einigen.“[12]. Dabei bringt jeder „Proof-of-Work“ Teilnehmer eine gewisse Rechenleistung auf. Die von den „Proof-of-Work“ Teilnehmern aufgewendete Rechenpower schützt die Blockchain vor Veränderungen. Um die Blockchain zu verändern, müsste somit ein sehr großer Betrag an Rechenaufwand aufgewendet werden, der größer sein müsste als jener von allen Teilnehmern zusammen.

Da Arbeitsnachweise in der Blockchain nicht das Hauptthema dieser Arbeit sind, werden weitere Konsens-Mechanismen nicht weiter behandelt.

11 Vgl. https://en.bitcoin.it/wiki/How_bitcoin_works#Cryptography, (09.03.18)

12 Vgl. https://www.btc-echo.de/tutorial/was-ist-proof-of-work-wie-funktioniert-konsens-mechanismus/ (3.3.18)

[13]Zusammengefasst wird in der Bitcoin Blockchain als erstes mithilfe eines Zufallsgenerators ein privater Schlüssel erstellt. Mithilfe der Elliptic-Curve Multiplikation kann man den öffentlichen Schlüssel berechnen. Die dazu benötigte Funktion ist eine Einwegfunktion. Somit kann ein Dritter mithilfe des öffentlichen Schlüssels nicht einfach auf den privaten Schlüssel kommen. Mit dem öffentlichen Schlüssel und einer kryptographischen Hashfunktion berechnet man schlussendlich die Bitcoin-Adresse. Zu weiteren Informationen über das Hashing siehe unter 5.1.

3.2.1 Elliptic-Curve Kryptographie(ECC)[14]

Die Elliptic-Curve Kryptographie, ist unter anderem die Grundlage der Bitcoin-Blockchain. Sie stellt sicher, dass Bitcoins nur von den richtigen Besitzern ausgegeben werden können.
Der Name der Elliptische-kurven Kryptographie kommt daher, dass sie auf Basis von Elliptischen Kurven basiert. Dabei ist die Elliptic-Curve Kryptographie nicht als eigenständig anzusehen. Man sollte es eher als Erweiterung betrachten. Mit der ECC lassen sich Verfahren erweitern, die auf den diskreten Logarithmus basieren. Ein Beispiel hierfür ist der schon im ersten Teil behandelte Diffie-Hellman Algorithmus.

13 Vgl. Antonopoulos Andreas: Mastering Bitcoin: Unlocking Digital Cryptocurrencies , zweite Auflage, 16. Juni 2017, S. 67ff.

14 Vgl. Klaus Schmeh: Kryptographie, S. 226

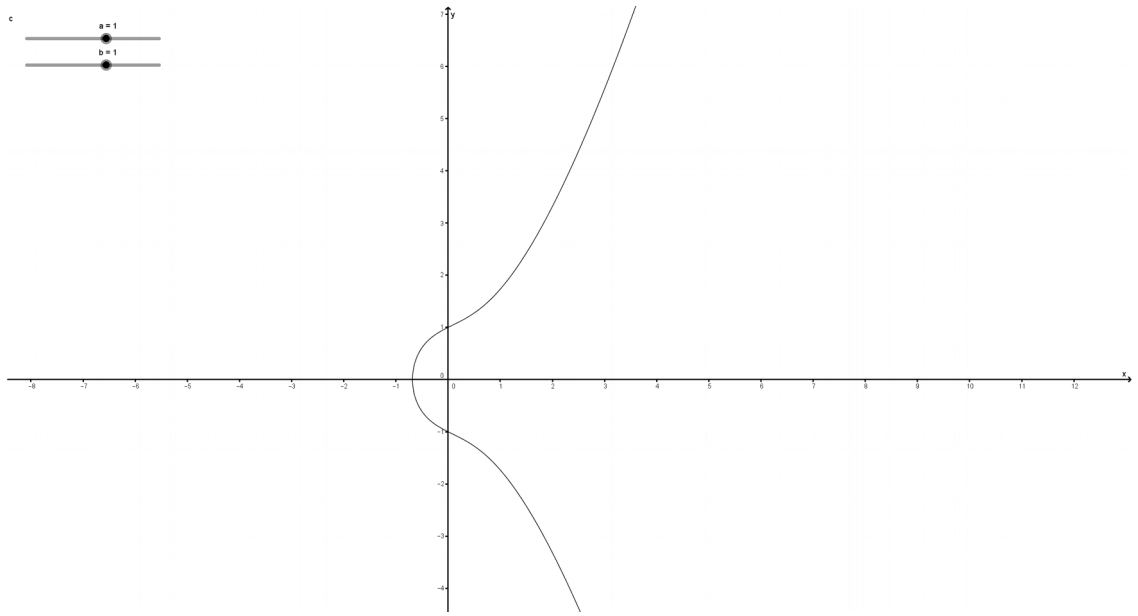


Abbildung 1: Darstellung einer Elliptischen Kurve mit der Grundgleichung $y^2 = x^3 + ax + b$ *(Quelle: Erstellt durch Daniel Vera Gilliard mit dem Programm GeoGebra)*

Eine Elliptische Kurve wird durch die allgemeine Gleichung: $y^2 = x^3 + ax + b$ beschrieben. Hinzu kommt noch ein Punkt im Unendlichen. Dieser Punkt wird oft als 0 bezeichnet. Eine verständliche Definition des Punktes im Unendlichen ist Folgende: „In [der] projektiven Geometrie definiert man zu jedem Paar paralleler Geraden einen ‚Fernpunkt', der unendlich weit entfernt ist und in dem sie sich schneiden. Aber dieser Punkt ist dann gleichberechtigt mit allen anderen Punkten, und in dieser Geometrie schneiden sich zwei Geraden immer. Manche in einem Punkt, der den Namen "Unendlich" trägt."[15]. Ebenso kann man sich den Punkt vorstellen, indem man ihm die gleiche Rolle wie der Zahl 0 in der „normalen" Addition gibt.

15 http://www.zeit.de/2017/09/geometrie-mathematik-parallele-geraden-unendlichkeit-stimmts (09.03.18)

Die in Abbildung 1 dargestellte Elliptische Kurve basiert auf reellen Zahlen. In der Kryptographie wird jedoch mit endlichen Primkörpern gerechnet. Dabei werden alle Berechnungen mit modulo p durchgeführt.

Zusammengefasst[16] lässt sich eine Elliptische Kurve in der Kryptographie über $\mathbb{Z}\,p, p>3$ mit der Gleichung: $y^2 = x^3 + ax + b \bmod p$ bezeichnen, wobei $x, y \in \mathbb{Z}\,p$, $a, b \in \mathbb{Z}\,p$ und die Bedingung $4a^3 + 27b^2 \neq 0 \bmod p$ ist. Da Elliptische Kurven auf endliche Körper keine gut darstellbare Figuren ergeben, werden zur Veranschaulichung von Elliptischen Kurven reelle Zahlen genommen.

3.2.1.1 Eigenschaften der Elliptic-Curve Kryptographie[17]

Elliptische Kurven haben eine grundlegende Eigenschaft, die sie für die Kryptographie sehr nützlich machen. Wenn eine Gerade die Kurve in zwei Schnittpunkten schneidet, gibt es auch einen dritten Schnittpunkt. Dabei gibt es zwei Spezialfälle. Wenn die Gerade parallel zur y-Achse ist, hat sie den dritten Schnittpunkt im Unendlichen (im oben angesprochenen Punkt 0). Ist die Gerade eine Tangente, hat ihr Berührpunkt einen Doppelten Schnittpunkt.

16 Vgl. Kryptographie Verständlich, S. 275

17 Vgl. Ebd., S. 227

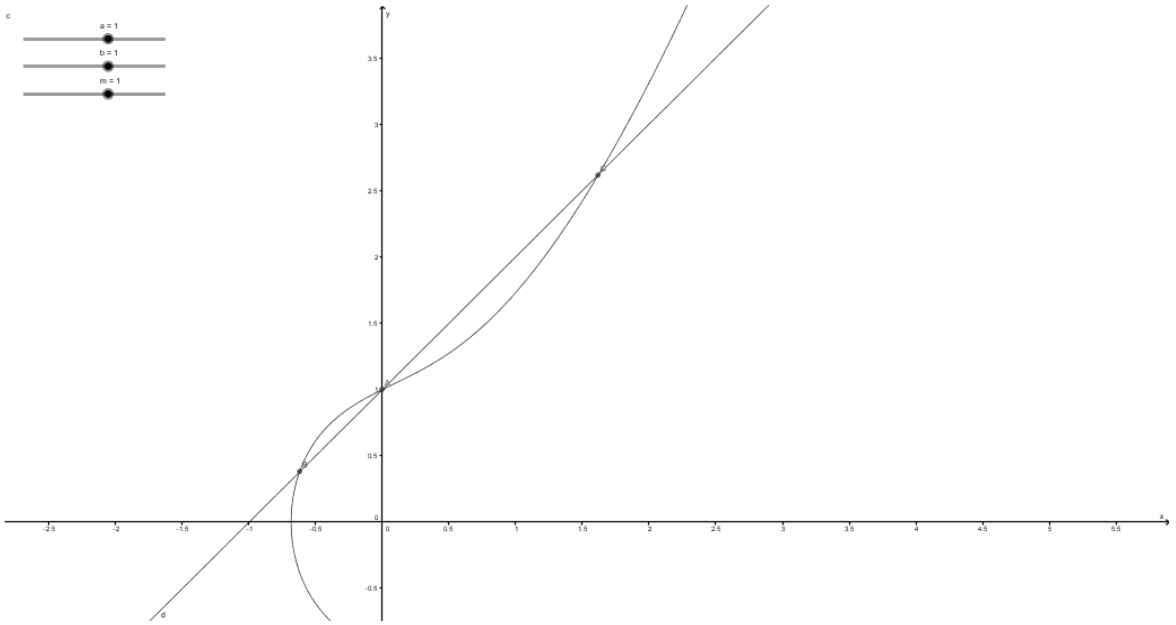


Abbildung 2: Wenn eine Gerade die Kurve in zwei Schnittpunkten schneidet, gibt es auch einen dritten Schnittpunkt (Erstellt von Daniel Vera Gilliard mit dem Programm GeoGebra).

Interessant dabei sind Möglichkeiten, mit denen man durch zwei Punkte auf einen dritten schließen kann. Bei einem der beiden Punkte handelt es sich um einen Generator Punkt der im nachfolgenden Abschnitt näher erklärt wird. Dabei kann man zum einen die „Punkt-Addition“ oder die „Punkt-Multiplikation“ benutzen“[18].

Bei der Punkt-Addition kann man eine Analogie zur der normalen Addition ganzer Zahlen machen. Dabei kann man zwei unterschiedliche Punkte $P1, P2$ auf einer Elliptischen Kurve addieren. Aufgrund der oben genannten Eigenschaft, geht die Gerade außer durch beide Punkten auch durch einen dritten Punkt. Diesen Punkt kann man mit der Punkt-Addition geometrisch bestimmen: $P1+P2=P3$.

18 Vgl. Ebd., S. 227

Wenn man jetzt das Ergebnis $P3'=(x,y)$ an der X-Achse spiegelt, erhält man den dritten Punkt $P3=(x,-y)$. Wichtig dabei, ist dass man die „Addition“ nicht wörtlich nimmt. Dies ist nur ein Name für die Methode und bedeutet nicht, dass man die zwei Punkte normal addieren kann. Der zur Veranschaulichung hilfreiche Graph wird im Abschnitt 2.2.1.2 gezeigt.

Bei der Punkt-Multiplikation geht eine Gerade durch zwei Punkte einer Elliptischen Kurve. Durch die oben erklärte Eigenschaft, muss die Gerade noch durch einen dritten Punkt gehen. Wenn man das Ergebnis der Multiplikation als Punkt betrachtet und wie bei der Addition diesen an der x-Achse spiegelt, erhält man den dritten Punk:. $P1*P2=P3$. Die Multiplikation kann man, wie in der Algebra basierend auf den Reellen Zahlen, auch als erweiterte Additionen darstellen. Dabei nimmt man die obigen Regeln der Punkt-Addition und erhält $Pk=P+P+P+P...+P(k\,mal)$. k kann auch als Exponent gesehen werden. Die Punkt-Multiplikation ist aus dem Grund hilfreich, da man mit ihr den dritten Punkt berechnen kann, falls die Gerade eine Tangente sein sollte. Wie auch bei der Addition, ist die Multiplikation nicht wörtlich als solche zu verstehen, sondern eher als Name für die Methode.
Die Möglichkeiten der Punkt-Addition und Punkt-Multiplikation geben der Elliptic-Curve Kryptographie ihre Schnelligkeit im Vergleich zu anderen Kryptographischen Verfahren, wie das im Abschnitt 2.3.2 behandelte RSA-Verfahren.
Ein Beispiel soll diese Überlegungen verdeutlichen:
Angenommen man hat den Punkt P und möchte den Punkt 100P ausrechnen:

1. Zum einen könnte man den Punkt P 100 mal mit sich selbst multiplizieren. Dies ist aber sehr ineffizient.
2. Mit der Punkt-Addition kann man das Vorgehen vereinfachen: $2P=P*P \rightarrow 2P+P=3P \rightarrow 3P^2=6P \rightarrow 6P^2=12P \rightarrow 12P^2=24P \rightarrow 24+P=25P \rightarrow 25P^2=50P \rightarrow 50P^2=100P$ [19]

Anstatt mit 99 Schritten, kann man den Punkt mithilfe der Elliptic-Curve Kryptographie mit nur 8 Schritten berechnen. Eine graphische Verdeutlichung dieser Methode ist in Abbildung 3 im Abschnitt 3.2.1.3 dargestellt.

19 Vgl. https://blockgeeks.com/guides/cryptocurrencies-cryptography/ (24.2.18)

Im Vergleich zu Multiplikation ist die Division bei der Elliptic-Curve Kryptographie sehr langsam. Dies ist jedoch von Vorteil. Denn man kann dann einfach den öffentlichen Schlüssel berechnen, jedoch schwer den privaten. Mehr dazu in Abschnitt 2.2.1.3.

3.2.1.2 Einsatz der Elliptic-Curve Kryptographie in der Bitcoin Blockchain

Die Bitcoin Blockchain benutzt ein Set von bestimmten mathematischen Konstanten, die im secp256k1 Standard festgelegt sind. Dieser Standard wurde von dem „National Intitute of Standarts and Technologie“(NIST)[20] festgelegt. Die Elliptische Kurve, die nach dem NIST festgelegt wurde, hat die Gleichung $y^2=x^3+ax+b$.

Auf Seite 9 des Dokumentes „SEC 2: Recommended Elliptic-Curve Domain Parameters“[21] gibt das NIST Aufschluss über die Werte der verschiedenen Parameter. Dabei hat die Primzahl p den Wert $2^{256}-2^{32}-2^9-2^8-2^7-2^6-2^4-1$, a den Wert 0 und b den Wert 7. Mit den eingesetzten Werten sieht die Gleichung folgendermaßen aus: $y^2 \bmod p=(x^3+7) \bmod p$.

Zu der Gleichung wird noch $\bmod p$ hinzugefügt, da man die Berechnungen in einem endlichen Feld Fp macht. Dieses endliche Feld Fp definiert sich als $Fp=\{0,..,p-1\}$ und ist sehr schwer zu visualisieren.

20 Vgl. http://www.secg.org/ (09.03.18)

21 Vgl. http://www.secg.org/sec2-v2.pdf (09.03.18), S. 9

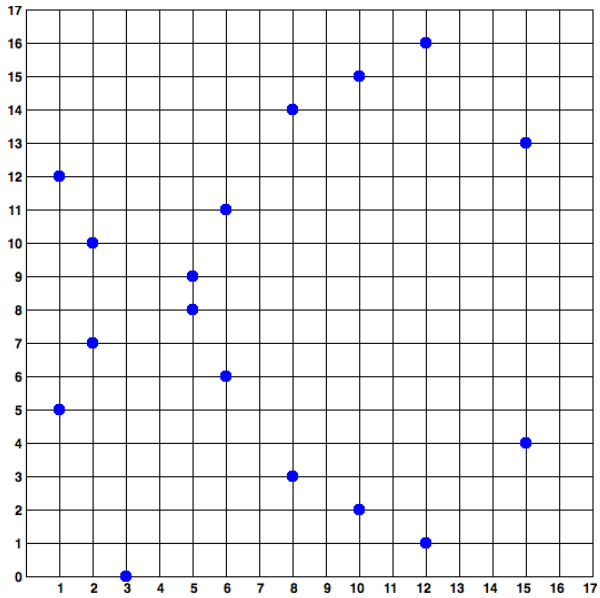


Abbildung 3: Darstellung einer Elliptischen Kurve über Fp mit p=17(Quelle: Quelle: Antonopoulos Andreas: Mastering Bitcoin: Unlocking Digital Cryptocurrencies , zweite Auflage, 16. Juni 2017, S. 67)

Damit jeder Teilnehmer Punkte auf der Kurve berechnen kann, braucht man einen sogenannten Generator Punkt $G=(Xg,Yg)$. Das NIST gibt dabei zwei Varianten des Generator Punktes an:

Zum einen eine komprimierte Form mit dem Präfix 02:
$G=02\,79\,BE\,667\,E\,F\,9\,DCBBAC\,55\,A\,06295\,CE\,870\,B\,07\,029\,BFCDB\,2\,DCE\,28\,D\,9\,59\,F\,2815\,B\,16\,F\,81798$
Wenn man das Präfix 02 weglässt, erhält man die x-Koordinate:
$G=79\,BE\,667\,E\,F\,9\,DCBBAC\,55\,A\,06295\,CE\,870\,B\,07\,029\,BFCDB\,2\,DCE\,28\,D\,9\,59\,F\,2815\,B\,16\,F\,81798$

Zum anderen gibt das NIST noch eine unkomprimierte Fassung an, mit dem Präfix 04: G = 04 79BE667E F9DCBBAC 55A06295 CE870B07 029BFCDB 2DCE28D9 59F2815B

16F81798 483ADA77 26A3C465 5DA4FBFC 0E1108A8 FD17B448 A6855419 9C47D08F FB10D4B8[22].

Wenn man dabei das Präfix 04 weglässt, kann man diese Zahl in zwei Teile aufteilen:

Die x-Koordinate: 79BE667E F9DCBBAC 55A06295 CE870B07 029BFCDB 2DCE28D9 59F2815B 16F81798

und die y-Koordinate: Gy= 483ADA77 26A3C465 5DA4FBFC 0E1108A8 FD17B448 A6855419 9C47D08F FB10D4B8.

Zuletzt gibt es noch die Konstanten N und H. N entspricht hierbei der größten Zahl, für die ein privater Schlüssel gebildet werden kann. Dabei kann der private Schlüssel jede Zahl im Rahmen von [1, n-1] sein. N ist auf: FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE BAAEDCE6 AF48A03B BFD25E8C D0364141 definiert worden.[23]. Die Konstante H ist mit dem Wert 1 definiert und spielt somit keine Rolle in der Schlüsselgenerierung.

3.2.1.3 Berechnung des öffentlichen Schlüssels

Der öffentliche Schlüssel einer Bitcoin Adresse wird durch den zufällig generierten privaten Schlüssel und der Elliptic-Curve Kryptographie erstellt.

Die Formel für die Berechnung lautet folgendermaßen: $K = k * G$.

G steht für den Generator Punkt und wird im oberen Abschnitt 2.2.1.2 näher erläutert.

k ist dabei der zufällig generierte private Schlüssel des Nutzers.

Die Umkehroperation, k aus dem öffentlichen Schlüssel K und dem Punkt G zu berechnen, nennt sich „den diskreten Logarithmus suchen“.

Dies ist sehr schwer oder gar unmöglich und stellt eine Sicherheit gegen die Berechnung des privaten Schlüssels k auf. Da G für jeden Bitcoin-Nutzer gleich ist, wird die Multiplikation von k mit G immer das gleiche Ergebnis liefern[24].

In der realen Welt werden diese Berechnungen mit sehr großen Zahlen gemacht. Um es jedoch zu vereinfachen, wird im Folgenden mit kleinen Zahlen gerechnet.

22 Vgl. Ebd., S. 9

23Vgl. Ebd.

24 Vgl. Antonopoulos Andreas: Mastering Bitcoin: Unlocking Digital Cryptocurrencies , zweite Auflage, 16. Juni 2017, S. 69

Die Berechnung des öffentlichen Schlüssels $K(x,y)$ kann man auch als Potenzierung oder Multiplikation von G sehen. Dabei wird G mit dem privaten Schlüssel k multipliziert. k steht dann in diesem Fall für den Exponenten. Rechnerisch ist dies folgendermaßen darzustellen: $K = k * G = G + G + G ... + G(k\,mal)$. Zeichnerisch gleicht dies einer Tangente, zu der man den dritten Punkt ermittelt und dabei die Spiegelung an der x-Achse macht:

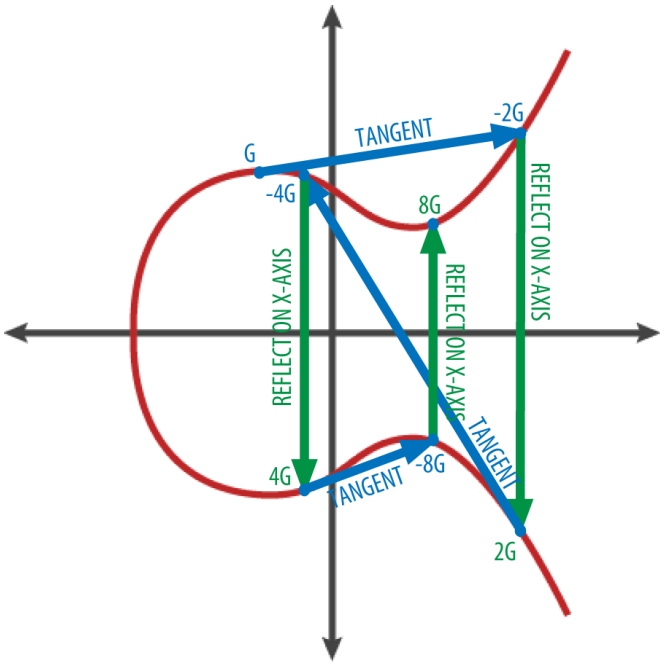


Abbildung 4: Darstellung der Multiplikation von dem Punkt G mit einer Ganzzahl k auf einer Elliptischen Kurve(Quelle: Antonopoulos Andreas: Mastering Bitcoin: Unlocking Digital Cryptocurrencies , zweite Auflage, 16. Juni 2017, S. 70).

3.2.1.4 Benutzung der Elliptic-Curve Kryptographie mit dem Diffie-Hellman Protokoll

Wie eingangs erwähnt, ist die Elliptic-Curve Kryptographie kein eigenständiges Kryptographisches Verfahren. Man kann es mit anderen Verfahren kombinieren, um ein bestehendes System zu verbessern.

In diesem Abschnitt wird die Elliptic-Curve Kryptographie zusammen mit dem Diffie-Hellman Protokoll genutzt.

Im nachfolgenden Beispiel werde ich Alice(A) und Bob(B) als Beispielpersonen benutzen.

Der Vorgang ist folgender:[25]

25 Vgl. https://blockgeeks.com/guides/cryptocurrencies-cryptography/ (24.2.18)

1. Alice und Bob einigen sich auf einen gemeinsamen Punkt P. Dieser Punkt entspricht dem im Abschnitt 2.2.1.2 behandelten Generator Punkt. Der Punkt P ist dabei für jeden frei verfügbar.
2. Als nächstes wählt Alice einen geheimen Punkt a und Bob einen geheimen Punkt b. Diese Punkte agieren als private Schlüssel.
3. Mithilfe der im Abschnitt 2.2.1.1 behandelten Punkt-Addition berechnet Alice $aP = a * P$ aus ihrem privaten Schlüssel und dem öffentlichen Punkt P. Alice schickt daraufhin Bob das Ergebnis aP zu.
4. Bob berechnet ebenfalls bP : $bP = b * P$ und sendet bP Alice zu.
5. Alice und Bob multiplizieren jetzt die Nachricht (aP , bP), die sie mit ihrem privaten Schlüssel(a, b): $a(bP)$ und $b(aP)$ bekommen.
   Wie im Abschnitt 2.3.1.1 über das Diffie-Hellman Protokoll schon näher erläutert gilt: $a(bP) = b(aP)$.

3.2.2 Signaturen in der Bitcoin Blockchain[26]

Für die nachfolgenden Gleichungen werden die Variablen d für den privaten Schlüssel, z für die Nachricht, Q für den öffentlichen Schlüssel, k für eine zufällige Zahl, die für jede Signatur neu erstellt wird, und n sowie G als Konstanten, die in Abschnitt 2.2.1.2 ausführlicher behandelt wurden, benutzt.
Der Signierungsvorgang einer Transaktion in der Bitcoin Blockchain ist der Folgender:

1. Wie in 2.2.1.3 beschrieben, berechnet sich der Öffentliche Schlüssel aus $Q = dG$
2. Man multipliziert den Punkt G mit der zufälligen Zahl k und erhält die Koordinaten: $(x, y) = kG$.
3. Als nächstes werden zwei weitere Werte r und s ermittelt. Diese Werte sind nötig, da man mit ihnen die Koordinaten für die Signatur erhalten kann. Diese Koordinaten werden dann zum Vergleich an den Empfänger geschickt, der die Nachricht verifizieren kann. In der Bitcoin Blockchain werden die Werte r und s mit folgenden Formeln berechnet: $r = x \bmod n$ und $s = (z + rd)k^{-1} \bmod n$. Die Variable x für die Berechnung von r wird aus dem zweiten Schritt entnommen.

26 Vgl. Ebd.

Schlussendlich wird der Punkt (r,s) an den Empfänger der Nachricht zur Verifizierung geschickt.

4. Als letzten Schritt bleibt die Verifizierung der Nachricht durch den Empfänger. Der Empfänger berechnet dabei den x-Wert mit den empfangenen Informationen des Punktes (r,s) .

   Dies geschieht mit folgender Formel: $z*s^{-1}*G+r*s^{-1}*Q=(x,y)$.

   Mit den Werten r und n berechnet der Empfänger eigenständig die X-Koordinate $r=x \bmod n$ und vergleicht die Ergebnisse beider Rechnungen. Wenn beide Rechnungen gleich sind, ist die Signatur verifiziert.

3.2.2.1 Beweis der Formel zur Verifizierung

Die Formel $z*s^{-1}*G+r*s^{-1}*Q=(x,y)$ ergibt den x-Wert, den man für die Verifizierung braucht. Dass dies stimmt, kann man durch das Vereinfachen der Gleichung beweisen:

1. Als erstes ersetzt man Q durch $d*G$, da $Q=d*G$: $z*s^{-1}*G+r*s^{-1}*d*G$
2. Als zweites kann man $(z+R*d)$ ausklammern: $(z+r*d)*s^{-1}*G$
3. Aus dem Abschnitt 2.2.2 kann man noch den Wert $s=(z+rd)k^{-1} \bmod n$ entnehmen. Dies in die Gleichung eingesetzt, ergibt: $(z+r*d)*(z+r*d)^{-1}*k*G$.
4. Nachdem man $(z+r*d)*(z+r*d)^{-1}$ auch als $(z+r*d)*1/(z+r*d)$ schreiben kann, kommt man nach dem Kürzen auf: $k*G=(x,y)$.

3.2.3 Vorteile der Elliptic-Curve Kryptographie gegenüber dem RSA Algorithmus

Der wichtigste Grund, die ECC gegenüber dem RSA Algorithmus zu benutzen, ist die Länge der Schlüssel. Bei der ECC erreicht man das gleiche Sicherheitslevel wie bei dem RSA, jedoch mit einem viel kleineren Schlüssel. Für ein 256-bit Schlüssel im ECC Verfahren bräuchte man einen 3072-bit Schlüssel, um das gleiche Sicherheitslevel im RSA Verfahren zu gewährleisten.

4 Schluss

Zusammengefasst ist die Kryptographie basierend auf den Elliptischen Kurven ein sehr sicheres und schnelles Verfahren.
Dennoch ist dies nur der Anfang. Mit dem Populärwerden des Bitcoin startete eine sehr große Welle der Begeisterung über die Blockchain. Dies hatte zur Folge, dass die dafür benötigte Kryptographie einen Aufschwung erhielt.
Neben Bitcoin sind heutzutage noch sehr viele andere Kryptowährungen und Blockchain-Anwendungen geschaffen worden. Aus kryptographischer Sicht interessant sind Kryptowährungen, die sich auf die Privatsphäre fokussiert haben. Monero, Zcash oder Dash sind nur einige davon.
Aus diesem Grund bleibt es spannend, welche kryptographischen Methoden in Zukunft geschaffen werden und welche sich schlussendlich durchsetzen werden.

5 Anhang

5.1 Hashing

Mithilfe einer Hashfunktion kann man Informationen unterschiedlicher Länge in einen Datensatz von bestimmter Länge umwandeln. Dabei bekommt man immer für den gleichen Input den gleichen Output in Form eines Hashes. Wenn jedoch nur ein Datenbestandteil geändert wird, ändert sich der daraus resultierende Hash komplett.
Bitcoin benutzt dabei den SHA-256 Hash Algorithmus um Zufallszahlen zu erzeugen. Vorteil bei der Methode des Hashings ist, dass man theoretisch unmöglich mithilfe eines Hashes die die Ausgangszahl herausfinden kann.[27]
In der Bitcoin Blockchain wird Hashing oft genutzt. Zur Anwendung kommt dies zum Beispiel bei der Erstellung von Schlüsseln oder der Benutzung für den Konsensus Algorithmus (Proof of Work).[28]

27 Vgl. https://de.bitcoin.it/wiki/Hash (26.03.18)

28 Vgl. https://en.bitcoin.it/wiki/Block_hashing_algorithm (26.03.18)

6 Literatur- und Quellenverzeichnis

6.1 Literatur

- Antonopoulos Andreas: Mastering Bitcoin: Unlocking Digital Cryptocurrencies , zweite Auflage, 16. Juni 2017
- Ernst Hartmut, Schmidt Jochen, Beneken Gerd: Grundkurs Informatik: Grundlagen und Konzepte für die erfolgreiche IT-Praxis - eine umfassende, praxisorientierte Einführung, - 5., vollst. überarb. Aufl., 20. März 2015
- Ertel Wolfgang: Angewandte Kryptographie, 4. überarbeitete und ergänzte Auflage, 5. Juli 2012
- Haffner. E. G.: Informatik für Dummies : das Lehrbuch, - 1. Auflage, 15. Februar 2017
- Haftendorn, Dörte: Mathematik sehen und verstehen: Schlüssel zur Welt, 11. März 2010
- Hajo Schulz: Das macht Blockchain. Die Technik hinter Bitcoin & Co., in: c‘t Magazin für Computertechnik, 28 August 2017, Nr 23, S 103 – 106.
- Paar Christof, Pelzl Jan: Kryptografie verständlich: Ein Lehrbuch für Studierende und Anwender, 25. September 2016
- Schmeh Klaus: Kryptografie. Verfahren, Protokolle, Infrastrukturen, 6. aktualisierte Auflage, April 2016

6.2 Internetdokumente

- „A gentle introduction to blockchain technology“ (09.11.2015), online unter URL: <https://bitsonblocks.net/2015/09/09/a-gentle-introduction-to-blockchain-technology/>, (09.03.18)
- „Block hashing algorithm“, online unter URL: <https://en.bitcoin.it/wiki/Block_hashing_algorithm>, (26.03.18)
- „Blockchain tutorial 11: Elliptic Curve key pair generation“ (10.04.2017), online unter URL: <https://www.youtube.com/watch?v=wpLQZhqdPaA&index=8&list=PLkyGSjskdfq8-WPORX-ZM_t_P8cV-kR6z>, (09.03.18)

- „Elliptic Curve Digital Signature Algorithm“, online unter URL: <https://en.bitcoin.it/wiki/Elliptic_Curve_Digital_Signature_Algorithm>, (09.03.18)
- „Hash“, online unter URL: <https://de.bitcoin.it/wiki/Hash>, (26.03.18)
- „How bitcoin works“, online unter URL: <https://en.bitcoin.it/wiki/How_bitcoin_works#Cryptography>, (09.03.18)
- „Hybride Verfahren“ (04.11.2010), online unter URL: <http://ddi.cs.uni-potsdam.de/Lehre/e-commerce/elBez2-5/page07.html>, (09.03.18)
- „Private key“, online unter URL: <https://en.bitcoin.it/wiki/Private_key>
- „Restklasse“, online unter URL: <https://de.wikipedia.org/wiki/Restklasse>, (09.03.18)
- „RSA-Verfahren (Ver- und Entschlüsseln)“ (09.04.2016), online unter URL: <https://www.youtube.com/watch?v=AIkS0r3Cuic&index=5&list=PLkyGSjskdfq8-WPORX-ZM_t_P8cV-kR6z>, (09.03.18)
- Aron, Manuel, Tobias und Steffen: „RSA-Verschlüsselung (mathematisch) | private und public key berechnen“ (07.04.2014), online unter URL: <https://www.youtube.com/watch?v=mnN2aV3OhM>, (09.03.18)
- Bergische Universität Wuppertal: „Diffie-Hellman-Algorithmus Schlüsselaustausch“, online unter URL: „http://ddi.uni-wuppertal.de/material/spioncamp/dl/austausch-diffie-hellman-station.pdf“, (09.03.18)
- Brünner Arndt: „Primzahlen“ (2005), online unter URL: <http://www.arndt-bruenner.de/mathe/scripts/primzahlen.htm>, (09.03.18)
- Busse, Michael, Schmitt, Matthias, Steeg, Jörg: „Der RSA-Algorithmus“ (08. 04. 1999), online unter URL: <https://www.zum.de/Faecher/Inf/RP/infschul/kr_rsa.html>, (09.03.18)
- Daniel R. L. Brown: „Standards for Efficient Cryptography SEC 1: Elliptic Curve Cryptography“ (May 21, 2009) Version 2.0, online unter URL: <http://www.secg.org/sec1-v2.pdf> (09.03.18)
- Daniel R. L. Brown: „Standards for Efficient Cryptography SEC 2: Recommended Elliptic Curve Domain Parameters“ (January 27, 2010) Version 2.0, online unter URL: <http://www.secg.org/sec2-v2.pdf> (09.03.18)
- Drösser Christoph: „Schneiden sich zwei parallele Geraden im Unendlichen?“ (9. März 2017, 2:45 Uhr), online unter URL: <http://www.zeit.de/2017/09/geometrie-mathematik-parallele-geraden-unendlichkeit-stimmts>, (09.03.18)

- Hamasni Karim, Earl James: „The Cryptography Behind Bitcoin“ (16.05.2015), online unter URL: <https://www.youtube.com/watch?v=5fSOd431l6A&list=PLkyGSjskdfq8-WPORX-ZM_t_P8cV-kR6z&index=4>, (09.03.18)
- Prof. Christian Spannagel: „RSA: Beispiel Teil 1“ (03.07.2012), online unter URL: <https://www.youtube.com/watch?v=XR6zeI_rNPw>, (09.03.18)
- Rosic Armeer: „The Science Behind Cryptocurrencies Cryptography“, online unter URL: <https://blockgeeks.com/guides/cryptocurrencies-cryptography/>, (09.03.18)
- Song, Jimmy: „Dev++ | Jimy Song - Foundational Math, ECDSA and Transactions“ (25.01.2018), online unter URL: <https://www.youtube.com/watch?v=e6voIwB-An4&list=PLkyGSjskdfq8-WPORX-ZM_t_P8cV-kR6z&index=14>, (09.03.18)
- Wagnon, John: „Elliptic Curve Cryptography Overview“ (14.10.2015), online unter URL: <https://www.youtube.com/watch?v=dCvB-mhkT0w&list=PLkyGSjskdfq8-WPORX-ZM_t_P8cV-kR6z&index=7>, (09.03.18)

6.3 Bilder

- Titelbild: in: <https://blockchain.info/fr/wallet/#/> (23.02.2018).

7 Eidesstattliche Erklärung

Ich erkläre, dass ich die Arbeit selbstständig angefertigt und nur die angegebenen Hilfsmittel benutzt habe. Alle Stellen, die dem Wortlaut oder dem Sinn nach anderen Werken, gegebenenfalls auch elektronischen Medien, entnommen sind, sind von mir durch Angabe der Quelle als Entlehnung kenntlich gemacht. Dies gilt auch für Zeichnungen, Skizzen, Bilder und andere visuelle Darstellungen.

Ort, Datum, Unterschrift