Einführung in die Kryptologie

Johannes Köbler

Institut für Informatik
Humboldt-Universität zu Berlin

SS 2022

Für das Verständnis der Public-Key Verfahren benötigen wir noch einige Hilfsmittel aus der Zahlentheorie

Definition. Sei G eine endliche Gruppe.

- Die Ordnung von G ist die Anzahl $|G|$ ihrer Elemente
- Die Ordnung eines Elements $a \in G$ ist $\mathrm{ord}_G(a) = \min\{n \geq 1 \mid a^n = 1\}$
- Ist $G = \mathbb{Z}_m^*$, so schreiben wir einfach $\mathrm{ord}_m(a)$ anstelle von $\mathrm{ord}_{\mathbb{Z}_m^*}(a)$
- Die von a in G erzeugte Untergruppe $\{a^0, \ldots, a^{\mathrm{ord}_G(a)-1}\}$ bezeichnen wir mit $\langle a \rangle_G$ oder mit $\langle a \rangle$, wenn G aus dem Kontext ersichtlich ist

- Sei $e \geq 1$ der kleinste Exponent mit $a^e = a^{e'}$ für ein $e' \in \{0, \ldots, e-1\}$
- Dann gilt $a^i \neq a^j$ für alle $0 \leq i < j < e$ und wegen $a^{e-e'} = a^e a^{-e'} = a^e a^{-e} = 1 = a^0$ muss $e - e' = e$, also $e' = 0$ sein
- Dies zeigt, dass $\mathrm{ord}_G(a) = e$ ist und die Menge $\{a^0, \ldots, a^{\mathrm{ord}_G(a)-1}\}$ tatsächlich eine Untergruppe von G der Ordnung $\mathrm{ord}_G(a)$ bildet

Die Ordnung von Gruppenelementen

- In den Übungen werden wir sehen, dass für beliebige ganze Zahlen $i, j \in \mathbb{Z}$ folgende Äquivalenz gilt:

$$a^i = a^j \;\Leftrightarrow\; i \equiv_{\mathrm{ord}(a)} j$$

- Da $\mathrm{ord}(a) = |\langle a \rangle|$ die Ordnung einer Untergruppe von G ist, muss $\mathrm{ord}(a)$ ein Teiler der Gruppenordnung $|G|$ sein (Satz von Lagrange)

Beispiel

Folgende Tabelle zeigt für jedes Element a der Gruppe $G = \mathbb{Z}_7^*$ die von a erzeugte Untergruppe $\langle a \rangle$ sowie seine Ordnung $\mathrm{ord}_G(a) = |\langle a \rangle|$:

a	1	2	3	4	5	6
$\langle a \rangle$	$\{1\}$	$\{1,2,4\}$	$\{1,3,2,6,4,5\}$	$\{1,4,2\}$	$\{1,5,4,6,2,3\}$	$\{1,6\}$
$\mathrm{ord}_G(a)$	1	3	6	3	6	2

◁

Die Ordnung von Gruppenelementen

Satz (Euler-Fermat)

In jeder Gruppe $(G, \cdot, 1)$ der Ordnung $|G| = m$ gilt $a^m = 1$ für alle $a \in G$

Beweis.

- Wir betrachten hier nur den Fall, dass G kommutativ ist, der allgemeine Fall wird in den Übungen bewiesen
- Sei also $G = \{b_1, \ldots, b_m\}$ abelsch und sei $a \in G$ beliebig
- Wegen $ab_i \neq ab_j$ für $i \neq j$ folgt $G = \{ab_1, \ldots, ab_m\}$
- Dies impliziert $\prod_{i=1}^m b_i = \prod_{i=1}^m ab_i = a^m \prod_{i=1}^m b_i$
- Also muss $a^m = 1$ sein $\square$

Angewandt auf die Gruppe $\mathbb{Z}_p^*$ erhalten wir den kleinen Satz von Fermat

Korollar (Kleiner Satz von Fermat)

Für jede Primzahl p und jede Zahl a mit $a \not\equiv_p 0$ gilt $a^{p-1} \equiv_p 1$

Diskrete Logarithmen

- Für ein beliebiges Gruppenelement $a \in G$ ist die Exponentialfunktion $\exp_{G,a} : x \mapsto a^x$ mit der Basis a eine Bijektion zwischen der Menge $\mathbb{Z}_{ord(a)} = \{0, 1, \ldots, ord(a) - 1\}$ und der von a erzeugten Untergruppe $\langle a \rangle = \{a^0, \ldots, a^{\mathrm{ord}_G(a)-1}\}$
- Die zugehörige Umkehrabbildung spielt in der Kryptografie eine wichtige Rolle

Definition. Sei G eine Gruppe und seien $a, b \in G$ mit $b \in \langle a \rangle$.

- Der eindeutig bestimmte Exponent $x \in \mathbb{Z}_{\mathrm{ord}(a)}$ mit $a^x = b$ heißt Index oder diskreter Logarithmus von b zur Basis a in G, kurz

$$x = \log_{G,a}(b)$$

- Im Fall $G = \mathbb{Z}_m^*$ schreiben wir auch einfach $\log_{m,a}(b)$ anstelle von $\log_{\mathbb{Z}_m^*,a}(b)$

Diskrete Logarithmen

- Die Funktion $\exp_{m,a} : x \mapsto a^x$ ist effizient berechenbar (siehe unten)
- Dagegen sind bis heute keine effizienten Verfahren zur Berechnung von $\log_{m,a}(b)$ bekannt (falls a und m geeignet gewählt werden)

Beispiel

- Das Element $a = 2$ hat in der Gruppe $G = \mathbb{Z}_{11}^*$ die maximal mögliche Ordnung $\mathrm{ord}_{11}(2) = |G| = 10$
- Die folgenden Tabellen zeigen den Werteverlauf der Funktionen $\exp_{11,2}$ und $\log_{11,2}$

x	0	1	2	3	4	5	6	7	8	9
2^x	1	2	4	8	5	10	9	7	3	6

b	1	2	3	4	5	6	7	8	9	10
$\log_{11,2}(b)$	0	1	8	2	4	9	7	3	6	5

◁

Zyklische Gruppen

Für manche Anwendungen sind Elemente $a \in G$ nützlich, mit denen sich die gesamte Gruppe erzeugen lässt

Definition

- Sei G eine endliche Gruppe der Ordnung $|G| = m$
- Ein Element $g \in G$ mit $\mathrm{ord}_G(g) = m$ heißt Erzeuger von G
- G heißt zyklisch, falls G mindestens einen Erzeuger besitzt

Ein Element $a \in G$ ist also genau dann ein Erzeuger, wenn die von a erzeugte Untergruppe $\langle a \rangle$ die gesamte Gruppe G umfasst

Satz (Gauß)

Genau für $m \in \{1, 2, 4, p^n, 2p^n \mid p \geq 3 \text{ prim}, n \geq 1\}$ ist die Gruppe $\mathbb{Z}_m^*$ zyklisch (ohne Beweis)

Lemma (Euler)

Für alle $m \geq 1$ gilt

$$\sum_{d|m} \varphi(d) = m,$$

wobei die Summe über alle Teiler $d \geq 1$ von m läuft

Beweis.

- Für jeden Teiler $d \geq 1$ von m sei $T_d := \{a \in \mathbb{Z}_m \mid \mathrm{ggT}(a, m) = d\}$
- Dann folgt wegen

$$\varphi(m/d) = |\{\underbrace{b \in \mathbb{Z}_{m/d}}_{\Leftrightarrow\, bd \in \mathbb{Z}_m} | \underbrace{\mathrm{ggT}(b, m/d) = 1}_{\Leftrightarrow\, \mathrm{ggT}(bd, m) = d}\}| = |T_d|$$

die Gleichheit

$$\sum_{d|m} \varphi(d) = \sum_{d|m} \varphi(m/d) = \sum_{d|m} |T_d| = |\mathbb{Z}_m| = m$$

$\square$

Zyklische Gruppen

Satz. Sei G eine Gruppe der Ordnung $|G| = m$.

- G ist genau dann zyklisch, wenn jede Gleichung der Form $x^n = 1$ $(1 \leq n \leq m)$ höchstens n verschiedene Lösungen $a \in G$ hat
- In diesem Fall hat G genau $\varphi(m)$ Erzeuger

Beweis ($\Rightarrow$).

- Falls G zyklisch und a ein Erzeuger von G ist, so ist $G = \{a^k \mid k \in \mathbb{Z}_m\}$
- Die Potenz a^k ist genau dann Lösung von $x^n = 1$, wenn $a^{kn} = 1$ ist
- Sei $g = \mathrm{ggT}(n, m)$ und seien $n' = n/g$ sowie $m' = m/g$
- Wegen $\mathrm{ggT}(n', m') = 1$ folgt dann mit dem Lemma von Euklid

$$a^{kn} = 1 \Leftrightarrow kn \equiv_m 0 \Leftrightarrow kn' \equiv_{m'} 0 \overset{\text{Euklid}}{\Leftrightarrow} k \equiv_{m'} 0$$

- Die Gleichung $x^n = 1$ hat also genau $g \leq n$ Lösungen $a^0, a^{m'}, \ldots, a^{(g-1)m'}$

Zyklische Gruppen

Satz. Sei G eine Gruppe der Ordnung $|G| = m$.

- G ist genau dann zyklisch, wenn jede Gleichung der Form $x^n = 1$ $(1 \leq n \leq m)$ höchstens n verschiedene Lösungen $a \in G$ hat
- In diesem Fall hat G genau $\varphi(m)$ Erzeuger

Beweis ($\Leftarrow$).

- Für die Rückrichtung betrachten für $d \leq m$ die beiden Mengen
  - $Ord_d = \{a \in G \mid \mathrm{ord}(a) = d\}$ und
  - $Sol_d = \{a \in G \mid a^d = 1\}$
- Wegen $Ord_d \subseteq Sol_d$ gilt nach Voraussetzung $|Ord_d| \leq |Sol_d| \leq d$
- Wir zeigen $|Sol_d| = d$ und $|Ord_d| = \varphi(d)$ für jeden Teiler d von m
- Da für jedes $a \in Ord_d$ die Untergruppe $\langle a \rangle$ die Größe d hat, folgt nach Euler-Fermat die Inklusion $\langle a \rangle \subseteq Sol_d$
- Wegen $d = |\langle a \rangle| \leq |Sol_d| \leq d$ folgt daher $Ord_d \subseteq Sol_d = \langle a \rangle$

Zyklische Gruppen

Beweis (Fortsetzung).

- Wegen $d = |\langle a \rangle| \le |Sol_d| \le d$ folgt daher $Ord_d \subseteq Sol_d = \langle a \rangle$
- Zudem gilt $\mathrm{ord}(a^i) = d$ genau dann, wenn $\mathrm{ggT}(i, d) = 1$ ist (siehe Üb.)
- Daher folgt $Ord_d = \{a^i \mid i \in \mathbb{Z}_d^*\}$ für jedes $a \in Ord_d$ und somit $|Ord_d| \in \{0, \varphi(d)\}$ für alle d
- Mit obigem Lemma folgt nun

$$\sum_{d|m} |Ord_d| = |G| = m = \sum_{d|m} \varphi(d)$$

- Wegen $|Ord_d| \in \{0, \varphi(d)\}$ muss also $|Ord_d| = \varphi(d)$ für alle Teiler d von m und insbesondere $|Ord_m| = \varphi(m)$ sein $\square$

Für endliche Körper $\mathbb{F}_q$ erhalten wir eine interessante Folgerung:

- Da die Gleichung $x^n = 1$ in jedem Körper höchstens n Lösungen hat (siehe Übungen), ist die Einheitengruppe $\mathbb{F}_q^*$ von $\mathbb{F}_q$ zyklisch und hat genau $\varphi(q-1)$ Erzeuger (insbesondere ist auch $\mathbb{Z}_p^*$, p prim, zyklisch)

Algorithmische Verifikation von Erzeugern

Sofern die Primfaktorzerlegung der Gruppenordnung m bekannt ist, lässt sich effizient überprüfen, ob ein Gruppenelement $a \in G$ ein Erzeuger ist

Satz

Ein Element a einer endlichen Gruppe G der Ordnung $|G| = m$ ist genau dann ein Erzeuger, wenn für jeden Primteiler p von m gilt:

$$a^{m/p} \neq 1$$

Beweis.

- Für einen Erzeuger a von G gilt $a^e \neq 1$ für alle $e \in \{1, \ldots, m-1\}$ und somit auch für alle Exponenten e der Form m/p, p prim
- Ist dagegen $\mathrm{ord}(a) < m$, so ist $\mathrm{ord}(a)$ ein echter Teiler von m und daher existiert eine Zahl $d \geq 2$ mit $d \cdot \mathrm{ord}(a) = m$
- Folglich gilt für einen beliebigen Primteiler p von d

$$a^{m/p} = a^{d\,\mathrm{ord}(a)/p} = (a^{\mathrm{ord}(a)})^{d/p} = 1$$

□

Berechnung von Erzeugern

Der folgende probabilistische Algorithmus berechnet einen Erzeuger a in einer zyklischen Gruppe G, falls sich die Elemente von G zufällig generieren lassen und alle Primteiler p von $m = |G|$ bekannt sind

ComputeGenerator$(G, p_1, \ldots, p_k)$

input zyklische Gruppe G und alle Primteiler $p_1, \ldots, p_k$ von $m = |G|$
repeat
  guess randomly $a \in G$
until $a^{m/p_i} \neq 1$ für alle $i \in \{1, \ldots, k\}$
output a

Eigenschaften von ComputeGenerator:

- Da $\varphi(m) \geq m/(2 \ln\ln m)$ für hinreichend große m gilt, findet der Algorithmus in jedem Schleifendurchlauf mit Wahrscheinlichkeit $\varphi(m)/m \geq 1/(2 \ln\ln m)$ einen Erzeuger
- Die erwartete Anzahl der Schleifendurchläufe ist also $O(\ln\ln m)$

Wiederholtes Quadrieren und Multiplizieren

- Falls in einer Halbgruppe oder einem Ring das Produkt zweier Elemente effizient berechenbar ist, sind auch die Potenzen a^e effizient berechenbar
- Hierzu sind maximal $2\lceil \log e \rceil$ Multiplikationen erforderlich
- Sei $(e_r \dots e_0)_2$ mit $r = \lfloor \log_2 e \rfloor$ die Binärdarstellung von $e = \sum_{i=0}^r e_i \cdot 2^i$
- Dann können wir den Exponenten e sukzessive mittels $b_0 = e_0$ und $b_i = b_{i-1} + e_i 2^i = \sum_{j=0}^i e_j \cdot 2^j$ für $i = 1, \dots, r$ zu $b_r = e$ berechnen
- Alternativ lässt sich e auch nach dem Horner-Schema berechnen:
  - Sei $c_r = e_r = 1$ und für $i = r-1, \dots, 0$ sei $c_i = 2c_{i+1} + e_i$
  - Dann ist $c_i = \sum_{j=i}^r e_j \cdot 2^{j-i}$, also $c_0 = \sum_{j=0}^r e_j \cdot 2^j = e$
- Pot berechnet die Potenzen $y_i = a^{b_i}$ und HornerPot die Potenzen $z_i = a^{c_i}$:

Pot(a, e, G)

```
x := a; y := a^{e_0}
for i := 1 to r do
  x := x^2; y := y · x^{e_i}
return(y)
```

HornerPot(a, e, G)

```
z := a
for i := r − 1 downto 0 do
  z := z^2 · a^{e_i}
return(z)
```

Wiederholtes Quadrieren und Multiplizieren

Beispiel

- Wir berechnen in in der Gruppe $\mathbb{Z}_m^*$ die Potenz a^e für $a = 1920$, $e = 19$ und $m = 2773$:

Pot(1920, 19, 2773)					HornerPot(1920, 19, 2773)			
i	e_i	b_i	$x_i = x_{i-1}^2 = a^{2^i}$	$y_i = y_{i-1} x_i^{e_i} = a^{b_i}$	i	e_i	c_i	$z_i = (z_{i+1})^2 a^{e_i} = a^{c_i}$
0	1	1	1920	$1920^1 = 1920$	4	1	1	$1920^1 = 1920$
1	1	3	$1920^2 = 1083$	$1920 \cdot 1083^1 = 2383$	3	0	2	$1920^2 \cdot 1920^0 = 1083$
2	0	3	$1083^2 = 2683$	$2383 \cdot 2683^0 = 2383$	2	0	4	$1083^2 \cdot 1920^0 = 2683$
3	0	3	$2683^2 = 2554$	$2383 \cdot 2554^0 = 2383$	1	1	9	$2683^2 \cdot 1920^1 = 1016$
4	1	19	$2554^2 = 820$	$2383 \cdot 820^1 = \mathbf{1868}$	0	1	19	$1016^2 \cdot 1920^1 = \mathbf{1868}$

- Es gilt also $1920^{19} \equiv_{2773} 1868$

◁

- Bezeichne $\mathcal{P}$ die Menge aller Primzahlen und sei π die Funktion, die jeder Teilmenge $A \subseteq \mathbb{N}$ die Anzahl $\pi(A) = |\mathcal{P} \cap A|$ der Primzahlen in der Menge A zuweist
- Die Menge $\mathcal{P} \cap [n]$ bezeichnen wir auch einfach mit $\mathcal{P}_n$ und die Zahl $\pi([n])$ mit $\pi(n)$ sowie $\pi([a,b])$ mit $\pi(a,b)$
- Für $c \in \mathbb{Z}_m$ sei $\pi_{c,m}(n) := |\{p \in \mathcal{P}_n \mid p \equiv_m c\}|$

Satz (Hadamard und de la Vallée Poussin, 1896)

- Es gilt $\pi(n) \sim n/\ln n$ und für $c \in \mathbb{Z}_m^*$ gilt $\pi_{c,m}(n) \sim n/(\varphi(m)\ln n)$
- Hierbei bedeutet $f(n) \sim g(n)$, dass die beiden Funktionen f und g asymptotisch äquivalent sind (d.h. es gilt $\lim_{n\to\infty} f(n)/g(n) = 1$)

n	$\pi(n)$	$\pi(n) - n/\ln n$	$Li(n) - \pi(n)$
10	4	−0.3	2.2
100	25	3.3	5.1
1 000	168	23	10
10 000	1 229	143	17
10 100	1 240	144	18
10^6	78 498	6 116	130
10^9	50 847 534	2 592 592	1 701
10^{12}	37 607 912 018	1 416 705 193	38 263
10^{15}	29 844 570 422 669	891 604 962 452	1 052 619
10^{18}	24 739 954 287 740 860	612 483 070 893 536	21 949 555
10^{21}	21 127 269 486 018 731 928	446 579 871 578 168 707	597 394 254

- Wie obige Tabelle zeigt, liefert die Funktion $Li(n) = \int_2^n (\ln x)^{-1} dx$ im Vergleich zu $n/\ln n$ eine deutlich bessere Abschätzung von $\pi(n)$
- Verwenden wir die Abschätzung $\pi(n) \approx \int_2^n (\ln x)^{-1} dx$, so ergibt sich für die Anzahl $\pi(a,b)$ der Primzahlen im Intervall $[a,b]$ der Näherungswert

$$\pi(a,b) \approx \int_a^b (\ln x)^{-1} dx \geq (b-a)/\ln b$$

Der Primzahlsatz

Beispiel

- Für das Intervall $[a, b] = [10\,000, 10\,100]$ ergibt sich z. B. der Wert

  $$\pi(a,b) \approx \int_a^b (\ln x)^{-1} dx \geq 100/\ln 10\,100 \approx 10{,}85$$

  während der exakte Wert $\pi(10\,100) - \pi(10\,000) = 11$ ist
- Für die Anzahl aller 100-stelligen Primzahlen (in Dezimaldarstellung) im Intervall $[a, b] = [10^{99}, 10^{100}]$ erhalten wir den Näherungswert

  $$\int_a^b (\ln x)^{-1} dx \geq 9 * 10^{99}/\ln 10^{100} = 9 * 10^{97}/\ln 10 \approx 3{,}91 \cdot 10^{97}$$
- Vergleichen wir diese Zahl mit der Anzahl $10^{100} - 10^{99} = 9 \cdot 10^{99}$ aller 100-stelligen Dezimalzahlen, so sehen wir, dass ungefähr jede $900/3{,}91 \approx 230$-te 100-stellige Dezimalzahl prim ist
- Für die Anzahl aller 1000-stelligen Primzahlen im Intervall $[a, b] = [10^{999}, 10^{1000}]$ erhalten wir dagegen den Näherungswert

  $$\int_a^b (\ln x)^{-1} dx \geq 9 * 10^{999}/\ln 10^{1000} = 9 * 10^{996}/\ln 10 \approx 3{,}91 \cdot 10^{996}$$
- Hier sehen wir, dass ungefähr jede $9000/3{,}91 \approx 2303$-te der $9 \cdot 10^{999}$ 1000-stelligen Dezimalzahlen prim ist $\triangleleft$