

Proseminar Kryptographie
Blockverschlüsselungen

Benjamin Richter

8. November 2004

Inhaltsverzeichnis

1	Blockchiffre	3
1.1	Shannons Konstruktionsprinzipien	3
1.2	Produkt-Chiffre	3
1.3	Iterierte Chiffren	4
1.4	Substitutions-Permutations-Netzwerke	4
1.4.1	Algorithmus	5
1.4.2	Beispiel	5
2	Kryptoanalyse	6
2.1	Überblick	6
2.2	Piling-Up Prinzip	6
2.2.1	Piling-Up Lemma(Matsui[2])	7
3	Quellen	8

1 Blockchiffre

- gängige Blockgrößen von 64 oder 128 Bits
- spezieller Type eines symmetrischen Kryptosystems
- Einteilung in Gruppen von Bits fester Länge
- generell: Blöcke möglichst groß , zwecks Kryptoanalyse

1.1 Shannons Konstruktionsprinzipien

- **Diffusion** (Durchmischung): "*avalanche effect*"
- Geheimtextbits hängen von Klartextbits ab
- Änderung **eines** Klartextbits ändert ca. **50%** der Geheimtextbits
- Grundbausteine sind Transpositionen

- **Konfusion** (Komplexität des Zusammenhangs):
- komplizierte Beziehung zwischen Klartext- und Geheimtextblock
- insbesondere hochgradig **nichtlinear**
- komplexe Abhängigkeit des Geheimtextblocks vom Schlüssel
- Grundbausteine sind Substitutionen

1.2 Produkt-Chiffre

Chiffre benutzt zwei endomorphe Kryptosysteme

- $\mathcal{P} = \mathcal{C}$
- $\mathcal{S}_1 = (\mathcal{P}, \mathcal{C}, \mathcal{K}_1, \mathcal{E}_1, \mathcal{D}_1)$
- $\mathcal{S}_2 = (\mathcal{P}, \mathcal{C}, \mathcal{K}_2, \mathcal{E}_2, \mathcal{D}_2)$

$$\mathcal{S}_1 \times \mathcal{S}_2 = (\mathcal{P}, \mathcal{C}, \mathcal{K}_1 \times \mathcal{K}_2, \mathcal{E}, \mathcal{D})$$

Ver- und Entschlüsselungsfunktion:

$$\begin{aligned} e_{k_1, k_2}(x) &= e_{k_2}(e_{k_1}(x)) \\ d_{k_1, k_2}(y) &= d_{k_1}(d_{k_2}(y)) \end{aligned}$$

1.3 Iterierte Chiffren

- Mengen

$$\mathcal{P} = \mathcal{C} = \{0, 1\}^n, \mathcal{K} = 0, 1^m$$

- Anzahl der Runden

$$(Nr)$$

- Rundenfunktion

$g : \{0, 1\}^n \times \{0, 1\}^l \rightarrow \{0, 1\}^n$ mit der Eigenschaft, dass $\forall z \in \{0, 1\}^l$ die Funktion

$g_z : \{0, 1\}^n \rightarrow \{0, 1\}^n, x \mapsto g(x, z)$ invertierbar ist

- Schlüsselplan

erzeugt aus $k \in \mathcal{K}$ Rundenschlüssel: $k^1, k^2, \dots, k^{Nr} \in \{0, 1\}^l$

1.4 Substitutions-Permutations-Netzwerke

- Anzahl der Runden

$$(Nr)$$

- Schlüsselplan

$$\mathcal{K} = (\mathcal{K}^1, \dots, \mathcal{K}^{Nr+1}), \mathcal{K} \in \{0, 1\}^l$$

- Mengen

$$\mathcal{P} = \mathcal{C} = (\{0, 1\}^l)^m, = \{0, 1\}^{lm}$$

- bijektive Substitutionsfunktion

$$\pi_S : \{0, 1\}^l \rightarrow \{0, 1\}^l$$

- Permutation

$$\pi_P : \{1, \dots, lm\}^l \rightarrow \{1, \dots, lm\}^l$$

Sei $x = (x_1, \dots, x_{lm}) \in \{0, 1\}^{lm}$ gegeben, dann bezeichnet $x_{(i+1)} = (x_{il+1}, \dots, x_{il+l})$ den i . Block der Länge l ($i = 1 \dots m$)

1.4.1 Algorithmus

$w^0 \leftarrow x$

for $r \leftarrow 1$ **to** $Nr - 1$

$$\mathbf{do} \left\{ \begin{array}{l} u^r \leftarrow w^{r-1} \oplus K^r \\ \mathbf{for} \ i \leftarrow 1 \ \mathbf{to} \ m \\ \quad \mathbf{do} \ v_{(i)}^r \leftarrow \pi_S(u_{(i)}^r) \\ w^r \leftarrow (v_{\pi_P(1)}^r, \dots, v_{\pi_P(lm)}^r) \end{array} \right.$$

$u^{Nr} \leftarrow w^{Nr-1} \oplus K^{Nr}$

for $i \leftarrow 1$ **to** m

do $v_{(i)}^{Nr} \leftarrow \pi_S(u_{(i)}^{Nr})$

$y \leftarrow v^{Nr} \oplus K^{Nr+1}$

output (y)

1.4.2 Beispiel

geg.: $l = m = 4$

$$\pi_S \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|} \hline 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & a & b & c & d & e & f \\ \hline e & 4 & d & 1 & 2 & f & b & 8 & 3 & a & 6 & c & 5 & 9 & 0 & 7 \\ \hline \end{array}$$

$$\pi_P \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|} \hline 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\ \hline 1 & 5 & 9 & 13 & 2 & 6 & 10 & 14 & 3 & 7 & 11 & 15 & 4 & 8 & 12 & 16 \\ \hline \end{array}$$

$\mathcal{K} = \underbrace{0011\ 1010\ 1001\ 0100}_{\mathcal{K}_1} \ 1101\ 0110\ 0011\ 1111$ mit 16 Bit pro Rundenschlüssel
für

$\mathcal{K}_2 = \text{shift}(\mathcal{K}_1)$ um 4 Bit nach rechts ...

$x = w^0 = 0010\ 0110\ 1011\ 0111$

$u^1 = 0001\ 1100\ 0010\ 0011$

$v^1 = 0100\ 1100\ 0010\ 0011$

$w^1 = 0010\ 1110\ 0000\ 0111$

⋮

2 Kryptoanalyse

2.1 Überblick

- Grundlage für Known-Plaintext-Attack
- $X_{i_1} \oplus X_{i_2} \oplus \dots \oplus X_{i_u} \oplus Y_{j_1} \oplus Y_{j_2} \oplus \dots \oplus Y_{j_v} = 0$
- Gleichungen gelten mit Wahrscheinlichkeit zwischen Quelltext-Bits und Zustands-Bits nach der vorletzten Runde
- durch lineare Beziehung, werden mögliche Schlüsselkandidaten für die letzte Runde bewertet

2.2 Piling-Up Prinzip

X_1 und X_2 sind zufällig gewählte Variablen

Beziehungen:

$$\begin{aligned} X_1 \oplus X_2 &= 0 \\ X_1 &= X_2 \\ X_1 \oplus X_2 &= 1 \\ X_1 &\neq X_2 \end{aligned}$$

Wahrscheinlichkeit:

$$\Pr(X_1 = i) = \begin{cases} p_1, & i = 0 \\ 1 - p_1, & i = 1 \end{cases}$$
$$\Pr(X_2 = i) = \begin{cases} p_2, & i = 0 \\ 1 - p_2, & i = 1 \end{cases}$$

sind die Variablen voneinander **unabhängig**:

$$\Pr(X_1 = i, X_2 = j) = \begin{cases} p_1 p_2 & , i = 0, j = 0 \\ p_1 (1 - p_2) & , i = 0, j = 1 \\ (1 - p_1) p_2 & , i = 1, j = 0 \\ (1 - p_1) (1 - p_2) & , i = 1, j = 1 \end{cases}$$

für $\Pr(X_1 \oplus X_2 = 0)$ folgt:

$$\begin{aligned} &= \Pr(X_1 = X_2) \\ &= \Pr(X_1 = 0, X_2 = 0) + \Pr(X_1 = 1, X_2 = 1) \\ &= p_1 p_2 + (1 - p_1) (1 - p_2) \\ &= p_1 p_2 + (1 - p_1 - p_2 + p_1 p_2) \\ &= 2p_1 p_2 - p_1 - p_2 + 1 \end{aligned}$$

setze für $p_1 = 1/2 + \epsilon_1$ bzw. $p_2 = 1/2 + \epsilon_2$ mit ϵ_1, ϵ_2 gleich "probability bias/deviation" von $-1/2 \leq \epsilon_1, \epsilon_2 \leq +1/2$ für $\Pr(X_1 \oplus X_2 = 0)$

$$\begin{aligned} &= 2(1/2 + \epsilon_1)(1/2 + \epsilon_2) - (1/2 + \epsilon_1) - (1/2 + \epsilon_2) + 1 \\ &= 1/2 + \epsilon_1 + \epsilon_2 + 2\epsilon_1\epsilon_2 - 1/2 - \epsilon_2 + 1 \\ &= 1/2 + 2\epsilon_1\epsilon_2 \end{aligned}$$

aus $\Pr(X_1 \oplus X_2 = 0) = 1/2 + 2\epsilon_1\epsilon_2$ folgt:

$$\epsilon_{1,2} = 2\epsilon_1\epsilon_2$$

Für $X_1 \dots X_n$ kann die Wahrscheinlichkeit $p_1 = 1/2 + \epsilon_1$ bis $p_n = 1/2 + \epsilon_n$ ausgerechnet werden. dabei gilt: $X_1 \oplus \dots \oplus X_n = 0$

2.2.1 Piling-Up Lemma(Matsui[2])

X_1, X_2, \dots, X_n sind unabhängige und zufällig gewählte Variablen

$$\Pr(X_1 \oplus \dots \oplus X_n = 0) = 1/2 + 2^{n-1} \prod_{i=1}^n \epsilon_i$$

ist äquivalent zu:

$$\epsilon_{1,2,\dots,n} = 2^{n-1} \prod_{i=1}^n \epsilon_i$$

Hinweis:

wenn $p_i = 0$ oder $1 \forall i$, dann $\Pr(X_1 \oplus \dots \oplus X_n = 0) = 0$ oder 1
wenn ein $p_i = 1/2$, dann $\Pr(X_1 \oplus \dots \oplus X_n = 0) = 1/2$

Beispiel zur Entwicklung einer linearen Approximation:

geg.: $X_1 \dots X_4$ mit $\Pr(X_1 \oplus X_2 = 0) = 1/2 + \epsilon_{1,2}$ und $\Pr(X_2 \oplus X_3 = 0) = 1/2 + \epsilon_{2,3}$

dabei wird $X_1 \oplus X_3$ aus $X_1 \oplus X_2$ und $X_2 \oplus X_3$ hergeleitet:

$$\Pr(X_1 \oplus X_3 = 0) = \Pr([X_1 \oplus X_2] \oplus [X_2 \oplus X_3] = 0)$$

Anwendung des Piling-Up Lemma:

$$\Pr(X_1 \oplus X_3 = 0) = 1/2 + 2\epsilon_{1,2}\epsilon_{2,3}$$

es folgt:

$$\epsilon_{1,3} = 2\epsilon_{1,2}\epsilon_{2,3}$$

$X_1 \oplus X_2 = 0$ und $X_1 \oplus X_3 = 0$ kann nun entsprechend für lineare Approximation von S-Boxen (SPN) verwendet werden. Für Ciffren Approximation eignet sich $X_1 \oplus X_3 = 0$.

3 Quellen

- 1 Douglas R. Stinson: "Cryptography: Theory and Practice.", 2nd Edition, Chapman & Hall/CRC 2002.
- 2 M. Matsui, "Linear Cryptanalysis Method for DES Cipher", Advances in Cryptology - EUROCRYPT '93 (Lecture Notes in Computer Science no. 765), Springer-Verlag, pp. 386-397, 1994.