Endliche Körper

Sind sehr wichtig in der Kryptographie und Codierungstheorie (Übertragungsfehlerkorrektur). Nützlich bei der Beschreibung von Rijndael.

Ein Körper ist eine Menge, in der man wie in $\mathbb{R}$ oder $\mathbb{C}$ rechnen kann, also mit $+, -, \cdot, /$.

Endliche Körper haben nur endlich viele Elemente.

p Primzahl. Modulo p rechnen liefert $\mathbb{Z}/(p)$. Ist endlicher Körper $\mathbb{F}_p$.

- Elemente dargestellt durch $\{0, 1, \ldots, p-1\}$.
- Beispiel $p = 5$: $-1 = 4$ weil $4 + 1 = 5 = 0$ in $\mathbb{F}_p$.
- Invertieren mit euklidischem Algorithmus zur ggT-Berechnung: $1 = ra + sp$ impliziert $r = 1/a$. Beispiel $1 = -3 \cdot 3 + 2 \cdot 5$, also $-3 = 1/3$.

Polynome

Polynome über (endlichen) Körpern K.

- Sind Ausdrücke der Form $a_0 + a_1 x + a_2 x^2 + \cdots + a_r x^r$ mit Koeffizienten $a_i \in K$ und x einer „Variablen“.
- Gleichheit zweier Polynome genau dann, wenn die Koeffizienten vor x^i gleich sind für alle i.
- $+, -, \cdot$ wie gewohnt und sinnvoll.
- Grad $\deg(f) := r$ wenn $f = a_0 + a_1 x + a_2 x^2 + \cdots + a_r x^r$ und $a_r \neq 0$.
- $\deg(f) \leq 0$: f heißt konstant.

Beispiel:

- $(x+1)(x-1) = x^2 + x - x - 1 = x^2 - 1$.
- $2x + 1 \neq 2x + 2$, $x + 2 \neq 2x + 2$, $x \neq 0$.
- $\deg(x) = 1$, $\deg(x^2) = 2$, $\deg(1) = 0$, $\deg(0) = -\infty$.

Polynome

Menge der Polynome über K ist damit ein Ring (Körper ohne $/$). Wird mit $K[x]$ bezeichnet.

Division mit Rest: Für $f, g \in K[x]$ schreibe $f = hg + r$ mit $h, r \in K[x]$ und $\deg(r) < \deg(g)$.

- Wie in der Schule möglich, ist analog zur Division mit Rest in $\mathbb{Z}$.
- h und r sind eindeutig bestimmt.
- f durch g teilbar $\Leftrightarrow r = 0$.
- Können damit in $K[x]$ modulo g rechnen, Ergebnisse sind die r.
- $f = x^2 + 1$, $g = x - 1$: $x^2 + 1 = x^2 - 1 + 2 = (x+1)(x-1) + 2$, also $h = x + 1$ und $r = 2$.

Polynome

Primpolynom $f \in K[x]$: Kann nicht als Produkt nicht konstanter Polynome geschrieben werden (analog zur Primzahl).

- $x^2 - 1$ nicht prim: $x^2 - 1 = (x-1)(x+1)$.
- $x^2 + x + 1 \in \mathbb{F}_2[x]$ prim.

Thm: Jedes Polynom kann in ein Produkt von Primpolynomen zerlegt werden. Die Zerlegung ist bis auf Multiplikation mit Konstanten eindeutig.

- Nicht eindeutig: $x^2 - 1 = (x-1)(x+1) = (2x-2)(3x+3)$ in $\mathbb{F}_5[x]$.

Bew: Üblicherweise in der Algebra.

Endliche Körper

$K = \mathbb{F}_p$, $f \in K[x]$ Primpolynom. In $K[x]$ modulo f rechnen, ergibt $K[x]/(f)$. Ist endlicher Körper $\mathbb{F}_{p^n}$ mit p^n Elementen.

- Primpolynome beliebigen Grads gibt es in Tabellen.
- Invertieren wieder mit euklidischem Algorithmus, jetzt für Polynome mit Hilfe der Polynomdivision. Beispiel: $K = \mathbb{F}_2$, $f = x^2 + x + 1$, $x \cdot (x+1) + f = 1$. Also $x = 1/(x+1)$ in $\mathbb{F}_4$.
- In $K[x]/(f)$ gilt $f(x) = 0$.

Man schreibt häufig $\mathbb{F}_{p^n} = \mathbb{F}_p[\zeta]$ mit $f(\zeta) = 0$.

Elemente in $\mathbb{F}_{p^n}$ können also durch Polynome vom Grad $\leq n-1$ in ζ über $\mathbb{F}_p$ dargestellt werden. Es gilt $\#\mathbb{F}_{p^n} = p^n$.

Endliche Körper

$\mathbb{F}_{p^n}$ auch ein n-dimensionaler $\mathbb{F}_p$-Vektorraum.

Thm: Für jedes n ist $\mathbb{F}_{p^n}$ bis auf Isomorphie eindeutig bestimmt.

Thm: Es gibt $w \in \mathbb{F}_{p^n}$, so daß sich jedes $a \in \mathbb{F}_{p^n}$ mit $a \neq 0$ in der Form $a = w^s$ für ein s mit $0 \leq s \leq p^n - 1$ schreiben läßt.

Bew: Üblicherweise in der Algebra.

$(\mathbb{F}_2)^8$:

- Sind Bytes.
- Addition entspricht XOR.
- $-1 = 1$, daher Subtraktion gleich Addition.
- Identifikation mit $\mathbb{F}_{2^8}$ unter einer Basis $1, \zeta, \ldots, \zeta^7$.

AES - Rijndael

Basiert auf (verallgemeinerten) Substitutions-Permutationsnetzwerk.

- 128 Bit Blocklänge,
- 128/192/256 Bit Schlüssellänge,
- entsprechend 10/12/14 Runden.

Im folgenden $(\mathbb{F}_2)^8 \cong \mathbb{F}_2[\zeta] \cong \mathbb{F}_{2^8}$ als $\mathbb{F}_2$-Vektorräume mit $\zeta^8 + \zeta^4 + \zeta^3 + \zeta + 1 = 0$ und $(b_7, \ldots, b_0) \mapsto \sum_{i=0}^{7} b_i \zeta^i$.

Der jeweils zu bearbeitende Block m_i (state block) und Rundenschlüssel k_i im Netzwerk ist eine Matrix $\in (\mathbb{F}_{2^8})^{4\times 4}$. Hierbei spaltenweise arbeiten: Ein Block oder Rundenschlüssel $(b_0, \ldots, b_{15})$ ergibt die Spalten $(b_i, \ldots, b_{i+3})^t$ mit $i \in \{0, 4, 8, 12\}$.

Genaue Details im FIPS-197 und unter `http://www.nist.gov/aes`.

AES - Rijndael

Die Operationen in den einzelnen Runden sind:

AddRoundKey: Addition von k_i zu m_i, liefert m_{i+1}.

SubstBytes: Koeffweise Anwendung von $\pi_S \in S(\mathbb{F}_{2^8})$ auf m_i:

- ϕ ist eine affin-lineare Abbildung auf $\mathbb{F}_{2^8}$ als $\mathbb{F}_2$-Vektorraum, wird weiter unten definiert.
- Damit $\pi_S(x) := \phi(1/x)$ für $x \neq 0$ und $\pi_S(0) := 0$.

ShiftRows: Zeile j in m_i um $j-1$ Positionen nach links shiften.

MixColumns: Multiplikation von m_i mit $M = \begin{pmatrix} \zeta & \zeta+1 & 1 & 1 \\ 1 & \zeta & \zeta+1 & 1 \\ 1 & 1 & \zeta & \zeta+1 \\ \zeta+1 & 1 & 1 & \zeta \end{pmatrix}$

von links.

AES - Rijndael

Ausführung der Runden:

1. Eingabe $m_0 = m$.
2. AddRoundKey für $i = 0$.
3. Für $i = 1, \ldots, n-1$: SubstBytes, ShiftRows, MixColumns, AddRoundKey.
4. Für $i = n$: SubstBytes, ShiftRows, AddRoundKey.
5. Ausgabe $c = m_n$.

Für Entschlüsselung inverse Operationen benutzen.

Diffusion durch ShiftRows und MixColumns.
Konfusion durch SubstBytes und AddRoundKey.

AES - Rijndael

Blocklänge $b = 4, 6, 8$ des Schlüssel k in Worten (4 Bytes).

Schlüsselschema zur Bestimmung der k_i (KeyExpansion):

- Schreibe $k = (w_0, \ldots, w_{b-1})$ mit $w_j \in (\mathbb{F}_{2^8})^4$.
- Für $j \geq b$: $w_j \leftarrow w_{j-b} + f(j, w_{j-1})$, wobei f unten definiert ist.
- $k_i \leftarrow (w_{4i}, \ldots, w_{4i+3})$ für $0 \leq i \leq n$.

Definition von f:

- SubstWord: π_S auf $(\mathbb{F}_{2^8})^4$ erweitern durch byteweise Anwendung.
- RotWord: RotWord$(B_0, B_1, B_2, B_3) = (B_1, B_2, B_3, B_0)$, $B_i \in \mathbb{F}_{2^8}$.
- Für $j = 0 \bmod b$ ist $f(j, w_{j-1})$ = SubstWord(RotWord(w_{j-1})) $+ (\zeta^{j/b-1}, 0, 0, 0)$.
- Für $b > 6$ und $j = 4 \bmod b$ ist $f(j, w_{j-1})$ = SubstWord(w_{j-1}).
- In allen anderen Fällen ist $f(j, w_{j-1}) = w_{j-1}$.

AES - Rijndael

Definition von ϕ:

- $\mathbb{F}_{2^8} \cong (\mathbb{F}_2)^8 \cong \{f \in \mathbb{F}_2[x] \mid \deg(f) < 8\} \cong \mathbb{F}_2[x]/(x^8+1)$ als $\mathbb{F}_2$-Vektorräume, $\mathbb{F}_{2^8}$ mit Basis $1, \zeta, \ldots, \zeta^7$.
- $f \in \mathbb{F}_2[x]$, $\deg(f) < 8$: $\phi(f) := (x^4 + x^3 + x^2 + x + 1)f + x^6 + x^5 + x + 1 \bmod x^8 + 1$.
- $g \in \mathbb{F}_2[x]$, $\deg(g) < 8$: $\phi^{-1}(g) = (x^6 + x^3 + x)g + x^2 + 1 \bmod x^8 + 1$.

Da ϕ affin-linear und bijektiv ist, kann man ϕ auch mit Hilfe einer invertierbaren Matrix aus $(\mathbb{F}_2)^{8\times 8}$ definieren (siehe z.B. das FIPS-197 Dokument).

AES - Rijndael

Rijndael ist sehr schnell, besonders auf Chipkarten (im Vergleich doppelt so schnell wie andere Verfahren).

Sicherheit von Rijndael:

- Rundenzahl recht knapp gehalten.
- Wird in den nächsten Jahren intensiv untersucht werden.

Die Struktur von Rijndael ist sehr algebraisch. Daher gibt es eine einfache, geschlosse algebraische Formel für die Verschlüsselung.

- Manche sehen dies als möglichen Angriffspunkt an.
- Es könnte aber auch helfen, die Sicherheit zu beweisen.
- XL (extended linearisation): Verfahren zum Lösen von Gleichungssystemen mit vielen Unbekannten.
- Effizienz von XL?

Angriffe auf Blockchiffren

Brute-Force, exhaustive search (nach dem Schlüssel),
Meet-in-the-middle, CPA.

- Alles oder trickreich ausprobieren → effektive Schlüssellänge.

Differentielle Kryptoanalyse

- Man untersucht, wie Änderungen (Differenzen) am Klartext sich durch die Runden fortpflanzen, u. mit welcher Wahrscheinlichkeit.

Lineare Kryptoanalyse

- Man untersucht, mit welcher Wahrscheinlichkeit lineare Relationen zwischen Klartext- und Chiffretextbits gelten.

Diese Techniken sind statistisch und ziemlich detailliert.
Lassen sich (im wesentlichen) nicht auf DES und Rijndael anwenden.

Angriffe auf Blockchiffren

Timing Angriffe, Power Analysis:

- wenn Ausführungszeit oder Energieverbrauch vom Schlüssel abhängt.
- Zeit- bzw. Energiemessung liefert Information über den Schlüssel.
- Gegenmaßnahme: Algorithmus entsprechend modifizieren.

Differential Fault Analysis:

- Hardware-mäßiger Eingriff auf Bits und Programmausführung (z.B. in Chipkarte).
- Speziell Related-Key Angriff.
- Dann Untersuchung der geänderten/fehlerhaften Verschlüsselung.
- Starke Anforderung an die Hardware ...

Diese Angriffe sind auch für andere kryptographische Algorithmen relevant.