Elliptische Kurven und ihre Anwendungen in der Kryptographie

Heiko Knospe

Fachhochschule Köln
heiko.knospe@fh-koeln.de

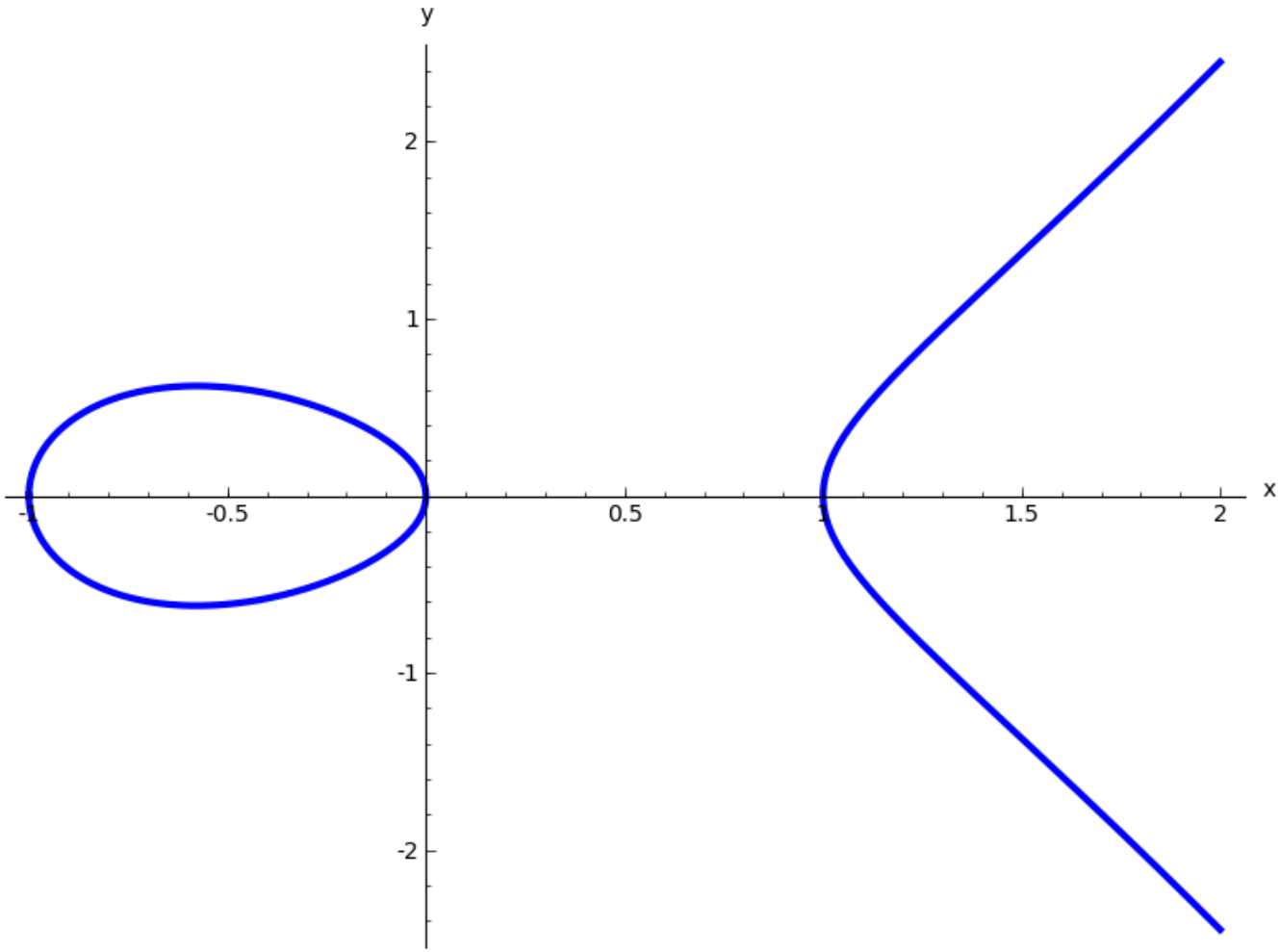
29. März 2014

Weierstraß-Gleichung

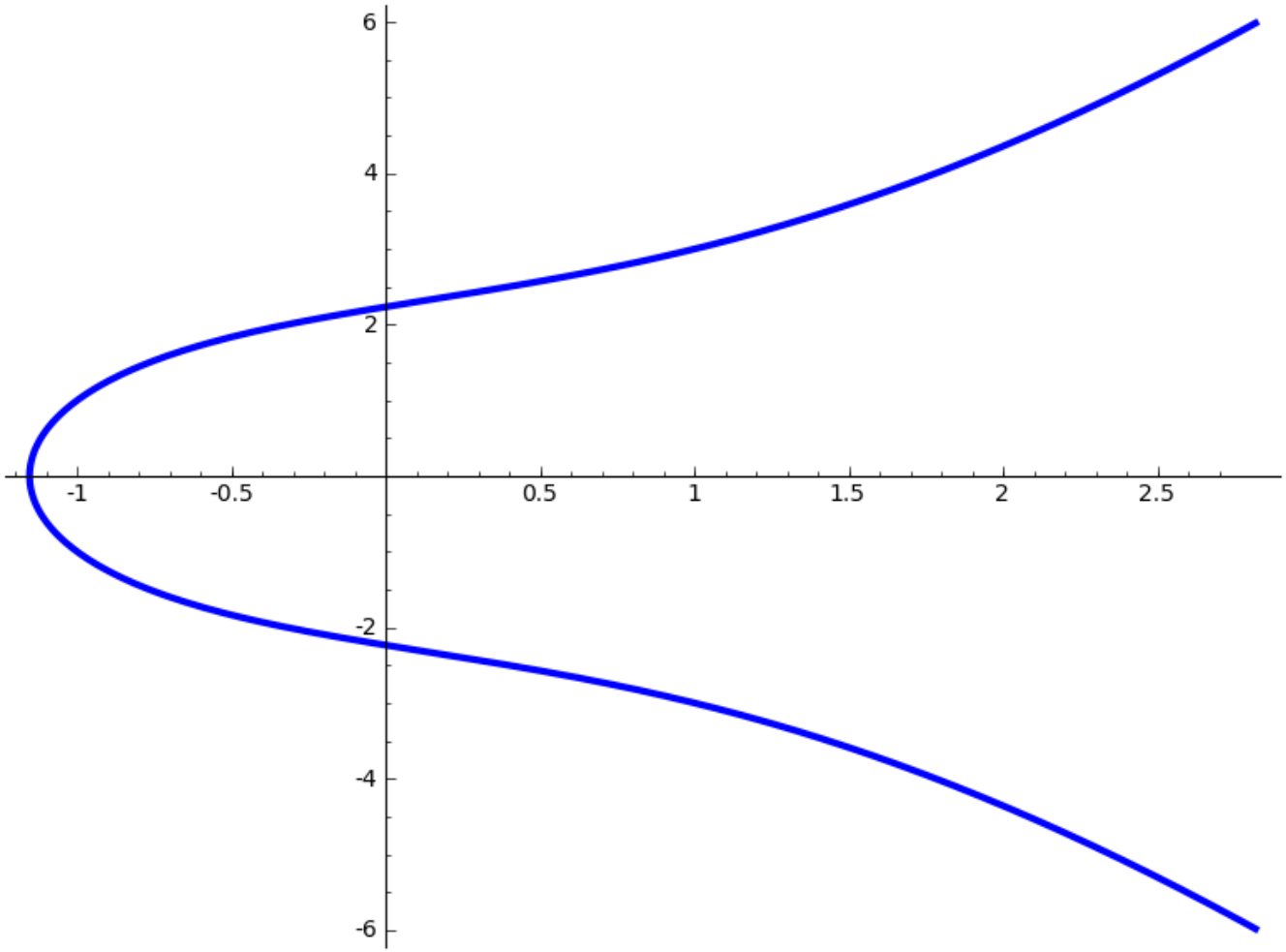
Elliptische Kurven sind nicht-singuläre Kurven, die durch eine *Weierstraß-Gleichung* definiert sind:

$$E: \ y^2 = x^3 + ax + b$$

Beispiel: $y^2 = x^3 - x,\ x, y \in \mathbb{R}$



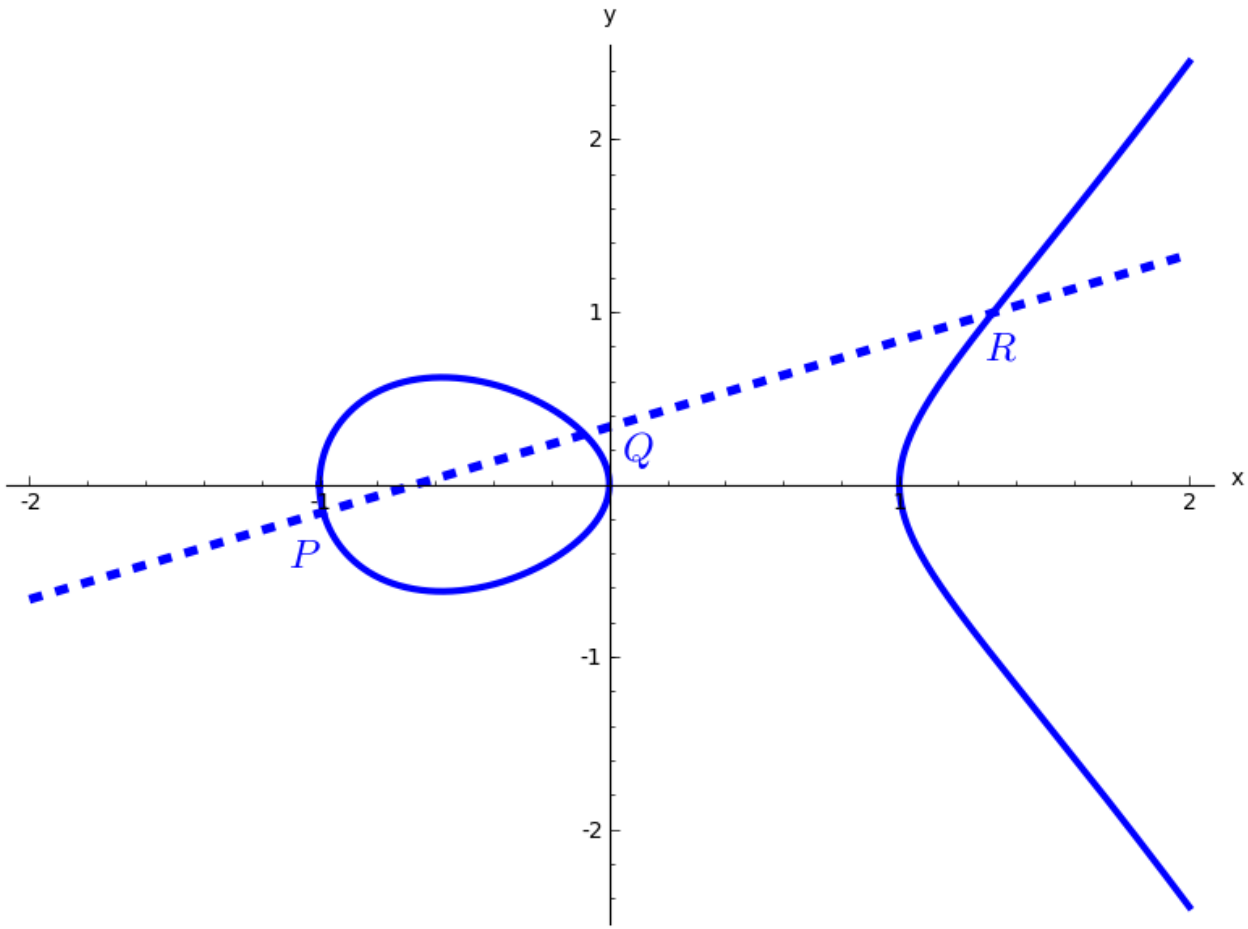
$y^2 = x^3 + 3x + 5,\ x, y \in \mathbb{R}$



Gruppe $E(K)$

Elliptische Kurven haben die Eigenschaft, dass ihre Punktemenge eine *abelsche Gruppe* bildet. Das neutrale Element ist der *projektive* Punkt $O = [0 : 1 : 0] \in \mathbb{P}^2(K)$.

$$P \oplus Q \oplus R = O \text{ bzw. } P \oplus Q = -R$$



Körper K

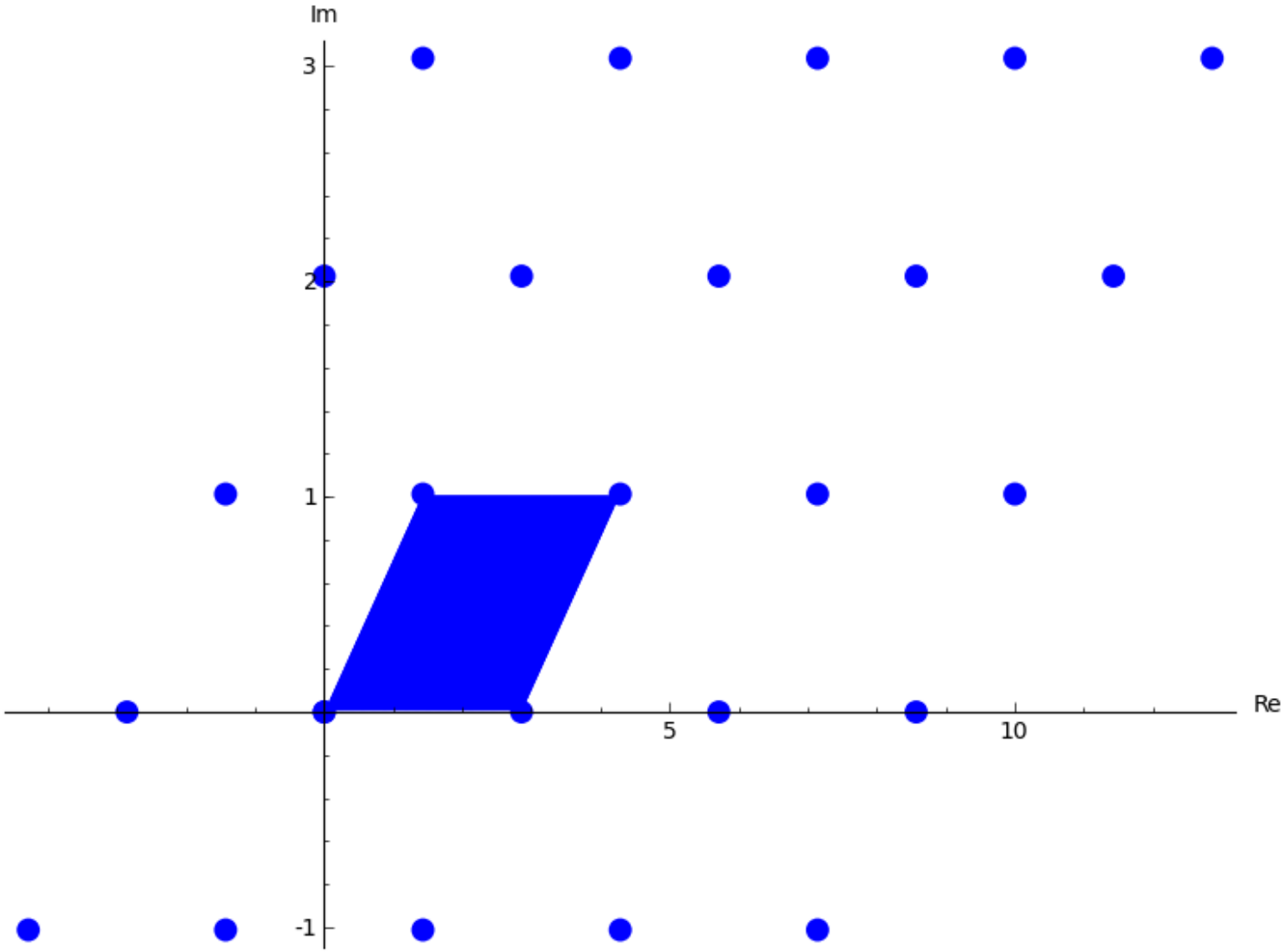
Elliptische Kurven kann man über unterschiedlichen Körpern K betrachten, außer $K = \mathbb{R}$ insbesondere:

- $K = \mathbb{C}$ (komplexe Zahlen)
- $K = \mathbb{Q}$ (rationale Zahlen)
- $K = GF(p)$ (endlicher Körper mit p Elementen, wobei p eine Primzahl ist). Die Elemente von $GF(p)$ sind die Restklassen modulo p:

$$GF(p) = \{\overline{0}, \overline{1}, \overline{2}, \ldots, \overline{p-1}\}$$

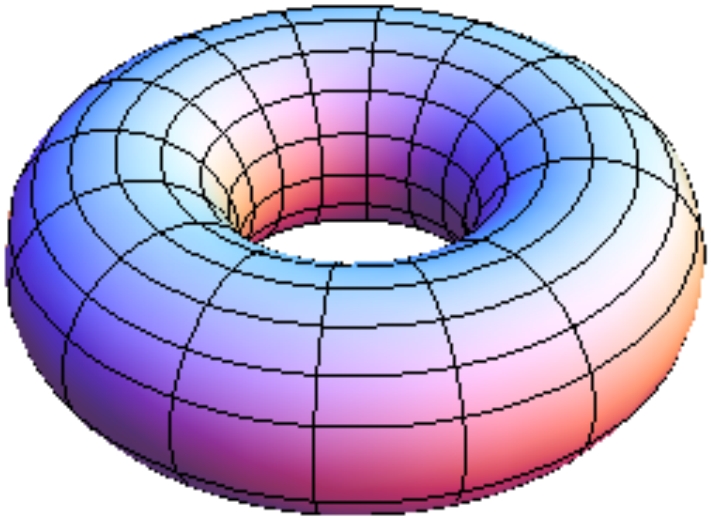
Elliptische Kurven über $\mathbb{C}$

Für $K = \mathbb{C}$ ist $E(\mathbb{C})$ isomorph zu einem komplexen Torus $\mathbb{C}/\Lambda$, wobei $\Lambda \subset \mathbb{C}$ ein Gitter in $\mathbb{C}$ ist.



Komplexe Punktegruppe

Die Faktorgruppe $\mathbb{C}/\Lambda$ ist topologisch ein Torus:



Die Isomorphie $E(\mathbb{C}) \cong \mathbb{C}/\Lambda$ ist nichttrivial und wird über elliptische Integrale bzw. die Weierstraßsche $\wp$-Funktion hergestellt. Jedes Gitter $\Lambda \subset \mathbb{C}$ liefert auf diese Weise eine elliptische Kurve E_Λ.

Arithmetik elliptischer Kurven

Sei K ein Zahlkörper ($K = \mathbb{Q}$ oder eine endliche Körpererweiterung von $\mathbb{Q}$). Die Untersuchung der Mordell-Weil Gruppe $E(K)$, ist seit Jahrzehnten Gegenstand mathematischer Forschung.

Satz von Mordell-Weil $E(K)$ ist eine endlich erzeugte abelsche Gruppe und

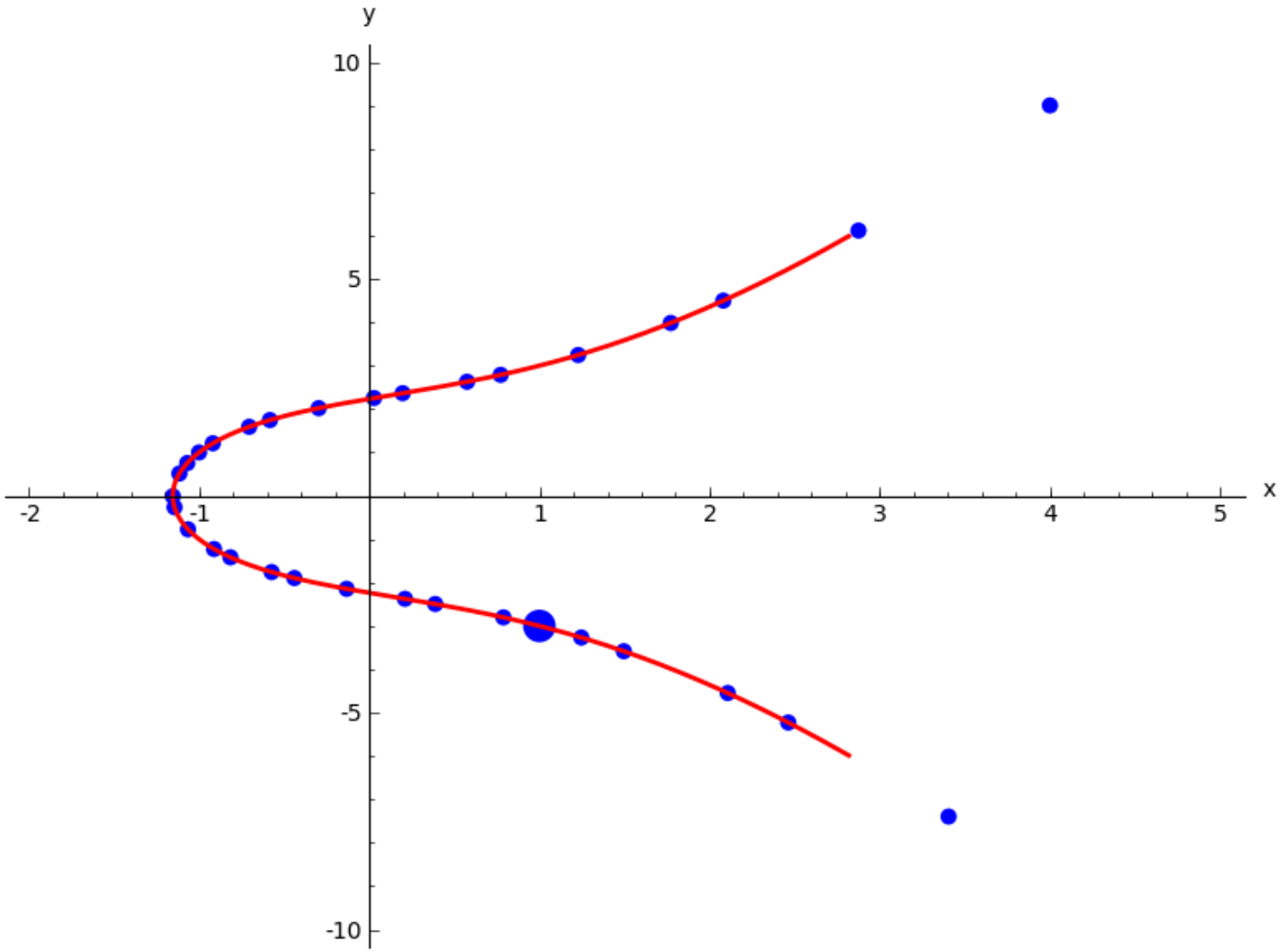
$$E(K) \cong \mathbb{Z}^r \oplus \mathbb{Z}/m_1\mathbb{Z} \oplus \mathbb{Z}/m_2\mathbb{Z}$$

wobei $r \geq 0$, $m_1, m_2 \geq 1$.

r ist der *Rang* der Punktegruppe und die endlichen Gruppen (die trivial sein können) bilden den Torsionsanteil. Es gibt Theoreme und Vermutungen über den Zusammenhang zwischen der L-Funktion von E und r.

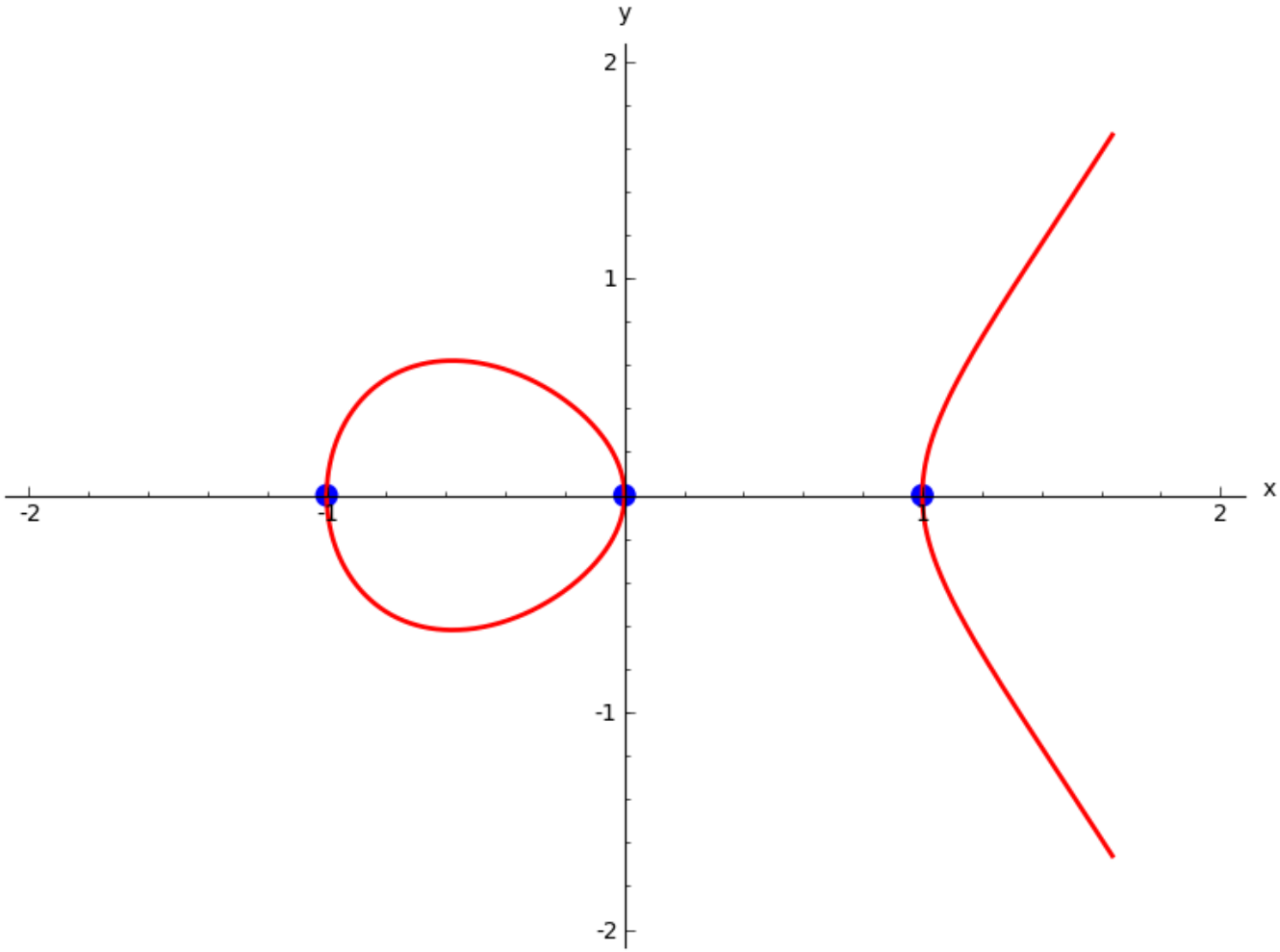
Mordell-Weil Gruppe

Für die elliptische Kurve $y^2 = x^3 + 3x + 5$ gilt z.B. $E(\mathbb{Q}) \cong \mathbb{Z}$ und der Punkt $G = (1, -3)$ ist ein Erzeuger dieser Gruppe. Es gilt $2 \cdot G = (-1, 1)$, $3 \cdot G = (4, 9)$, $4 \cdot G = (11, -37)$ usw.



Mordell-Weil Gruppe (2)

Für die elliptische Kurve $y^2 = x^3 - x$ gilt z.B.
$E(\mathbb{Q}) \cong \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$.



Elliptische Kurven über endlichen Körpern

Die Punktegruppe einer elliptischen Kurve über einem endlichen Körper ist endlich. Die Zahl N_p der Punkte $E(GF(p))$ weicht nicht allzu stark von p ab (Theorem von Hasse):

$$|p + 1 - N_p| \leq 2\sqrt{p}$$

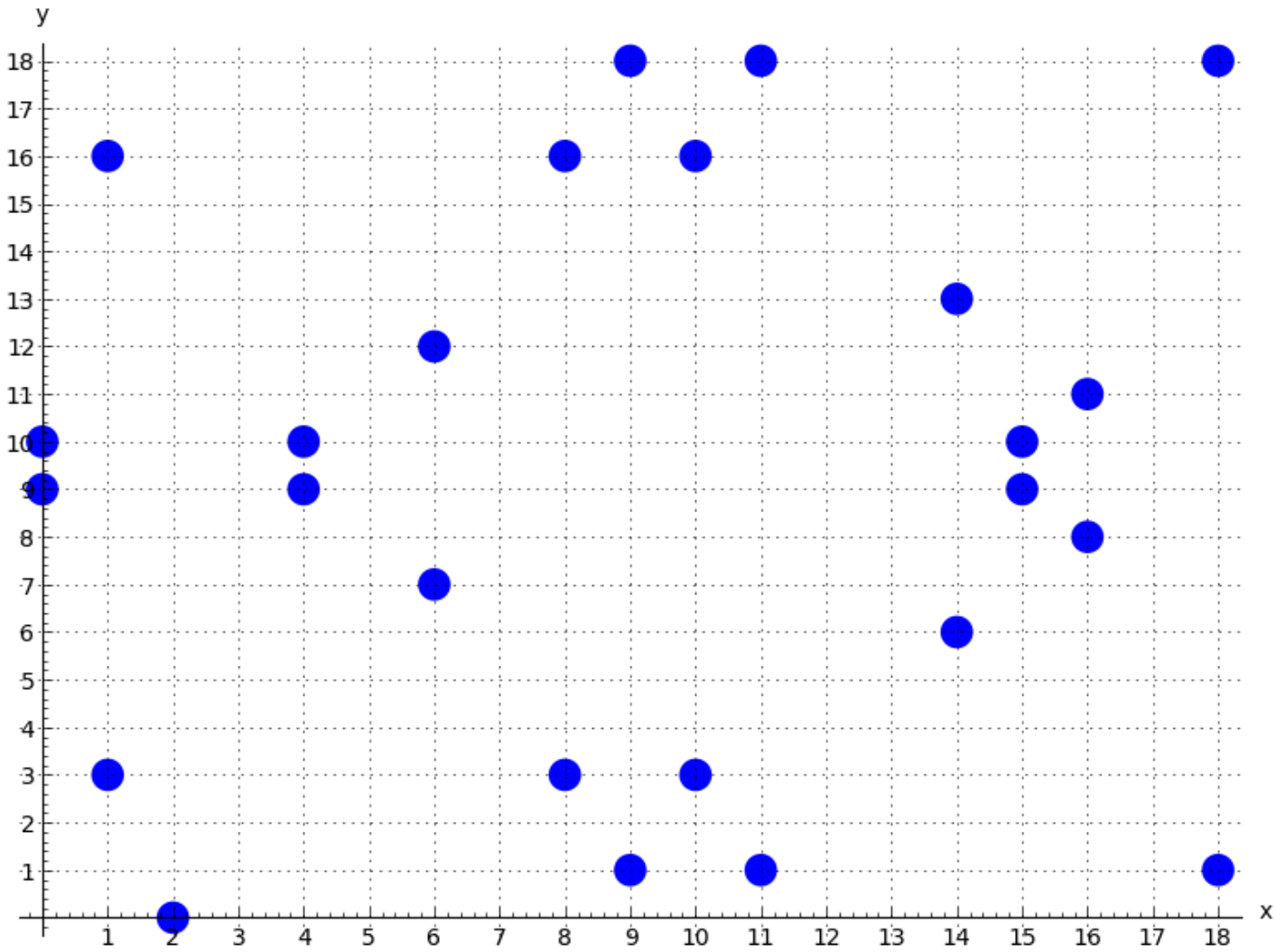
Beispiel: $y^2 = x^3 + 3x + 5$

p	5	7	11	13	17	19	23	31	37	41	43	47
N_p	10	7	9	9	23	26	28	38	48	44	46	61

Für $p = 2,\ 3,\ 29$ ist E singulär über $GF(p)$.

Elliptische Kurven über GF(p)

Beispiel $E : y^2 = x^3 + 3x + 5$ über $GF(19)$: $E(GF(19)) \cong \mathbb{Z}/26\mathbb{Z}$.



Kryptographie mit Gruppen

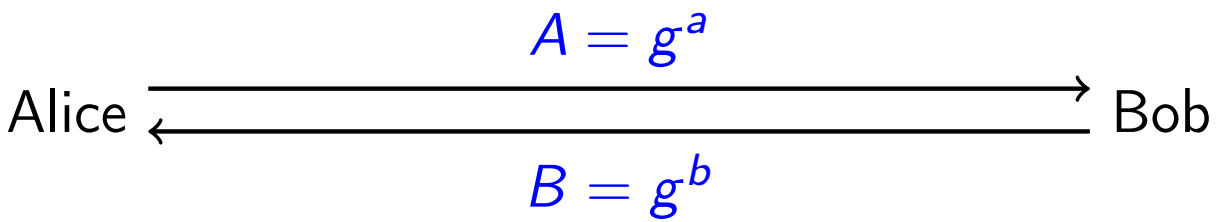
Gesucht ist ein Gruppe G und ein Gruppenelement g so dass $A = g^a$ effizient berechnet werden kann, wenn a bekannt ist, aber die Gleichung $g^x = A$ (*der diskrete Logarithmus*) schwer lösbar ist. $f(x) = g^x$ ist dann eine *Einwegfunkion*.

Mögliche Gruppen:

- $(GF(p)^*, \cdot)$, die Restklassen modulo p mit Multiplikation; diese Gruppen werden seit Entstehung der Public-Key Kryptographie eingesetzt. Empfohlene Länge von p: 2048 Binärstellen.
- $(E(GF(p)), +)$, Punkte einer elliptischen Kurve über $GF(p)$ mit Punkteaddition. Seit einigen Jahren zunehmende Verwendung. Empfohlene Länge von p: 256 Binärstellen. Effizienzvorteil gegenüber $GF(p)^*$.

Diffie-Hellmann Verfahren

Die Gruppe G und ein Element $g \in G$ der Ordnung q werden vereinbart (öffentlich). Alice wählt einen geheimen Schlüssel a und Bob einen geheimen Schlüssel b, wobei a, b natürliche Zahlen zwischen 1 und q sind. Sie berechnen $A = g^a$ bzw. $B = g^b$ und tauschen die Werte über einen öffentlichen (unsicheren) Kanal aus.



Beide berechnen dann den gemeinsamen geheimen Schlüssel K:

$$K = A^b = B^a = g^{ab}$$

Ein Angreifer hat keine Chance, solange er nicht das *Diskrete Logarithmus Problem* lösen kann.

Kryptographie mit Elliptischen Kurven

Wähle Primzahl p, Parameter $A, B \in GF(p)$ so dass $E : y^2 = x^3 + Ax + B$, sowie einen Punkt $P \in E(GF(p))$ der Ordnung q. Die Parameter p, E, P und q sind öffentlich und müssen bestimmte Eigenschaften haben (*Elliptic Curve Domain Parameters*).

Beispiel für *Technische Eigenschaften:* $p \equiv 3 \mod 4$. Dann gilt $(y^2)^{(p+1)/4} = y^{(p-1)/2} \cdot y = \pm y$, so dass $\pm y$ berechnet werden kann und nicht gespeichert werden muss.

Beispiel für *Sicherheitseigenschaften:* q ist eine Primzahl mit ≥ 160 Binärstellen. Das Diskrete-Logarithmus Problem für zusammengesetzte Zahlen ist einfacher und kann für jede Primzahlpotenz separat gelöst werden (Chinesischer Restsatz).

Beispiel einer elliptischen Kurve (256 Bit)

brainpoolP256r1

p=A9FB57DBA1EEA9BC3E660A909D838D726E3BF623D526202820134 81D1F6E5377

A=7D5A0975FC2C3057EEF67530417AFFE7FB8055C126DC5C6CE94A4B44F330B5D9

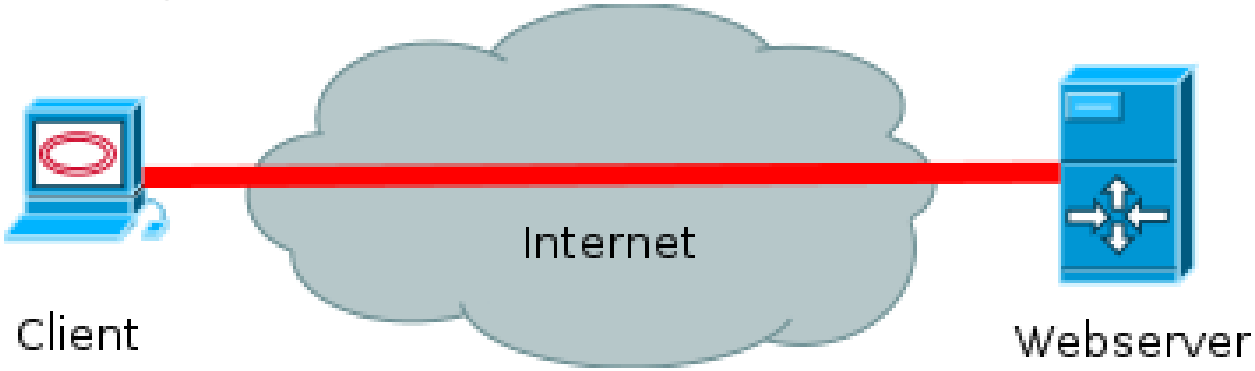
B=26DC5C6CE94A4B44F330B5D9BBD77CBF958416295CF7E1CE6BCCDC18FF8C07B6

P=(8BD2AEB9CB7E57CB2C4B482FFC81B7AFB9DE27E1E3BD23C23A4453BD9ACE3262,
547EF835C3DAC4FD97F8461A14611DC9C27745132DED8E545C1D54C72F046997)

q=A9FB57DBA1EEA9BC3E660A909D838D718C397AA3B561A6F7901E0E82974856A7

TLS Protokoll

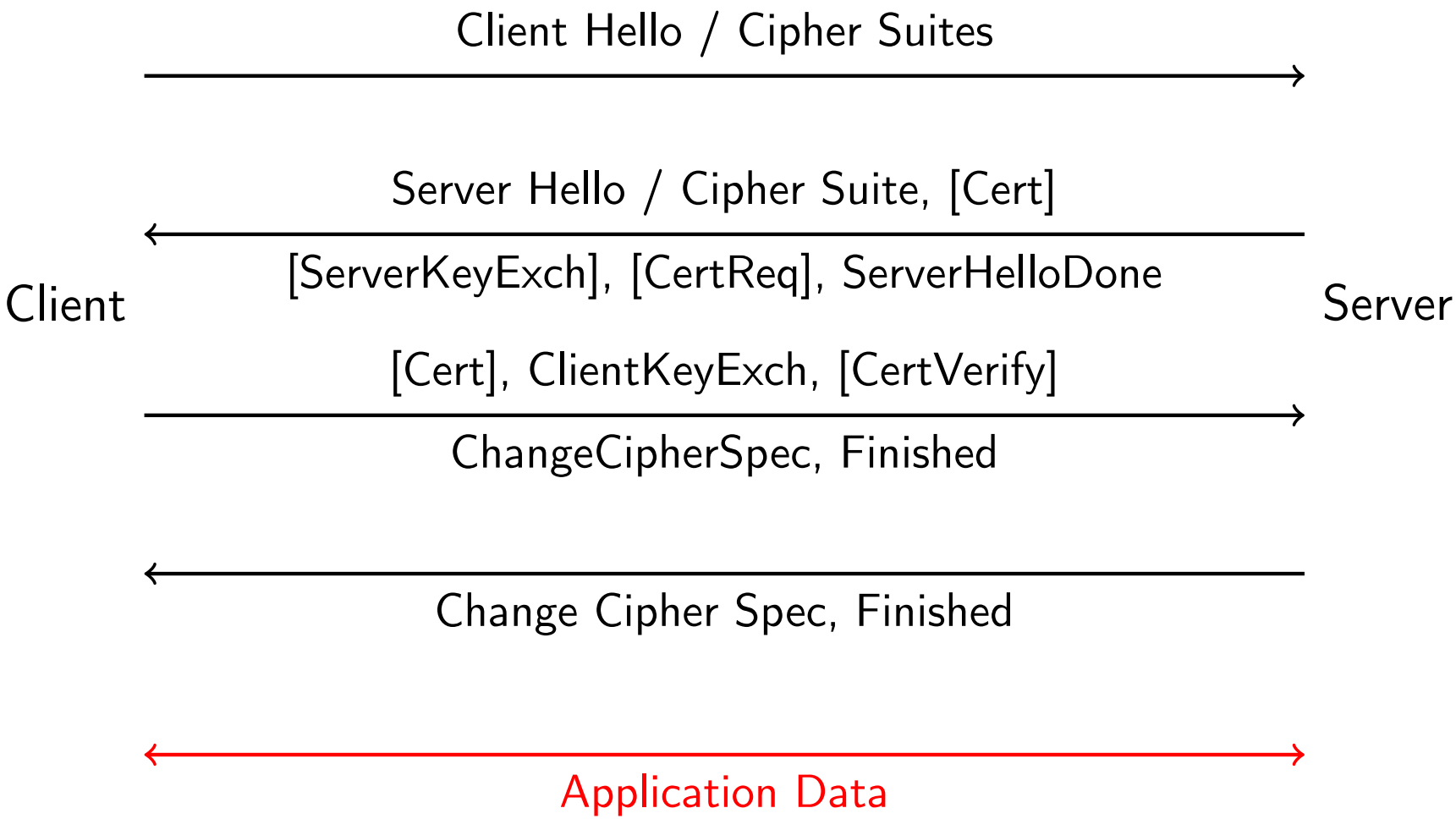
TLS (Transport Layer Security, früher SSL) zählt zu den wichtigsten Protokollen der Internet- und der Web-Sicherheit.



TLS beinhaltet:

- Handshake zur Aushandlung der Parameter, Authentifikation und Schlüsselvereinbarung
- Verschlüsselung der Anwendungsdaten und Sicherung der Nachrichtenintegrität zwischen Client und Server (Ende-zu-Ende Sicherheit)

TLS Handshake



TLS und Elliptische Kurven Kryptographie (ECC)

Beispiele für Cipher Suites:

- `DHE-RSA-AES128-SHA`: keine ECC
- `ECDHE-RSA-AES256-SHA384`: Diffie-Hellmann mit ECC
- `ECDHE-ECDSA-AES256-SHA`: Diffie-Hellmann und Server-Authentifikation mit ECC Signatur

Elliptic Curve Diffie-Hellmann Exchange (ECDHE) bei TLS

Client Hello: sendet Cipher Suites und `supported elliptic curves`, z.B. secp256r1 oder brainpoolP256r1 (beinhaltet p, E, P und q) und `ec_point_formats`

Server Hello: Auswahl der Cipher Suite.
Server Key Exchange: Auswahl der Kurve. Öffentlicher Diffie-Hellmann Schlüssel $aP \in E(GF(p))$ des Servers. Signatur der Parameter zur Authentifikation des Servers.

Client Key Exchange: Öffentlicher Diffie-Hellmann Schlüssel $bP \in E(GF(p))$ des Clients. Prüfung der Signatur des Servers und der Zertifikate.

Gemeinsames *Pre-Master-Secret*: $K = x(abP)$. Das Master-Secret und die weiteren TLS Sitzungsschlüssel werden von K mit Hilfe einer *pseudo random function* abgeleitet.

Beispiel Diffie-Hellmann mit ECC

brainpoolP256r1 mit Testvektoren aus RFC 7027

a=81DB1EE100150FF2EA338D708271BE38300CB54241D79950F77B063039804F1D

b=55E40BC41E37E3E2AD25C3C6654511FFA8474A91A0032087593852D3E7D76BD3

A=aP=(44106E913F92BC02A1705D9953A8414DB95E1AAA49E81D9E85F929A8E3100BE5,
8AB4846F11CACCB73CE49CBDD120F5A900A69FD32C272223F789EF10EB089BDC)

B=bP=(8D2D688C6CF93E1160AD04CC4429117DC2C41825E1E9FCA0ADDD34E6F1B39F7B,
990C57520812BE512641E47034832106BC7D3E8DD0E4C7F1136D7006547CEC6A)

abP=bA=aB=
(89AFC39D41D3B327814B80940B042590F96556EC91E6AE7939BCE31F3A18BF2B,
49C27868F4ECA2179BFD7D59B1E3BF34C1DBDE61AE12931648F43E59632504DE)

K=89AFC39D41D3B327814B80940B042590F96556EC91E6AE7939BCE31F3A18BF2B

Sicherheit von TLS mit Elliptischen Kurven

ECC gilt als sehr sicher: keine Angriffe bekannt mit einer Komplexität besser als $O(\sqrt{q})$; falls q also 160 oder mehr Binärstellen hat, so ist der Aufwand zu hoch für reale Angriffe !

Vergleich: für das Faktorisierungsproblem (bei RSA) und den diskreten Logarithmus in $GF(p)^*$ (bei Diffie-Hellmann) sind sub-exponentielle Angriffe bekannt. Daher sind deutlich mehr Binärstellen (ca. 2000) für ein ähnliches Sicherheitsniveau erforderlich.

Sicherheit und kryptographische Verfahren

Die kryptographischen Algorithmen sind nur ein Bestandteil der gesamten Sicherheit. Hinzu kommen Protokoll-Design, die Art der Verwendung, die Wahl und Aufbewahrung der Schlüssel, die Implementierung usw.

Beispiele: a) Schwächen des Elliptische-Kurven Pseudo Zufallszahlen-Generators Dual_EC_DRBG.

b) Bei TLS sind Angriffe (insbesondere *Man-in-the-Middle*) gegen die Authentifikation des Servers möglich, falls die Zertifikate nicht zuverlässig ausgestellt wurden oder vom Browser nicht ausreichend geprüft werden !

Alice vereinbart $K = g^{ab}$ mit Mallory statt mit dem Server !
Mallory und der Server verwenden den Schlüssel $K' = g^{a'b'}$.

Zusammenfassung

Der Beitrag behandelt elliptische Kurven, die Bezüge zu verschiedenen Teilgebieten der Mathematik besitzen. Für kryptographische Anwendungen kann man die Gruppe der Punkte elliptischer Kurven über endlichen Körpern verwenden. Inzwischen gehört die Kryptographie mit elliptischen Kurven zu den Standardverfahren, die beispielsweise zur Schlüsselvereinbarung eingesetzt werden. Dabei bieten elliptische Kurven bestimmte Effizienzvorteile gegenüber anderen Verfahren. Der Einsatz elliptischer Kurven wird anhand des Internet Standards TLS (Transport Layer Security) erläutert. TLS ist eines der wichtigsten Protokolle zur Sicherung des Netzwerkverkehrs.

Literatur


Johannes Buchmann.
Einführung in die Kryptographie.
Springer, 2010.


Simon Blake-Wilson et al.
Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS).
Internet Request for Comment RFC 4492, 2006.


Johannes Merkle, Manfred Lochter
Elliptic Curve Cryptography (ECC) Brainpool Curves for Transport Layer Security (TLS).
Internet Request for Comment RFC 7027, 2013.


Joseph H. Silverman.
The Arithmetic of Elliptic Curves.
Springer, 2009.


Wade Trappe, Lawrance C. Washington.
Introduction to Cryptography with Coding Theory.
Pearson, 2005.