

Proseminar Kryptographie

Prof. Dr. Chr. Kreitz, Dr. E. Richter

18. Oktober 2004

1. Einfache kryptographische Systeme

Datum: 26.10.04

Literatur: Kapitel 1.1

- ▶ formale Definition eines Kryptosystems,
- ▶ monoalphabetische Verschlüsselungen: Ersetzung und Verschiebung,
- ▶ Verschlüsselungen mit wechselnden Schlüsseln: Vigenère, Hill, Permutationen auf Blöcken.

2. Kryptoanalyse

Datum: 02.11.04

Literatur: Kapitel 1.2

Techniken zum "Knacken" von Verschlüsselungen

- ▶ verschiedene Angriffsmodelle: Schlüsseltextangriff, Bekannte-Quelle-Angriff, Ausgewählte-Quelle-Angriff, Ausgewählter- Schlüsseltext-Angriff,
- ▶ Angriffe auf die im 1. Vortrag vorgestellten Verschlüsselungen mit Hilfe von elementarer Statistik.

3. Datensicherheit und Shannon's Informationstheorie

Datum: 09.11.04

Literatur: Kapitel 2

- ▶ Begriffe von Sicherheit: berechenbare Sicherheit, beweisbar sicher, unbedingt sicher,
- ▶ Grundbegriffe der Wahrscheinlichkeitstheorie (Zufallsgröße, bedingte Wahrscheinlichkeit),
- ▶ *Entropie* als Maß des Informationsgehaltes bzw. der Unsicherheit,
- ▶ Produktverschlüsselungen.

4. Blockverschlüsselungen

Datum: 16.11.04

Literatur: Kapitel 3.1- 3.4

- ▶ Verschlüsselung erfolgt blockweise mit einer Serie von Produktschlüsseln,
- ▶ Beispiel Ersetzungs-Permutations-Schlüsselkombination,
- ▶ Lineare Kryptoanalyse (Bekannter-Quelltext-Angriff):
lineare probabilistische Beziehung zwischen Quelltext-Bits und Zustands-Bits nach der vorletzten Runde,
mögliche Schlüsselkandidaten für die letzte Runde werden anhand der lineare Beziehung bewertet,
- ▶ Anhäufungslemma über kombinierte zweiwertige Zufallsvariablen.

5. Die Verschlüsselungsstandards DES und AES

Datum: 23.11.04

Literatur: Kapitel 3.5-3.6

- ▶ Differentielle Kryptoanalyse (Ausgewählter-Quelltext-Angriff): untersucht probabilistische Beziehung zwischen $x \oplus x^*$ und $y \oplus y^*$; mögliche Schlüsselkandidaten für die letzte Runde werden anhand der probabilistischen Beziehung bewertet,
- ▶ Data Encryption Standard:
 1. ausgeschrieben von NIST in den 70-er Jahren,
 2. spezielle FEISTEL-Verschlüsselung: Zahl n festlegen, Bitstring S der Länge $2n$ aufteilen $S = (L, R)$,
 $F_t(L, R) = (L \oplus f(R), R)$ für eine n -stellige Funktion f ,
 3. interessante Geschichte bzgl. der Angriffsversuche.

▶ Advanced Encryption Standard:

1. ausgeschrieben von NIST 1999, Gütekriterien: Sicherheit, Kosten, Algorithmen und Implementierungseigenschaften,
2. RIJNDEAL: verwendet eine Schlüsselmischung, einen Ersetzungsschritt in jeder Runde, einen Permutationsschritt,
3. hat längere Schlüssel und enthält eine zusätzliche lineare Transformation,
4. ist sicher in Bezug auf alle bekannten Attacken.

6. Kryptographische Hash Funktionen

Datum: 30.11.04

Literatur: Kapitel 4.1-4.3

- ▶ dienen der Authentifizierung von Nachrichten ohne Schlüssel,
- ▶ zu einer Nachricht x wird eine Signatur $h(x) = y$ erzeugt,
- ▶ Komplexität von Sicherheitsproblemen (Urbild, Zweites-Urbild, Kollision) werden mit Hilfe des Zufalls-Orakel-Modells untersucht,
- ▶ iterierte Hash-Funktionen als Methode zur Konstruktion von beweisbar sicheren Hash-Funktionen aus sicheren Kompressionsfunktionen.

7. Message Authentication Codes

Datum: 07.12.04

Literatur: Kapitel 4.4-4.5

- ▶ MACs sind verschlüsselte Hash-Funktionen mit bestimmten Sicherheitsanforderungen,
- ▶ allgemeine Konstruktionen und Sicherheitsbeweise bzw. Angriffsmöglichkeiten,
- ▶ Konstruktion von unbedingt sicheren MACs: jeder Schlüssel wird nur einmal benutzt; die Wahrscheinlichkeit für den Erfolg eines Einmal-Angriffs fällt unter eine gewisse Schranke (unabhängig von der Rechenleistung)
- ▶ Streng universelle Hash-Familien: die Anzahl der Schlüssel, mit denen sich Kollisionen erzeugen lassen, ist kleinstmöglich.

8. Public-key Kryptographie mit dem RSA Schema

Datum: 14.12.04

Literatur: Kapitel 5.1-5.3

- ▶ Idee: die Berechnung von d_K aus bekanntem e_K ist zu "schwierig", es werden sogenannte *One-Way-Funktionen* zur Verschlüsselung benutzt,
- ▶ RSA (Rivest, Shamir, Adleman, 1977) basiert auf der Schwierigkeit zur Primzahlzerlegung großer Zahlen,

$$e(x) = x^b \pmod{n} \quad d(y) = y^{b^{-1}} \pmod{n},$$

für $n = p * q$ ist die Berechnung von b^{-1} „leicht“,

- ▶ Euklidischer Algorithmus, Chinesischer Restsatz,
- ▶ RSA-Algorithmus und Komplexitätsabschätzungen.

9. Primzahltests und Faktorisierung

Datum: 04.01.05

Literatur: Kapitel 5.4-5.6

- ▶ für RSA braucht man *große* Primzahlen, also erzeugt man große Zahlen und testet,
- ▶ verschiedene Testverfahren: SOLVARY-STRASSEN, MILLER-RABIN,
- ▶ RSA ist geknackt, wenn man die Primfaktorzerlegung von n kennt,
- ▶ effektive Faktoralgorithmen.

10. Andere Attacken auf RSA, Rabin's Kryptosystem

Datum: 11.01.05

Literatur: Kapitel 5.7-5.9

- ▶ Falls man durch Zufall b^{-1} bekommt, kann man die Faktorisierung von n in polynomieller Zeit berechnen, also ist nicht nur b sondern auch n wertlos, *RSA-FACTOR*(n, a, b)
- ▶ WIENER'S ALGORITHMUS(n, b) zur Berechnung von a ,
- ▶ RABINS Kryptosystem: $n = p * q$, $p, q \equiv 3 \pmod{4}$,
 $e_K(x) = x^2 \pmod{n}$, $d_K(y) = \sqrt{y} \pmod{n}$, beweisbar sicheres Kryptosystem, aber Verschlüsselung nicht injektiv,
- ▶ Sicherheitsbeweis für RABIN über Problemreduktion,
- ▶ Sicherheit von RSA bzgl. verschiedener Angriffsziele: *Partielle Entschlüsselung, Unterscheidbarkeit von Text und Zufall.*

11. Public-key Kryptographie mit Diskreten Logarithmen

Datum: 18.01.05

Literatur: Kapitel 6.1-6.3

- ▶ $(G, *)$ multiplikative Gruppe, $\langle \alpha \rangle$ ist die von $\alpha \in G$ erzeugte endliche zyklische Untergruppe,
- ▶ Diskreter Logarithmus: $f : \langle \alpha \rangle \rightarrow \{0, \dots, \text{ord}(\alpha)\}$

$$f(\beta) = \log_{\alpha} \beta = a \text{ mit } \alpha^a = \beta$$

- ▶ ElGamal-Verschlüsselung in \mathbb{Z}_p^* (geheim: a, k öffentlich: p, α, β)

$$e_K(x, k) = (\alpha^k \text{ mod } p, x * \beta^k \text{ mod } p)$$

- ▶ Entschlüsselung: $d_K(y_1, y_2) = y_2(y_1^a)^{-1} \text{ mod } p$
- ▶ verschiedene Algorithmen zur Berechnung diskreter Logarithmen.

11. Public-key Kryptographie mit Diskreten Logarithmen

Datum: 18.01.05

Literatur: Kapitel 6.1-6.3

- ▶ $(G, *)$ multiplikative Gruppe, $\langle \alpha \rangle$ ist die von $\alpha \in G$ erzeugte endliche zyklische Untergruppe,
- ▶ Diskreter Logarithmus: $f : \langle \alpha \rangle \rightarrow \{0, \dots, \text{ord}(\alpha)\}$

$$f(\beta) = \log_{\alpha} \beta = a \text{ mit } \alpha^a = \beta$$

- ▶ ElGamal-Verschlüsselung in \mathbb{Z}_p^* (geheim: a, k öffentlich: p, α, β)

$$e_K(x, k) = (\alpha^k \bmod p, x * \beta^k \bmod p)$$

- ▶ Entschlüsselung: $d_K(y_1, y_2) = y_2 (y_1^a)^{-1} \bmod p$
- ▶ verschiedene Algorithmen zur Berechnung diskreter Logarithmen.

11. Public-key Kryptographie mit Diskreten Logarithmen

Datum: 18.01.05

Literatur: Kapitel 6.1-6.3

- ▶ $(G, *)$ multiplikative Gruppe, $\langle \alpha \rangle$ ist die von $\alpha \in G$ erzeugte endliche zyklische Untergruppe,
- ▶ Diskreter Logarithmus: $f : \langle \alpha \rangle \rightarrow \{0, \dots, \text{ord}(\alpha)\}$

$$f(\beta) = \log_{\alpha} \beta = a \text{ mit } \alpha^a = \beta$$

- ▶ ElGamal-Verschlüsselung in \mathbb{Z}_p^* (geheim: a, k öffentlich: p, α, β)

$$e_K(x, k) = (\alpha^k \text{ mod } p, x * \beta^k \text{ mod } p)$$

- ▶ Entschlüsselung: $d_K(y_1, y_2) = y_2(y_1^a)^{-1} \text{ mod } p$
- ▶ verschiedene Algorithmen zur Berechnung diskreter Logarithmen.

11. Public-key Kryptographie mit Diskreten Logarithmen

Datum: 18.01.05

Literatur: Kapitel 6.1-6.3

- ▶ $(G, *)$ multiplikative Gruppe, $\langle \alpha \rangle$ ist die von $\alpha \in G$ erzeugte endliche zyklische Untergruppe,
- ▶ Diskreter Logarithmus: $f : \langle \alpha \rangle \rightarrow \{0, \dots, \text{ord}(\alpha)\}$

$$f(\beta) = \log_{\alpha} \beta = a \text{ mit } \alpha^a = \beta$$

- ▶ ElGamal-Verschlüsselung in \mathbb{Z}_p^* (geheim: a, k öffentlich: p, α, β)

$$e_K(x, k) = (\alpha^k \text{ mod } p, x * \beta^k \text{ mod } p)$$

- ▶ Entschlüsselung: $d_K(y_1, y_2) = y_2(y_1^a)^{-1} \text{ mod } p$
- ▶ verschiedene Algorithmen zur Berechnung diskreter Logarithmen.

11. Public-key Kryptographie mit Diskreten Logarithmen

Datum: 18.01.05

Literatur: Kapitel 6.1-6.3

- ▶ $(G, *)$ multiplikative Gruppe, $\langle \alpha \rangle$ ist die von $\alpha \in G$ erzeugte endliche zyklische Untergruppe,
- ▶ Diskreter Logarithmus: $f : \langle \alpha \rangle \rightarrow \{0, \dots, \text{ord}(\alpha)\}$

$$f(\beta) = \log_{\alpha} \beta = a \text{ mit } \alpha^a = \beta$$

- ▶ ElGamal-Verschlüsselung in \mathbb{Z}_p^* (geheim: a, k öffentlich: p, α, β)

$$e_K(x, k) = (\alpha^k \text{ mod } p, x * \beta^k \text{ mod } p)$$

- ▶ Entschlüsselung: $d_K(y_1, y_2) = y_2(y_1^a)^{-1} \text{ mod } p$
- ▶ verschiedene Algorithmen zur Berechnung diskreter Logarithmen.

12. Sicherheit von ElGamal Systemen

Datum: 18.01.05

Literatur: Kapitel 6.5-6.7

- ▶ Gruppen, für die die Berechnung diskreter Logarithmen „unmöglich“ ist: multiplikative Gruppen über endlichen Körpern, Gruppen von Punkten auf elliptischen Kurven über endlichen Körpern,
- ▶ Konstruktion von endlichen Körpern mit p^n Elementen,
- ▶ Elliptische Kurven, zweistellige kommutative, assoziative Operation auf Punkten mit neutralem Element und Inversen,
- ▶ Sicherheitsabschätzungen für ElGamal Funktionen.

13. Digitale Unterschriften

Datum: 25.01.05

Literatur: Kapitel 7.1-7.4

- ▶ es gibt keine mechanische Verbindung zwischen Dokument und elektronischer Unterschrift, elektronische Kopien sind nicht voneinander zu unterscheiden,
- ▶ Signaturschema: 1. Algorithmus zum Erzeugen der Unterschrift (geheim), 2. Algorithmus zum Verifizieren (öffentlich),
- ▶ verschiedene Angriffe, verschiedene Angriffsziele,
- ▶ da Verifikationsalgorithmus öffentlich, gibt es keine unbedingt sicheren Signaturen,
- ▶ Signaturalgorithmen mit hoher Komplexität der Unterschriftserzeugung.

14. Nachweislich sichere Signatursysteme

Datum: 01.02.05

Literatur: Kapitel 7.5-7

- ▶ Einmal-Unterschriften: LAMPORT-SIGNATURE-SYSTEM, FULL-DOMAIN-HASH,
- ▶ (CHAUM und VAN ANTWERPEN,1989) Unbestreitbare Unterschriften: Verifikation kann nicht ohne Mitarbeit des Senders erfolgen (Frage-Antwort-Protokoll), Unterzeichner beweist die Fälschung (Nichtanerkennungsprotokoll).